

Lampiran

Lampiran 1 Instrumen Penelitian

Lampiran 1.1. Lembar Instrumen Angket Media Pembelajaran

Lampiran 1.2. Lembar Instrumen Angket Materi Pembelajaran

Lampiran 1.3. Lembar Instrumen Angket Keefektifan Media Pembelajaran

Lampiran 1.4. Uji *Blackbox*

Lampiran 1.1. Lembar Instrumen Angket Media Pembelajaran

ANGKET PENILAIAN AHLI MEDIA

Dalam rangka penelitian Tugas Akhir Skripsi, saya mohon bantuan bapak / ibu / saudara untuk menjadi validator **“Pengembangan Aplikasi Pendeteksi Objek Bersegi Menggunakan Metode *Contour* Sebagai Media Pembelajaran Penginderaan Visual Robot”** agar layak digunakan sebagai media pembelajaran.

Mata Kuliah : Praktik Penginderaan Visual Robot
Pembuat : Ihsan Syarifuddin
Tanggal :

Prosedur Pengisian Instrumen Media Pembelajaran:

1. Dimohon kepada bapak / ibu / saudara untuk memberikan penilaian pada salah satu kolom di dalam angket ini berdasarkan kriteria berikut.
4 : Sangat Setuju
3 : Setuju
2 : Tidak Setuju
1 : Sangat Tidak Setuju
2. Dimohon untuk memberikan tanda *check* (✓) pada salah satu pilihan dalam setiap pernyataan sesuai pendapat bapak / ibu / saudara.
3. Jika terdapat kekurangan pada media pembelajaran ini, dimohon bapak / ibu / saudara memberikan saran dan masukan pada bagian yang telah disediakan.
4. Mohon untuk menuliskan nama sebagai validator (jika belum tersedia) pada bagian akhir lembar instrumen ini sebagai bukti bahwa instrumen ini dibuat sebenar-benarnya tanpa manipulasi data.

Atas bantuan bapak / ibu / saudara, kami mengucapkan terima kasih.

Pernyataan:

No.	Pernyataan	Pilihan Jawaban			
		4	3	2	1
1	Penggunaan media pembelajaran aplikasi pendeteksi objek bersegi dapat memberikan informasi mengenai tujuan pembelajaran penginderaan visual robot pada peserta didik.				
2	Penggunaan media pembelajaran aplikasi pendeteksi objek bersegi dapat menjelaskan secara rinci mengenai tujuan belajar penginderaan visual robot pada peserta didik.				
3	Penggunaan media pembelajaran membantu mempermudah kegiatan pembelajaran praktik penginderaan visual robot.				
4	Media pembelajaran aplikasi pendeteksi objek bersegi dapat digunakan pengajar dalam menjelaskan materi agar menjadi lebih mudah difahami peserta didik.				
5	Media pembelajaran aplikasi pendeteksi objek bersegi dapat meningkatkan keaktifan peserta didik dalam pembelajaran.				
6	Penggunaan media pembelajaran aplikasi pendeteksi objek bersegi menumbuhkan semangat belajar peserta didik.				
7	Media pembelajaran aplikasi pendeteksi objek bersegi memiliki keterkaitan dengan mata kuliah lain.				
8	Media pembelajaran aplikasi pendeteksi objek bersegi dapat mendukung pembelajaran pada mata kuliah lain.				
9	Tampilan GUI pada media pembelajaran berfungsi dengan baik.				
10	Informasi pada tampilan GUI mudah difahami.				
11	Bentuk media pembelajaran menarik.				
12	Peletakan komponen pada media pembelajaran tersusun dengan baik.				
13	<i>Trackbar</i> pada media pembelajaran aplikasi pendeteksi objek bersegi mudah digunakan.				

14	<i>DomainUpDown</i> pada media pembelajaran mudah digunakan.				
15	Sensor kamera yang digunakan pada media pembelajaran dapat bekerja dengan baik.				
16	Semua komponen media pembelajaran berfungsi dengan baik.				
17	Media pembelajaran aplikasi pendeteksi objek bersegi lebih menarik dengan adanya tampilan GUI.				
18	Media pembelajaran dapat memberikan rasa ingin tahu kepada pengguna.				
19	Media pembelajaran menarik untuk digunakan.				
20	Media pembelajaran mudah digunakan.				

Kesimpulan:

Menurut saya, Media Pembelajaran Aplikasi Pendeteksi Objek Bersegi ini dinyatakan:

1. Layak digunakan tanpa revisi
2. Layak digunakan dengan revisi pada saran
3. Tidak layak digunakan.

Saran dan Perbaikan:

.....

.....

.....

.....

.....

.....

Yogyakarta,

Validator

(.....)

NIP.....

Lampiran 1.2. Lembar Instrumen Angket Materi Pembelajaran

ANGKET PENILAIAN AHLI MATERI

Dalam rangka penelitian tugas akhir skripsi, saya mohon bantuan bapak / ibu / saudara untuk menjadi validator **“Pengembangan Aplikasi Pendeteksi Objek Bersegi Menggunakan Metode *Contour* Sebagai Media Pembelajaran Penginderaan Visual Robot”** agar layak digunakan sebagai media pembelajaran.

Mata Kuliah : Praktik Penginderaan Visual Robot

Pembuat : Ihsan Syarifuddin

Tanggal :

Prosedur Pengisian Instrumen Materi Pembelajaran

1. Dimohon kepada bapak / ibu / saudara untuk memberikan penilaian pada salah satu kolom di dalam angket ini berdasarkan kriteria berikut:
4 : Sangat Setuju
3 : Setuju
2 : Tidak Setuju
1 : Sangat Tidak Setuju
2. Dimohon untuk memberikan tanda *check* (✓) pada salah satu pilihan dalam setiap pertanyaan sesuai pendapat bapak / ibu /saudara.
3. Jika terdapat kekurangan pada media pembelajaran ini, mohon bapak / ibu / saudara memberikan saran dan masukan pada bagian yang telah disediakan.
4. Mohon untuk menuliskan nama sebagai validator (jika belum tersedia) pada bagian akhir lembar instrumen ini sebagai bukti bahwa instrumen ini dibuat sebenar-benarnya tanpa manipulasi data.
5. Atas bantuan bapak / ibu / saudara, kami mengucapkan terima kasih.

Pernyataan:

No	Pernyataan	Pilihan Jawaban			
		4	3	2	1
1	Materi pelajaran yang diajarkan menggunakan media pembelajaran sesuai dengan silabus.				
2	Materi yang diajarkan dalam media aplikasi pendeteksi objek bersegi relevan dengan kompetensi peserta didik pada bidang penginderaan visual robot.				
3	Materi yang dicantumkan dalam media pembelajaran sudah lengkap.				
4	Cara menghubungkan <i>library</i> EmguCV pada media pembelajaran aplikasi pendeteksi objek bersegi dijelaskan di dalam modul.				
5	Cara kerja deteksi objek bersegi dijelaskan di dalam modul.				
6	Cara membuat program yang digunakan dalam pemrograman aplikasi pendeteksi objek bersegi dijelaskan di dalam modul.				
7	Materi tentang dasar penginderaan visual robot disajikan dengan jelas.				
8	Materi tentang deteksi <i>Contour</i> disajikan dengan jelas.				
9	Alur pengoperasian media pembelajaran disajikan dengan jelas pada menu Bantuan.				
10	Materi pelajaran lebih mudah dipahami setelah menggunakan media pembelajaran aplikasi pendeteksi objek bersegi.				

11	Materi yang terdapat pada modul didukung dengan adanya media pembelajaran aplikasi pendeteksi objek bersegi.				
12	Materi pelajaran sudah sesuai dengan tuntutan lapangan pekerjaan di bidang otomasi industri.				
13	Materi disajikan secara runtut.				
14	Materi yang disajikan didukung dengan dasar teori yang jelas.				
15	Materi yang dibuat dapat meningkatkan motivasi belajar peserta didik.				
16	Materi yang disajikan mendukung keterlibatan peserta didik dalam pembelajaran penginderaan visual robot.				
17	Materi yang disajikan dapat dibaca dengan jelas dan sesuai dengan perkembangan intelektual peserta didik.				
18	Materi yang disajikan dapat dibaca dengan jelas dan sesuai dengan perkembangan emosional peserta didik.				
19	Tata bahasa pada modul sudah baku.				
20	Istilah-istilah pada modul sudah baku.				
21	Materi yang disajikan desain <i>cover</i> -nya menarik daya minat peserta didik sehingga tertarik dan betah untuk belajar.				
22	Materi yang disajikan desain isi-nya menarik daya minat peserta didik sehingga tertarik dan betah untuk belajar.				

23	Kualitas cetakan materi sesuai standar di pasaran buku.				
24	Kualitas jilidan materi sesuai standar di pasaran buku.				

Kesimpulan:

Menurut saya, materi untuk aplikasi pendeteksi objek bersegi dinyatakan:

1. Layak digunakan tanpa revisi
2. Layak digunakan dengan revisi pada saran
3. Tidak layak digunakan

Saran dan Perbaikan:

.....

Yogyakarta,

Validator

(.....)

NIP.....

Lampiran 1.3. Lembar Instrumen Angket Keefektifan Media Pembelajaran

ANGKET PENILAIAN
PENGEMBANGAN APLIKASI PENDETEKSI OBJEK BERSEGI
MENGUNAKAN METODE *CONTOUR*
SEBAGAI MEDIA PEMBELAJARAN
PRAKTIK PENGINDERAAN VISUAL ROBOT



IDENTITAS PESERTA DIDIK

Nama :

NIM :

PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2018

Angket Penilaian Media

Hal : Pengisian Angket Penilaian

Kepada : Peserta didik Program Studi Pendidikan Teknik Mekatronika

Assalamu'alaikum Wr.Wb.

Dengan hormat,

Mohon kesediaan dan bantuan saudara/i untuk meluangkan waktu guna mengisi angket ini. Angket ini berguna untuk mengumpulkan data terkait dengan “Pengembangan Aplikasi Pendeteksi Objek Bersegi Menggunakan Metode *Contour* Sebagai Media Pembelajaran Praktik Penginderaan Visual Robot”.

Angket ini bukan merupakan tes, sehingga jawaban yang anda berikan tidak akan memengaruhi nilai mata kuliah. Jawaban yang baik adalah jawaban yang sesuai dengan kenyataan dan diisi berdasarkan hati nurani saudara/i, serta akan kami jamin kerahasiaannya. Kejujuran saudara/i dalam menjawab angket ini sangat diharapkan demi mendapatkan hasil penelitian yang maksimal.

Atas bantuan dan kerjasama dari saudara/i, saya ucapkan terimakasih.

Wassalamu'alaikum Wr.Wb.

Yogyakarta, 12 Oktober 2018

Hormat saya,

Peneliti

A. Petunjuk Pengisian Angket

1. Angket ini dimaksudkan untuk mengetahui pendapat/penilaian anda sebagai pengguna media pembelajaran APLIKASI PENDETEKSI OBJEK BERSEGI.
2. Anda diharapkan memilih salah satu pilihan jawaban pada setiap pernyataan yang tersedia dengan memberikan TANDA SILANG (X) pada kolom jawaban.

NO	PERNYATAAN	JAWABAN			
1	Desain tata letak komponen pada GUI sudah rapi.	1	2	X	4

3. Jika anda ingin mengubah jawaban, maka berikan tanda SAMA DENGAN (=) pada pilihan jawaban yang akan diganti dan memberikan TANDA SILANG (X) pada kolom penggantinya.

NO	PERNYATAAN	JAWABAN			
1	Desain tata letak komponen pada GUI sudah rapi.	1	=	X	4

4. Keterangan jawaban:
1 = Sangat Tidak Setuju / Sangat Tidak Sesuai / Sangat Tidak Baik
2 = Tidak Setuju / Tidak Sesuai / Tidak Baik
3 = Setuju / Sesuai / Baik
4 = Sangat Setuju / Sangat Sesuai / Sangat Baik
5. Komentar atau saran anda mohon ditulis pada lembar yang telah disediakan. Atas kesediaan anda untuk mengisi angket ini, saya ucapkan terima kasih

B. Angket Penilaian

NO	PERNYATAAN	JAWABAN			
1	Media pembelajaran yang dibuat sesuai dengan materi pembelajaran yang diajarkan.	1	2	3	4
2	Materi yang ada dalam modul mudah untuk dipahami.	1	2	3	4
3	Contoh program pada modul mudah untuk dipahami.	1	2	3	4
4	Materi yang disajikan sesuai dengan mata kuliah praktik penginderaan visual robot.	1	2	3	4
5	Materi yang disajikan berisi kompetensi yang dibutuhkan.	1	2	3	4
6	Ilustrasi dalam modul mempermudah dalam melaksanakan kegiatan praktikum.	1	2	3	4
7	Langkah kerja dalam modul mudah untuk diikuti.	1	2	3	4
8	Komponen pada GUI tidak membuat bingung.	1	2	3	4
9	Pengoperasian media pembelajaran aplikasi pendeteksi objek bersegi dapat dilakukan dengan mudah.	1	2	3	4
10	GUI sebagai hasil <i>output</i> data berfungsi dengan baik dan akurat.	1	2	3	4
11	Desain media pembelajaran menarik.	1	2	3	4
12	Media pembelajaran aplikasi pendeteksi objek bersegi memberi tambahan pengetahuan dalam pemrograman pada mata kuliah praktik penginderaan visual robot.	1	2	3	4
13	Media pembelajaran aplikasi pendeteksi objek bersegi memberi tambahan pengetahuan tentang penginderaan visual robot yang lebih kompleks.	1	2	3	4
14	Media pembelajaran aplikasi pendeteksi objek bersegi dapat membantu dalam memahami materi pada mata kuliah lain.	1	2	3	4
15	Media pembelajaran aplikasi pendeteksi objek bersegi membantu mempelajari pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	3	4

16	Media pembelajaran aplikasi pendeteksi objek bersegi dapat memotivasi untuk belajar pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	3	4
17	Media pembelajaran aplikasi pendeteksi objek bersegi menambah kompetensi dalam pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	3	4
18	Media pembelajaran aplikasi pendeteksi objek bersegi dapat meningkatkan keaktifan belajar saat pembelajaran praktik penginderaan visual robot.	1	2	3	4
19	Media pembelajaran aplikasi pendeteksi objek bersegi efektif dalam meningkatkan kemampuan belajar peserta didik praktik penginderaan visual robot.	1	2	3	4
20	Media pembelajaran aplikasi pendeteksi objek bersegi secara efisien dapat meningkatkan kemampuan belajar peserta didik.	1	2	3	4

C. Komentar dan Saran Umum

.....

.....

.....

Yogyakarta,

(.....)

Lampiran 1.4. Uji *Blackbox*

Lembar Uji *Blackbox*

Mata Kuliah : Praktik Penginderaan Visual Robot
Sasaran : Mahasiswa Pendidikan Teknik Mekatronika, FT, UNY
Judul : Pengembangan Aplikasi Pendeteksi Objek Bersegi
Menggunakan Metode *Contour* Sebagai Media
Pembelajaran Penginderaan Visual Robot
Peneliti : Ihsan Syarifuddin

Dalam rangkai penelitian Tugas Akhir Skripsi saya mohon bantuan Saudara untuk menjadi penguji kinerja aplikasi pendeteksi objek bersegi sebagai media pembelajaran untuk mata kuliah praktik penginderaan visual robot ini agar dapat menjadi media inovatif yang layak digunakan oleh mahasiswa.

A. Petunjuk Pengisian Angket

1. Operasikan media pembelajaran aplikasi pendeteksi objek bersegi sesuai pernyataan dalam angket.
2. Jawaban diberikan berdasarkan kesesuaian antara pernyataan pada angket dengan kondisi aktual pada media pembelajaran aplikasi pendeteksi objek bersegi.
3. Mohon diberikan tanda *checklist* (✓) pada kolom penilaian sesuai dengan pendapat saudara.
4. Apabila terdapat kekurangan, mohon kiranya dapat memberikan masukan pada kotak saran.

B. Angket Uji *Blackbox*

No.	Keterangan	Fungsi	
		Ya	Tidak
1	Fungsi <i>button</i> Pengaturan		
2	Fungsi <i>button</i> Bantuan		
3	Fungsi <i>button</i> Tentang		
4	Fungsi <i>button</i> Play		
5	Fungsi <i>button</i> Pause		
6	Fungsi <i>button</i> Stop		
7	Fungsi <i>button</i> View Code		
8	Fungsi <i>comboBox</i> untuk pemilihan Kamera pada <i>Form</i> Pengaturan		
9	Fungsi <i>checkBox</i> untuk pemilihan Objek pada <i>Form</i> Pengaturan		
10	Fungsi <i>comboBox</i> Tampilan		
11	Fungsi <i>button</i> Terapkan		
12	Fungsi <i>imageBox</i> untuk tampilan Gambar Asli		
13	Fungsi <i>imageBox</i> untuk tampilan Blur		
14	Fungsi <i>imageBox</i> untuk tampilan Canny		
15	Fungsi <i>imageBox</i> untuk tampilan Hasil Akhir		
16	Fungsi <i>comboBox</i> untuk pemilihan jenis Blur		
17	Fungsi <i>domainUpDown</i> untuk nilai blur		
18	Fungsi <i>trackbar</i> nilai <i>Threshold1</i>		
19	Fungsi <i>domainUpDown</i> nilai pengali untuk <i>Threshold2</i>		
20	Fungsi <i>domainUpDown</i> untuk nilai Akurasi <i>Exterior Angle</i>		
21	Fungsi penampil dari Parameter Keseluruhan		
22	Fungsi deteksi Segitiga		
23	Fungsi deteksi Segiempat		

24	Fungsi deteksi Segilima		
25	Fungsi deteksi Segienam		

C. Komentari dan Saran Umum

.....

.....

.....

.....

Yogyakarta,

(.....)

Lampiran 2 Validasi Instrumen Penelitian

Lampiran 2.1. Lembar Validasi Instrumen Penelitian Adopsi

Lampiran 2.1. Lembar Validasi Instrumen Penelitian Adopsi

SURAT PERMOHONAN VALIDASI INSTRUMEN PENELITIAN

Hal : Permohonan Validasi Media TAS
Lampiran : 1 Bendel

Kepada Yth,

Dr. Dra. Zamtinah, M.Pd.
Dosen Jurusan Pendidikan Teknik Elektro
Di Fakultas Teknik UNY

Sehubungan dengan rencana pelaksanaan Tugas Akhir Skripsi (TAS),
dengan ini saya:

Nama : Muhammad Rafii Naufal
NIM : 14518244003
Program Studi : Pendidikan Teknik Mekatronika
Judul TAS : Pengembangan Aplikasi Pendeteksi Objek Lingkaran
Menggunakan Metode *Houghcircle* Sebagai Media
Pembelajaran Penginderaan Visual Robot

dengan hormat, mohon Bapak/Ibu berkenan memberikan validasi terhadap media
penelitian TAS yang telah saya susun. Sebagai bahan pertimbangan, bersama ini
saya lampirkan: (1) Proposal TAS, (2) Kisi-kisi Instrumen Penelitian, dan
(3) Draft Instrumen Penelitian TAS.

Demikian permohonan Saya, atas bantuan perhatian Bapak / Ibu, saya
ucapkan terima kasih

Yogyakarta, 10 April 2018

Pemohon,



Muhammad Rafii Naufal
NIM. 14518244003

Mengetahui,

Kaprodi Pendidikan Teknik
Mekatronika



Herlambang Sigit Pramono, S.T., M.Sc
NIP. 19650829 199903 1 001

Pembimbing TAS



Herlambang Sigit Pramono, S.T., M.Sc
NIP. 19650829 199903 1 001

SURAT PERNYATAAN VALIDASI
INSTRUMEN PENELITIAN TUGAS AKHIR SKRIPSI

Saya yang bertanda tangan di bawah ini:

Nama : Dr. Dra. Zamtinah, M.Pd.
NIP : 196202171989032002
Jurusan : Pendidikan Teknik Elektro

Menyatakan bahwa instrumen penelitian TAS tersebut atas nama mahasiswa:

Nama : Muhammad Rafii Naufal
NIM : 14518244003
Program Studi : Pendidikan Teknik Mekatronika
Judul TAS : Pengembangan Aplikasi Pendeteksi Objek Lingkaran
Menggunakan Metode *Houghcircle* Sebagai Media
Pembelajaran Penginderaan Visual Robot

Setelah dilakukan kajian atas instrument penelitian TAS tersebut dapat dinyatakan:

- ☐ Layak digunakan untuk penelitian
☒ Layak digunakan dengan perbaikan
☐ Tidak layak digunakan untuk penelitian yang bersangkutan

dengan saran/ perbaikan sebagaimana terlampir.

Demikian agar dapat digunakan sebagaimana semestinya

Yogyakarta, 12 April 2018

Validator,



Dr. Dra. Zamtinah, M.Pd.

NIP. 196202171989032002

Catatan:

☐ Beri tanda ✓

HASIL VALIDASI INSTRUMEN PENELITIAN TAS

Nama : Muhammad Rafii Naufal
 NIM : 14518244003
 Program Studi : Pendidikan Teknik Mekatronika
 Judul TAS : Pengembangan Aplikasi Pendeteksi Objek Lingkaran
 Menggunakan Metode *Houghcircle* Sebagai Media Pembelajaran Penginderaan Visual Robot

No	Variabel	Saran / Tanggapan
1.	Uji Black Box	1. Perlu ditambahkan uji kinerja sistem secara menyeluruh. 2. Perlu ditambahkan buku?.
		Hj. K3 dan sistem, serta integrasi dg. standar yg. berlaku, mis. uk. Standar PPL.
2.	Instrumen uk. kegunaan	1. Perlu ditambahkan buku Hj. efektivitas & efisiensi media
3.	Uji uk. instrumen	media & materi p. Hj. efektivitas dan efisiensi media dr. tujuan pbn.
4.	Instrumen materi	1. Perlu ditambahkan buku integrasi standar dg. standar capaian CBT/VI
Komentar Umum / Lain-lain: Tels dan tabel cukup & spasi saja.		

Dr. Istikh

Dr. Yandelys

Hj.

Yogyakarta, 12 April 2018

Validator,

Dr. Dra. Zamtinah, M.Pd.

NIP. 196202171989032002

SURAT PERMOHONAN VALIDASI INSTRUMEN PENELITIAN

Hal : Permohonan Validasi Media TAS
Lampiran : 1 Bendel

Kepada Yth,

Dr. Nurhening Yuniarti, S.Pd., M.T.
Dosen Jurusan Pendidikan Teknik Elektro
Di Fakultas Teknik UNY

Sehubungan dengan rencana pelaksanaan Tugas Akhir Skripsi (TAS),
dengan ini saya:

Nama : Muhammad Rafii Naufal
NIM : 14518244003
Program Studi : Pendidikan Teknik Mekatronika
Judul TAS : Pengembangan Aplikasi Pendeteksi Objek Lingkaran
Menggunakan Metode *Houghcircle* Sebagai Media
Pembelajaran Penginderaan Visual Robot

dengan hormat, mohon Bapak/Ibu berkenan memberikan validasi terhadap media
penelitian TAS yang telah saya susun. Sebagai bahan pertimbangan, bersama ini
saya lampirkan: (1) Proposal TAS, (2) Kisi-kisi Instrumen Penelitian, dan
(3) Draft Instrumen Penelitian TAS.

Demikian permohonan Saya, atas bantuan perhatian Bapak / Ibu, saya
ucapkan terima kasih

Yogyakarta, 13 April 2018
Pemohon,


Muhammad Rafii Naufal
NIM. 14518244003

Mengetahui,

Kaprodi Pendidikan Teknik
Mekatronika

Pembimbing TAS


Herlambang Sigit Pramono, S.T., M.Sc
NIP. 19650829 199903 1 001


Herlambang Sigit Pramono, S.T., M.Sc
NIP. 19650829 199903 1 001

**SURAT PERNYATAAN VALIDASI
INSTRUMEN PENELITIAN TUGAS AKHIR SKRIPSI**

Saya yang bertanda tangan di bawah ini:

Nama : Dr. Nurhening Yuniarti, S.Pd., M.T.
NIP : 197506092002122002
Jurusan : Pendidikan Teknik Elektro

Menyatakan bahwa instrumen penelitian TAS tersebut atas nama mahasiswa:

Nama : Muhammad Rafii Naufal
NIM : 14518244003
Program Studi : Pendidikan Teknik Mekatronika
Judul TAS : Pengembangan Aplikasi Pendeteksi Objek Lingkaran
Menggunakan Metode *Houghcircle* Sebagai Media
Pembelajaran Penginderaan Visual Robot

Setelah dilakukan kajian atas instrument penelitian TAS tersebut dapat dinyatakan:

- ☐ Layak digunakan untuk penelitian
☒ Layak digunakan dengan perbaikan
☐ Tidak layak digunakan untuk penelitian yang bersangkutan

dengan saran/ perbaikan sebagaimana terlampir.

Demikian agar dapat digunakan sebagaimana semestinya

Yogyakarta, 16 April 2018

Validator,



Dr. Nurhening Yuniarti, S.Pd., M.T.

NIP. 197506092002122002

Catatan:

☐ Beri tanda ✓

HASIL VALIDASI INSTRUMEN PENELITIAN TAS

Nama : Muhammad Rafii Naufal
NIM : 14518244003
Program Studi : Pendidikan Teknik Mekatronika
Judul TAS : Pengembangan Aplikasi Pendeteksi Objek Lingkaran Menggunakan Metode *Houghcircle* Sebagai Media Pembelajaran Penginderaan Visual Robot

Hasil Validasi :

Utk Instrumen (Kelengkapan Media) : Butir 1 sd 4
tdk sesuai dgn kriteria & indikator

Utk Instrumen (Kelengkapan materi) : Butir 3
harus disesuaikan dengan indikator.

Butir 19 breakdown menjadi 2 butir pernyataan. (lihat lembar Instrumen).

Yogyakarta, 16 April 2018

Validator,



Dr. Nurhening Yuniarti, S.Pd., M.T.

NIP. 197306092002122002

Lampiran 3 Hasil Uji *Blackbox*

Lampiran 3.1. Hasil Uji *Blackbox*

Lampiran 3.1. Hasil Uji *Blackbox*

Lembar Uji *Blackbox*

Mata Kuliah : Praktik Penginderaan Visual Robot
Sasaran : Mahasiswa Pendidikan Teknik Mekatronika, FT, UNY
Judul : Pengembangan Aplikasi Pendeteksi Objek Bersegi
Menggunakan Metode *Contour* Sebagai Media
Pembelajaran Penginderaan Visual Robot
Peneliti : Ihsan Syarifuddin

Dalam rangka penelitian Tugas Akhir Skripsi saya mohon bantuan Saudara untuk menjadi penguji kinerja aplikasi pendeteksi objek bersegi sebagai media pembelajaran untuk mata kuliah praktik penginderaan visual robot ini agar dapat menjadi media inovatif yang layak digunakan oleh mahasiswa.

A. Petunjuk Pengisian Angket

1. Operasikan media pembelajaran aplikasi pendeteksi objek bersegi sesuai pernyataan dalam angket.
2. Jawaban diberikan berdasarkan kesesuaian antara pernyataan pada angket dengan kondisi aktual pada media pembelajaran aplikasi pendeteksi objek bersegi.
3. Mohon diberikan tanda *checklist* (✓) pada kolom penilaian sesuai dengan pendapat saudara.
4. Apabila terdapat kekurangan, mohon kiranya dapat memberikan masukan pada kotak saran.

B. Angket Uji Blackbox

No.	Keterangan	Fungsi	
		Ya	Tidak
1	Fungsi <i>button</i> Pengaturan	✓	
2	Fungsi <i>button</i> Bantuan	✓	
3	Fungsi <i>button</i> Tentang	✓	
4	Fungsi <i>button</i> Play	✓	
5	Fungsi <i>button</i> Pause	✓	
6	Fungsi <i>button</i> Stop	✓	
7	Fungsi <i>button</i> View Code	✓	
8	Fungsi <i>comboBox</i> untuk pemilihan Kamera pada <i>Form</i> Pengaturan	✓	
9	Fungsi <i>checkBox</i> untuk pemilihan Objek pada <i>Form</i> Pengaturan	✓	
10	Fungsi <i>comboBox</i> Tampilan	✓	
11	Fungsi <i>button</i> Terapkan	✓	
12	Fungsi <i>imageBox</i> untuk tampilan Gambar Asli	✓	
13	Fungsi <i>imageBox</i> untuk tampilan Blur	✓	
14	Fungsi <i>imageBox</i> untuk tampilan Canny	✓	
15	Fungsi <i>imageBox</i> untuk tampilan Hasil Akhir	✓	
16	Fungsi <i>comboBox</i> untuk pemilihan jenis Blur	✓	
17	Fungsi <i>domainUpDown</i> untuk nilai blur	✓	
18	Fungsi <i>trackbar</i> nilai <i>Threshold1</i>	✓	
19	Fungsi <i>domainUpDown</i> nilai pengali untuk <i>Threshold2</i>	✓	
20	Fungsi <i>domainUpDown</i> untuk nilai Akurasi <i>Exterior Angle</i>	✓	
21	Fungsi penampil dari Parameter Keseluruhan	✓	
22	Fungsi deteksi Segitiga	✓	
23	Fungsi deteksi Segiempat	✓	
24	Fungsi deteksi Segilima	✓	

25	Fungsi deteksi Segienam	V	
----	-------------------------	---	--

C. Komentor dan Saran Umum

- Akan lebih baik jika terdapat auto scroll pada kiri/kanan dan bawah / atas untuk menyesuaikan layar
- font ukuran font untuk di besarkan agar lebih terlihat jelas warna kuning kalem

Yogyakarta,


(.....Yuli Ramono.....)

Lampiran 4 Hasil Validasi Produk

Lampiran 4.1. Hasil Validasi Media

Lampiran 4.2. Hasil Validasi Materi

Lampiran 4.1. Hasil Validasi Media

ANGKET PENILAIAN AHLI MEDIA

Dalam rangka penelitian Tugas Akhir Skripsi, saya mohon bantuan bapak / ibu / saudara untuk menjadi validator **“Pengembangan Aplikasi Pendeteksi Objek Bersegi Menggunakan Metode *Contour* Sebagai Media Pembelajaran Penginderaan Visual Robot”** agar layak digunakan sebagai media pembelajaran.

Mata Kuliah : Praktik Penginderaan Visual Robot
Pembuat : Ihsan Syarifuddin
Tanggal :

Prosedur Pengisian Instrumen Media Pembelajaran:

1. Dimohon kepada bapak / ibu / saudara untuk memberikan penilaian pada salah satu kolom di dalam angket ini berdasarkan kriteria berikut.
4 : Sangat Setuju
3 : Setuju
2 : Tidak Setuju
1 : Sangat Tidak Setuju
2. Dimohon untuk memberikan tanda *check* (✓) pada salah satu pilihan dalam setiap pernyataan sesuai pendapat bapak / ibu / saudara.
3. Jika terdapat kekurangan pada media pembelajaran ini, dimohon bapak / ibu / saudara memberikan saran dan masukan pada bagian yang telah disediakan.
4. Mohon untuk menuliskan nama sebagai validator (jika belum tersedia) pada bagian akhir lembar instrumen ini sebagai bukti bahwa instrumen ini dibuat sebenar-benarnya tanpa manipulasi data.
5. Atas bantuan bapak / ibu / saudara, kami mengucapkan terima kasih.

Pernyataan:

No.	Pernyataan	Pilihan Jawaban			
		4	3	2	1
1	Penggunaan media pembelajaran aplikasi pendeteksi objek bersegi dapat memberikan informasi mengenai tujuan pembelajaran penginderaan visual robot pada peserta didik.			✓	
2	Penggunaan media pembelajaran aplikasi pendeteksi objek bersegi dapat menjelaskan secara rinci mengenai tujuan belajar penginderaan visual robot pada peserta didik.			✓	
3	Penggunaan media pembelajaran membantu mempermudah kegiatan pembelajaran praktik penginderaan visual robot.		✓		
4	Media pembelajaran aplikasi pendeteksi objek bersegi dapat digunakan pengajar dalam menjelaskan materi agar menjadi lebih mudah difahami peserta didik.		✓		
5	Media pembelajaran aplikasi pendeteksi objek bersegi dapat meningkatkan keaktifan peserta didik dalam pembelajaran.		✓		
6	Penggunaan media pembelajaran aplikasi pendeteksi objek bersegi menumbuhkan semangat belajar peserta didik.		✓		
7	Media pembelajaran aplikasi pendeteksi objek bersegi memiliki keterkaitan dengan mata kuliah lain.		✓		
8	Media pembelajaran aplikasi pendeteksi objek bersegi dapat mendukung pembelajaran pada mata kuliah lain.		✓		
9	Tampilan GUI pada media pembelajaran berfungsi dengan baik.		✓		
10	Informasi pada tampilan GUI mudah difahami.		✓		
11	Bentuk media pembelajaran menarik.		✓		
12	Peletakan komponen pada media pembelajaran tersusun dengan baik.		✓		
13	Trackbar pada media pembelajaran aplikasi pendeteksi objek bersegi mudah digunakan.		✓		

14	DomainUpDown pada media pembelajaran mudah digunakan.		✓		
15	Sensor kamera yang digunakan pada media pembelajaran dapat bekerja dengan baik.		✓		
16	Semua komponen media pembelajaran berfungsi dengan baik.		✓		
17	Media pembelajaran aplikasi pendeteksi objek bersegi lebih menarik dengan adanya tampilan GUI.		✓		
18	Media pembelajaran dapat memberikan rasa ingin tahu kepada pengguna.		✓		
19	Media pembelajaran menarik untuk digunakan.	✓			
20	Media pembelajaran mudah digunakan.		✓		

Kesimpulan:

Menurut saya, Media Pembelajaran Aplikasi Pendeteksi Objek Bersegi ini dinyatakan:

1. Layak digunakan tanpa revisi
2. Layak digunakan dengan revisi pada saran
3. Tidak layak digunakan.

Saran dan Perbaikan:

Tambahkan menu tujuan pembelajaran /diawal tampilan.

.....

.....

.....

.....

.....

Yogyakarta, 11 Desember 2018

Validator



(Andik Asmara, M.Pd)

NIP. 11510860908616

ANGKET PENILAIAN AHLI MEDIA

Dalam rangka penelitian Tugas Akhir Skripsi, saya mohon bantuan bapak / ibu / saudara untuk menjadi validator "**Pengembangan Aplikasi Pendeteksi Objek Bersegi Menggunakan Metode *Contour* Sebagai Media Pembelajaran Penginderaan Visual Robot**" agar layak digunakan sebagai media pembelajaran.

Mata Kuliah : Praktik Penginderaan Visual Robot
Pembuat : Ihsan Syarifuddin
Tanggal :

Prosedur Pengisian Instrumen Media Pembelajaran:

1. Dimohon kepada bapak / ibu / saudara untuk memberikan penilaian pada salah satu kolom di dalam angket ini berdasarkan kriteria berikut.
4 : Sangat Setuju
3 : Setuju
2 : Tidak Setuju
1 : Sangat Tidak Setuju
2. Dimohon untuk memberikan tanda *check* (✓) pada salah satu pilihan dalam setiap pernyataan sesuai pendapat bapak / ibu / saudara.
3. Jika terdapat kekurangan pada media pembelajaran ini, dimohon bapak / ibu / saudara memberikan saran dan masukan pada bagian yang telah disediakan.
4. Mohon untuk menuliskan nama sebagai validator (jika belum tersedia) pada bagian akhir lembar instrumen ini sebagai bukti bahwa instrumen ini dibuat sebenar-benarnya tanpa manipulasi data.
5. Atas bantuan bapak / ibu / saudara, kami mengucapkan terima kasih.

Pernyataan:

No.	Pernyataan	Pilihan Jawaban			
		4	3	2	1
1	Penggunaan media pembelajaran aplikasi pendeteksi objek bersegi dapat memberikan informasi mengenai tujuan pembelajaran penginderaan visual robot pada peserta didik.	✓			
2	Penggunaan media pembelajaran aplikasi pendeteksi objek bersegi dapat menjelaskan secara rinci mengenai tujuan belajar penginderaan visual robot pada peserta didik.	✓			
3	Penggunaan media pembelajaran membantu mempermudah kegiatan pembelajaran praktik penginderaan visual robot.	✓			
4	Media pembelajaran aplikasi pendeteksi objek bersegi dapat digunakan pengajar dalam menjelaskan materi agar menjadi lebih mudah difahami peserta didik.	✓			
5	Media pembelajaran aplikasi pendeteksi objek bersegi dapat meningkatkan keaktifan peserta didik dalam pembelajaran.	✓			
6	Penggunaan media pembelajaran aplikasi pendeteksi objek bersegi menumbuhkan semangat belajar peserta didik.	✓			
7	Media pembelajaran aplikasi pendeteksi objek bersegi memiliki keterkaitan dengan mata kuliah lain.		✓		
8	Media pembelajaran aplikasi pendeteksi objek bersegi dapat mendukung pembelajaran pada mata kuliah lain.		✓		
9	Tampilan GUI pada media pembelajaran berfungsi dengan baik.	✓			
10	Informasi pada tampilan GUI mudah difahami.	✓			
11	Bentuk media pembelajaran menarik.	✓			
12	Peletakan komponen pada media pembelajaran tersusun dengan baik.	✓			
13	Trackbar pada media pembelajaran aplikasi pendeteksi objek bersegi mudah digunakan.		✓		

14	Domain/Up/Down pada media pembelajaran mudah digunakan.		✓		
15	Sensor kamera yang digunakan pada media pembelajaran dapat bekerja dengan baik.	✓			
16	Semua komponen media pembelajaran berfungsi dengan baik.	✓			
17	Media pembelajaran aplikasi pendeteksi objek bersegi lebih menarik dengan adanya tampilan GUI.	✓			
18	Media pembelajaran dapat memberikan rasa ingin tahu kepada pengguna.	✓			
19	Media pembelajaran menarik untuk digunakan.	✓			
20	Media pembelajaran mudah digunakan.		✓		

Kesimpulan:

Menurut saya, Media Pembelajaran Aplikasi Pendeteksi Objek Bersegi ini dinyatakan:

1. Layak digunakan tanpa revisi
- ② Layak digunakan dengan revisi pada saran
3. Tidak layak digunakan.

Saran dan Perbaikan:

... langkah-langkah praktik pada labsheet perlu diperjelas.
 - Pada modul pendamping, tambahkan informasi aplikasi demo.
 - Perlu ada contoh image untuk keperluan uji pada labsheet Gaussian Blur.

Yogyakarta, 12 Desember 2018.

Validator



(Ariadie Chandra Nugraha, S.T., M.T)

NIP. 19770913 200501 1 002

Lampiran 4.2. Hasil Validasi Materi

ANGKET PENILAIAN AHLI MATERI

Dalam rangka penelitian tugas akhir skripsi, saya mohon bantuan bapak / ibu / saudara untuk menjadi validator **“Pengembangan Aplikasi Pendeteksi Objek Bersegi Menggunakan Metode *Contour* Sebagai Media Pembelajaran Penginderaan Visual Robot”** agar layak digunakan sebagai media pembelajaran.

Mata Kuliah : Praktik Penginderaan Visual Robot
Pembuat : Ihsan Syarifuddin
Tanggal :

Prosedur Pengisian Instrumen Materi Pembelajaran

1. Dimohon kepada bapak / ibu / saudara untuk memberikan penilaian pada salah satu kolom di dalam angket ini berdasarkan kriteria berikut:
4 : Sangat Setuju
3 : Setuju
2 : Tidak Setuju
1 : Sangat Tidak Setuju
2. Dimohon untuk memberikan tanda *check* (✓) pada salah satu pilihan dalam setiap pertanyaan sesuai pendapat bapak / ibu /saudara.
3. Jika terdapat kekurangan pada media pembelajaran ini, mohon bapak / ibu / saudara memberikan saran dan masukan pada bagian yang telah disediakan.
4. Mohon untuk menuliskan nama sebagai validator (jika belum tersedia) pada bagian akhir lembar instrumen ini sebagai bukti bahwa instrumen ini dibuat sebenar-benarnya tanpa manipulasi data.
5. Atas bantuan bapak / ibu / saudara, kami mengucapkan terima kasih.

Pernyataan:

No	Pernyataan	Pilihan Jawaban			
		4	3	2	1
1	Materi pelajaran yang diajarkan menggunakan media pembelajaran sesuai dengan silabus.	✓			
2	Materi yang diajarkan dalam media aplikasi pendeteksi objek bersegi relevan dengan kompetensi peserta didik pada bidang penginderaan visual robot.	✓			
3	Materi yang dicantumkan dalam media pembelajaran sudah lengkap.		✓		
4	Cara menghubungkan <i>library</i> Fmgucv pada media pembelajaran aplikasi pendeteksi objek bersegi dijelaskan di dalam modul.	✓			
5	Cara kerja deteksi objek bersegi dijelaskan di dalam modul.	✓			
6	Cara membuat program yang digunakan dalam pemrograman aplikasi pendeteksi objek bersegi dijelaskan di dalam modul.	✓			
7	Materi tentang dasar penginderaan visual robot disajikan dengan jelas.	✓			
8	Materi tentang deteksi <i>Contour</i> disajikan dengan jelas.	✓			
9	Alur pengoperasian media pembelajaran disajikan dengan jelas pada menu Bantuan.	✓			
10	Materi pelajaran lebih mudah dipahami setelah menggunakan media pembelajaran aplikasi pendeteksi objek bersegi.	✓			

Perlu
ditambahkan
Roi, Threshold,
hough, tracing
objec

11	Materi yang terdapat pada modul didukung dengan adanya media pembelajaran aplikasi pendeteksi objek bersegi.	✓			
12	Materi pelajaran sudah sesuai dengan tuntutan lapangan pekerjaan di bidang otomasi industri.	✓			
13	Materi disajikan secara runtut.		✓		
14	Materi yang disajikan didukung dengan dasar teori yang jelas.	✓			
15	Materi yang dibuat dapat meningkatkan motivasi belajar peserta didik.	✓			
16	Materi yang disajikan mendukung keterlibatan peserta didik dalam pembelajaran penginderaan visual robot.	✓			
17	Materi yang disajikan dapat dibaca dengan jelas dan sesuai dengan perkembangan intelektual peserta didik.	✓			
18	Materi yang disajikan dapat dibaca dengan jelas dan sesuai dengan perkembangan emosional peserta didik.	✓			
19	Tata bahasa pada modul sudah baku.	✓			
20	Istilah-istilah pada modul sudah baku.	✓			
21	Materi yang disajikan desain cover-nya menarik daya minat peserta didik sehingga tertarik dan betah untuk belajar.	✓			
22	Materi yang disajikan desain isi-nya menarik daya minat peserta didik sehingga tertarik dan betah untuk belajar.	✓			

Perlu ditinjau

23	Kualitas cetakan materi sesuai standar di pasaran buku.	✓			
24	Kualitas jilidan materi sesuai standar di pasaran buku.	✓			

Kesimpulan:

Menurut saya, materi untuk aplikasi pendeteksi objek bersegi dinyatakan:

1. Layak digunakan tanpa revisi
2. Layak digunakan dengan revisi pada saran
3. Tidak layak digunakan

Saran dan Perbaikan:

Tambahkan Modul
a) Threshold
b) Roi
c) histogram
d) Optical flow

Yogyakarta,

5/12/2018

Validator



(Moh. Khairudin, M.T., Ph.D)

NIP. 19790412 200212 1 002

ANGKET PENILAIAN AHLI MATERI

Dalam rangka penelitian tugas akhir skripsi, saya mohon bantuan bapak / ibu / saudara untuk menjadi validator **"Pengembangan Aplikasi Pendeteksi Objek Bersegi Menggunakan Metode *Contour* Sebagai Media Pembelajaran Penginderaan Visual Robot"** agar layak digunakan sebagai media pembelajaran.

Mata Kuliah : Praktik Penginderaan Visual Robot
Pembuat : Ihsan Syarifuddin
Tanggal :

Prosedur Pengisian Instrumen Materi Pembelajaran

1. Dimohon kepada bapak / ibu / saudara untuk memberikan penilaian pada salah satu kolom di dalam angket ini berdasarkan kriteria berikut:
4 : Sangat Setuju
3 : Setuju
2 : Tidak Setuju
1 : Sangat Tidak Setuju
2. Dimohon untuk memberikan tanda *check* (✓) pada salah satu pilihan dalam setiap pertanyaan sesuai pendapat bapak / ibu /saudara.
3. Jika terdapat kekurangan pada media pembelajaran ini, mohon bapak / ibu / saudara memberikan saran dan masukan pada bagian yang telah disediakan.
4. Mohon untuk menuliskan nama sebagai validator (jika belum tersedia) pada bagian akhir lembar instrumen ini sebagai bukti bahwa instrumen ini dibuat sebenar-benarnya tanpa manipulasi data.
5. Atas bantuan bapak / ibu / saudara, kami mengucapkan terima kasih.

Pernyataan:

No	Pernyataan	Pilihan Jawaban			
		4	3	2	1
1	Materi pelajaran yang diajarkan menggunakan media pembelajaran sesuai dengan silabus.	✓			
2	Materi yang diajarkan dalam media aplikasi pendeteksi objek bersegi relevan dengan kompetensi peserta didik pada bidang penginderaan visual robot.	✓			
3	Materi yang dicantumkan dalam media pembelajaran sudah lengkap.		✓		
4	Cara menghubungkan <i>library</i> EmguCV pada media pembelajaran aplikasi pendeteksi objek bersegi dijelaskan di dalam modul.	✓			
5	Cara kerja deteksi objek bersegi dijelaskan di dalam modul.		✓		
6	Cara membuat program yang digunakan dalam pemrograman aplikasi pendeteksi objek bersegi dijelaskan di dalam modul.	✓			
7	Materi tentang dasar penginderaan visual robot disajikan dengan jelas.	✓			
8	Materi tentang deteksi <i>Contour</i> disajikan dengan jelas.	✓			
9	Alur pengoperasian media pembelajaran disajikan dengan jelas pada menu Bantuan.		✓		
10	Materi pelajaran lebih mudah dipahami setelah menggunakan media pembelajaran aplikasi pendeteksi objek bersegi.	✓			

11	Materi yang terdapat pada modul didukung dengan adanya media pembelajaran aplikasi pendeteksi objek bersegi.	✓			
12	Materi pelajaran sudah sesuai dengan tuntutan lapangan pekerjaan di bidang otomasi industri.	✓			
13	Materi disajikan secara runtut.	✓			
14	Materi yang disajikan didukung dengan dasar teori yang jelas.		✓		
15	Materi yang dibuat dapat meningkatkan motivasi belajar peserta didik.		✓		
16	Materi yang disajikan mendukung keterlibatan peserta didik dalam pembelajaran penginderaan visual robot.	✓			
17	Materi yang disajikan dapat dibaca dengan jelas dan sesuai dengan perkembangan intelektual peserta didik.	✓			
18	Materi yang disajikan dapat dibaca dengan jelas dan sesuai dengan perkembangan emosional peserta didik.		✓		
19	Tata bahasa pada modul sudah baku.		✓		
20	Istilah-istilah pada modul sudah baku.	✓			
21	Materi yang disajikan desain <i>cover</i> -nya menarik daya minat peserta didik sehingga tertarik dan betah untuk belajar.		✓		
22	Materi yang disajikan desain isi-nya menarik daya minat peserta didik sehingga tertarik dan betah untuk belajar.	✓			

23	Kualitas cetakan materi sesuai standar di pasaran buku.	✓			
24	Kualitas jilidan materi sesuai standar di pasaran buku.	✓			

Kesimpulan:

Menurut saya, materi untuk aplikasi pendeteksi objek bersegi dinyatakan:

1. Layak digunakan tanpa revisi
- ② Layak digunakan dengan revisi pada saran
3. Tidak layak digunakan

Saran dan Perbaikan:

- * Nama dari masing-masing objek lebih baik ditampilkan pada antarmuka Form.
- * Informasi tambahan akan mengisi tabel penyajian akan dilengkapi
misal kode objek no 6 akan cara akan mendapatkan data kemiringan objek
dikembangkan dengan cara bagaimana dan.

Yogyakarta,

Validator


(Sigit Yarmo, S.T., M.T.)

NIP. 19730125 199903 1 001

Lampiran 5 Hasil Penelitian

Lampiran 5.1. Hasil Uji Pengguna

Lampiran 5.1. Hasil Uji Pengguna

ANGKET PENILAIAN
PENGEMBANGAN APLIKASI PENDETEKSI OBJEK BERSEGI
MENGUNAKAN METODE *CONTOUR*
SEBAGAI MEDIA PEMBELAJARAN
PRAKTIK PENGINDERAAN VISUAL ROBOT



IDENTITAS PESERTA DIDIK

Nama : Angela M
NIM : 16510241001

PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2018

Angket Penilaian Media

Hal : Pengisian Angket Penilaian
Kepada : Peserta didik Program Studi Pendidikan Teknik Mekatronika

Assalamu'alaikum Wr.Wb.

Dengan hormat,

Mohon kesediaan dan bantuan saudara/i untuk meluangkan waktu guna mengisi angket ini. Angket ini berguna untuk mengumpulkan data terkait dengan "Pengembangan Aplikasi Pendeteksi Objek Bersegi Menggunakan Metode *Contour* Sebagai Media Pembelajaran Praktik Penginderaan Visual Robot".

Angket ini bukan merupakan tes, sehingga jawaban yang anda berikan tidak akan memengaruhi nilai mata kuliah. Jawaban yang baik adalah jawaban yang sesuai dengan kenyataan dan diisi berdasarkan hati nurani saudara/i, serta akan kami jamin kerahasiaannya. Kejujuran saudara/i dalam menjawab angket ini sangat diharapkan demi mendapatkan hasil penelitian yang maksimal.

Atas bantuan dan kerjasama dari saudara/i, saya ucapkan terimakasih.
Wassalamu'alaikum Wr.Wb.

Yogyakarta, 12 Oktober 2018

Hormat saya,

Peneliti

A. **Petunjuk Pengisian Angket**

1. Angket ini dimaksudkan untuk mengetahui pendapat/penilaian anda sebagai pengguna media pembelajaran APLIKASI PENDETEKSI OBJEK BERSEGI.
2. Anda diharapkan memilih salah satu pilihan jawaban pada setiap pernyataan yang tersedia dengan memberikan TANDA SILANG (X) pada kolom jawaban.

NO	PERNYATAAN	JAWABAN			
1	Desain tata letak komponen pada GUI sudah rapi.	1	2	X	4

3. Jika anda ingin mengubah jawaban, maka berikan tanda SAMA DENGAN (=) pada pilihan jawaban yang akan diganti dan memberikan TANDA SILANG (X) pada kolom penggantinya.

NO	PERNYATAAN	JAWABAN			
1	Desain tata letak komponen pada GUI sudah rapi.	1	=	X	4

4. Keterangan jawaban:

- 1 = Sangat Tidak Setuju / Sangat Tidak Sesuai / Sangat Tidak Baik
 2 = Tidak Setuju / Tidak Sesuai / Tidak Baik
 3 = Setuju / Sesuai / Baik
 4 = Sangat Setuju / Sangat Sesuai / Sangat Baik

5. Komentar atau saran anda mohon ditulis pada lembar yang telah disediakan. Atas kesediaan anda untuk mengisi angket ini, saya ucapkan terima kasih

B. Angket Penilaian

NO	PERNYATAAN	JAWABAN			
		1	2	3	4
1	Media pembelajaran yang dibuat sesuai dengan materi pembelajaran yang diajarkan.	1	2	3	X
2	Materi yang ada dalam modul mudah untuk dipahami.	1	2	X	4
3	Contoh program pada modul mudah untuk dipahami.	1	2	X	4
4	Materi yang disajikan sesuai dengan mata kuliah praktik penginderaan visual robot.	1	2	3	X
5	Materi yang disajikan berisi kompetensi yang dibutuhkan.	1	2	3	X
6	Ilustrasi dalam modul mempermudah dalam melaksanakan kegiatan praktikum.	1	2	3	X
7	Langkah kerja dalam modul mudah untuk diikuti.	1	2	X	4
8	Komponen pada GUI tidak membuat bingung.	1	2	X	4
9	Pengoperasian media pembelajaran aplikasi pendeteksi objek bersegi dapat dilakukan dengan mudah.	1	2	X	4
10	GUI sebagai hasil <i>output</i> data berfungsi dengan baik dan akurat.	1	2	3	X
11	Desain media pembelajaran menarik.	1	2	3	X
12	Media pembelajaran aplikasi pendeteksi objek bersegi memberi tambahan pengetahuan dalam pemrograman pada mata kuliah praktik penginderaan visual robot.	1	2	3	X
13	Media pembelajaran aplikasi pendeteksi objek bersegi memberi tambahan pengetahuan tentang penginderaan visual robot yang lebih kompleks.	1	2	X	4
14	Media pembelajaran aplikasi pendeteksi objek bersegi dapat membantu dalam memahami materi pada mata kuliah lain.	1	2	X	4
15	Media pembelajaran aplikasi pendeteksi objek bersegi membantu mempelajari pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	3	X

16	Media pembelajaran aplikasi pendeteksi objek bersegi dapat memotivasi untuk belajar pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	3	X
17	Media pembelajaran aplikasi pendeteksi objek bersegi menambah kompetensi dalam pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	X	4
18	Media pembelajaran aplikasi pendeteksi objek bersegi dapat meningkatkan keaktifan belajar saat pembelajaran praktik penginderaan visual robot.	1	2	X	4
19	Media pembelajaran aplikasi pendeteksi objek bersegi efektif dalam meningkatkan kemampuan belajar peserta didik praktik penginderaan visual robot.	1	2	3	X
20	Media pembelajaran aplikasi pendeteksi objek bersegi secara efisien dapat meningkatkan kemampuan belajar peserta didik.	1	2	X	4

C. Komentor dan Saran Umum

Warna tampilan terlalu cerah, Saran diganti black pink

Yogyakarta, 12 Desember 2018



(Anggela Moers L. 2)

ANGKET PENILAIAN
PENGEMBANGAN APLIKASI PENDETEKSI OBJEK BERSEGI
MENGUNAKAN METODE *CONTOUR*
SEBAGAI MEDIA PEMBELAJARAN
PRAKTIK PENGINDERAAN VISUAL ROBOT



IDENTITAS PESERTA DIDIK

Nama : Devj Larasati
NIM : 16518241025

PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2018

Angket Penilaian Media

Hal : Pengisian Angket Penilaian
Kepada : Peserta didik Program Studi Pendidikan Teknik Mekatronika

Assalamu'alaikum Wr. Wb.

Dengan hormat,

Mohon kesediaan dan bantuan saudara/i untuk meluangkan waktu guna mengisi angket ini. Angket ini berguna untuk mengumpulkan data terkait dengan "Pengembangan Aplikasi Pendeteksi Objek Bersegi Menggunakan Metode *Contour* Sebagai Media Pembelajaran Praktik Penginderaan Visual Robot".

Angket ini bukan merupakan tes, sehingga jawaban yang anda berikan tidak akan memengaruhi nilai mata kuliah. Jawaban yang baik adalah jawaban yang sesuai dengan kenyataan dan diisi berdasarkan hati nurani saudara/i, serta akan kami jamin kerahasiaannya. Kejujuran saudara/i dalam menjawab angket ini sangat diharapkan demi mendapatkan hasil penelitian yang maksimal.

Atas bantuan dan kerjasama dari saudara/i, saya ucapkan terimakasih.

Wassalamu'alaikum Wr. Wb.

Yogyakarta, 12 Oktober 2018

Hormat saya,

Peneliti

A. Petunjuk Pengisian Angket

1. Angket ini dimaksudkan untuk mengetahui pendapat/penilaian anda sebagai pengguna media pembelajaran APLIKASI PENDETEKSI OBJEK BERSEGI.
2. Anda diharapkan memilih salah satu pilihan jawaban pada setiap pernyataan yang tersedia dengan memberikan TANDA SILANG (X) pada kolom jawaban.

NO	PERNYATAAN	JAWABAN			
1	Desain tata letak komponen pada GUI sudah rapi.	1	2	X	4

3. Jika anda ingin mengubah jawaban, maka berikan tanda SAMA DENGAN (=) pada pilihan jawaban yang akan diganti dan memberikan TANDA SILANG (X) pada kolom penggantinya.

NO	PERNYATAAN	JAWABAN			
1	Desain tata letak komponen pada GUI sudah rapi.	1	=	X	4

4. Keterangan jawaban:

- 1 = Sangat Tidak Setuju / Sangat Tidak Sesuai / Sangat Tidak Baik
2 = Tidak Setuju / Tidak Sesuai / Tidak Baik
3 = Setuju / Sesuai / Baik
4 = Sangat Setuju / Sangat Sesuai / Sangat Baik

5. Komentar atau saran anda mohon ditulis pada lembar yang telah disediakan. Atas kesediaan anda untuk mengisi angket ini, saya ucapkan terima kasih

B. Angket Penilaian

NO	PERNYATAAN	JAWABAN			
1	Media pembelajaran yang dibuat sesuai dengan materi pembelajaran yang diajarkan.	1	2	3	☒
2	Materi yang ada dalam modul mudah untuk dipahami.	1	2	3	☒
3	Contoh program pada modul mudah untuk dipahami.	1	2	3	☒
4	Materi yang disajikan sesuai dengan mata kuliah praktik penginderaan visual robot.	1	2	3	☒
5	Materi yang disajikan berisi kompetensi yang dibutuhkan.	1	2	☒	4
6	Ilustrasi dalam modul mempermudah dalam melaksanakan kegiatan praktikum.	1	2	3	☒
7	Langkah kerja dalam modul mudah untuk diikuti.	1	2	☒	4
8	Komponen pada GUI tidak membuat bingung.	1	2	☒	4
9	Pengoperasian media pembelajaran aplikasi pendeteksi objek bersegi dapat dilakukan dengan mudah.	1	2	☒	4
10	GUI sebagai hasil <i>output</i> data berfungsi dengan baik dan akurat.	1	2	3	☒
11	Desain media pembelajaran menarik.	1	2	☒	4
12	Media pembelajaran aplikasi pendeteksi objek bersegi memberi tambahan pengetahuan dalam pemrograman pada mata kuliah praktik penginderaan visual robot.	1	2	3	☒
13	Media pembelajaran aplikasi pendeteksi objek bersegi memberi tambahan pengetahuan tentang penginderaan visual robot yang lebih kompleks.	1	2	3	☒
14	Media pembelajaran aplikasi pendeteksi objek bersegi dapat membantu dalam memahami materi pada mata kuliah lain.	1	2	3	☒
15	Media pembelajaran aplikasi pendeteksi objek bersegi membantu mempelajari pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	☒	4

16	Media pembelajaran aplikasi pendeteksi objek bersegi dapat memotivasi untuk belajar pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	X	4
17	Media pembelajaran aplikasi pendeteksi objek bersegi menambah kompetensi dalam pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	3	X
18	Media pembelajaran aplikasi pendeteksi objek bersegi dapat meningkatkan keaktifan belajar saat pembelajaran praktik penginderaan visual robot.	1	2	X	4
19	Media pembelajaran aplikasi pendeteksi objek bersegi efektif dalam meningkatkan kemampuan belajar peserta didik praktik penginderaan visual robot.	1	2	X	4
20	Media pembelajaran aplikasi pendeteksi objek bersegi secara efisien dapat meningkatkan kemampuan belajar peserta didik.	1	2	X	4

C. Komentar dan Saran Umum

Secara keseluruhan media tersebut sudah bagus dan sangat membantu kami/saya dalam belajar visual studio. Namun untuk lebih diperhatikan dalam pemilihan tampilan warnanya agar lebih menarik lagi untuk dipelajari.

Yogyakarta, 10 Desember 2018



(.....Devy Larasati.....)

ANGKET PENILAIAN
PENGEMBANGAN APLIKASI PENDETEKSI OBJEK BERSEGI
MENGUNAKAN METODE *CONTOUR*
SEBAGAI MEDIA PEMBELAJARAN
PRAKTIK PENGINDERAAN VISUAL ROBOT



IDENTITAS PESERTA DIDIK

Nama : Inggri Putriana
NIM : 16518241020

PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2018

Angket Penilaian Media

Hal : Pengisian Angket Penilaian

Kepada : Peserta didik Program Studi Pendidikan Teknik Mekatronika

Assalamu'alaikum Wr.Wb.

Dengan hormat,

Mohon kesediaan dan bantuan saudara/i untuk meluangkan waktu guna mengisi angket ini. Angket ini berguna untuk mengumpulkan data terkait dengan "Pengembangan Aplikasi Pendeteksi Objek Bersegi Menggunakan Metode *Contour* Sebagai Media Pembelajaran Praktik Penginderaan Visual Robot".

Angket ini bukan merupakan tes, sehingga jawaban yang anda berikan tidak akan memengaruhi nilai mata kuliah. Jawaban yang baik adalah jawaban yang sesuai dengan kenyataan dan diisi berdasarkan hati nurani saudara/i, serta akan kami jamin kerahasiaannya. Kejujuran saudara/i dalam menjawab angket ini sangat diharapkan demi mendapatkan hasil penelitian yang maksimal.

Atas bantuan dan kerjasama dari saudara/i, saya ucapkan terimakasih.

Wassalamu'alaikum Wr.Wb.

Yogyakarta, 12 Oktober 2018

Hormat saya,

Peneliti

A. Petunjuk Pengisian Angket

1. Angket ini dimaksudkan untuk mengetahui pendapat/penilaian anda sebagai pengguna media pembelajaran APLIKASI PENDEKTEKST OBJEK BERSEGI.
2. Anda diharapkan memilih salah satu pilihan jawaban pada setiap pernyataan yang tersedia dengan memberikan TANDA SILANG (X) pada kolom jawaban.

NO	PERNYATAAN	JAWABAN			
1	Desain tata letak komponen pada GUI sudah rapi.	1	2	X	4

3. Jika anda ingin mengubah jawaban, maka berikan tanda SAMA DENGAN (=) pada pilihan jawaban yang akan diganti dan memberikan TANDA SILANG (X) pada kolom penggantinya.

NO	PERNYATAAN	JAWABAN			
1	Desain tata letak komponen pada GUI sudah rapi.	1	=	X	4

4. Keterangan jawaban:

- 1 = Sangat Tidak Setuju / Sangat Tidak Sesuai / Sangat Tidak Baik
2 = Tidak Setuju / Tidak Sesuai / Tidak Baik
3 = Setuju / Sesuai / Baik
4 = Sangat Setuju / Sangat Sesuai / Sangat Baik

5. Komentar atau saran anda mohon ditulis pada lembar yang telah disediakan. Atas kesediaan anda untuk mengisi angket ini, saya ucapkan terima kasih

B. Angket Penilaian

NO	PERNYATAAN	JAWABAN			
1	Media pembelajaran yang dibuat sesuai dengan materi pembelajaran yang diajarkan.	☒	2	3	☒
2	Materi yang ada dalam modul mudah untuk dipahami.	1	2	☒	4
3	Contoh program pada modul mudah untuk dipahami.	1	2	☒	4
4	Materi yang disajikan sesuai dengan mata kuliah praktik penginderaan visual robot.	1	2	3	☒
5	Materi yang disajikan berisi kompetensi yang dibutuhkan.	1	2	3	☒
6	Ilustrasi dalam modul mempermudah dalam melaksanakan kegiatan praktikum.	1	2	☒	4
7	Langkah kerja dalam modul mudah untuk diikuti.	1	2	☒	4
8	Komponen pada GUI tidak membuat bingung.	1	2	3	☒
9	Pengoperasian media pembelajaran aplikasi pendeteksi objek bersegi dapat dilakukan dengan mudah.	1	2	☒	4
10	GUI sebagai hasil <i>output</i> data berfungsi dengan baik dan akurat.	1	2	☒	4
11	Desain media pembelajaran menarik.	1	2	3	☒
12	Media pembelajaran aplikasi pendeteksi objek bersegi memberi tambahan pengetahuan dalam pemrograman pada mata kuliah praktik penginderaan visual robot.	1	2	3	☒
13	Media pembelajaran aplikasi pendeteksi objek bersegi memberi tambahan pengetahuan tentang penginderaan visual robot yang lebih kompleks.	1	2	3	☒
14	Media pembelajaran aplikasi pendeteksi objek bersegi dapat membantu dalam memahami materi pada mata kuliah lain.	1	2	☒	4
15	Media pembelajaran aplikasi pendeteksi objek bersegi membantu mempelajari pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	☒	4

16	Media pembelajaran aplikasi pendeteksi objek bersegi dapat memotivasi untuk belajar pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	3	X
17	Media pembelajaran aplikasi pendeteksi objek bersegi menambah kompetensi dalam pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	3	X
18	Media pembelajaran aplikasi pendeteksi objek bersegi dapat meningkatkan keaktifan belajar saat pembelajaran praktik penginderaan visual robot.	1	2	3	X
19	Media pembelajaran aplikasi pendeteksi objek bersegi efektif dalam meningkatkan kemampuan belajar peserta didik praktik penginderaan visual robot.	1	2	3	X
20	Media pembelajaran aplikasi pendeteksi objek bersegi secara efisien dapat meningkatkan kemampuan belajar peserta didik.	1	2	3	X

C. **Komentar dan Saran Umum**

Saran ya mas, jangan pernah berhenti menjadi tenaga pendidik, terimakasih untuk ilmunya semoga sukses dimasa depan, semangat.

Yogyakarta, 12 Desember 2018



(Inga Putriana)

ANGKET PENILAIAN
PENGEMBANGAN APLIKASI PENDETEKSI OBJEK BERSEGI
MENGUNAKAN METODE *CONTOUR*
SEBAGAI MEDIA PEMBELAJARAN
PRAKTIK PENGINDERAAN VISUAL ROBOT



IDENTITAS PESERTA DIDIK

Nama : Putri Rahmawati
NIM : 16518241006

PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2018

Angket Penilaian Media

Hal : Pengisian Angket Penilaian
Kepada : Peserta didik Program Studi Pendidikan Teknik Mekatronika

Assalamu'alaikum Wr.Wb.

Dengan hormat,

Mohon kesediaan dan bantuan saudara/i untuk meluangkan waktu guna mengisi angket ini. Angket ini berguna untuk mengumpulkan data terkait dengan "Pengembangan Aplikasi Pendeteksi Objek Bersegi Menggunakan Metode *Contour* Sebagai Media Pembelajaran Praktik Penginderaan Visual Robot".

Angket ini bukan merupakan tes, sehingga jawaban yang anda berikan tidak akan memengaruhi nilai mata kuliah. Jawaban yang baik adalah jawaban yang sesuai dengan kenyataan dan diisi berdasarkan hati nurani saudara/i, serta akan kami jamin kerahasiaannya. Kejujuran saudara/i dalam menjawab angket ini sangat diharapkan demi mendapatkan hasil penelitian yang maksimal.

Atas bantuan dan kerjasama dari saudara/i, saya ucapkan terimakasih.

Wassalamu'alaikum Wr.Wb.

Yogyakarta, 12 Oktober 2018

Hormat saya,

Peneliti

A. Petunjuk Pengisian Angket

1. Angket ini dimaksudkan untuk mengetahui pendapat/penilaian anda sebagai pengguna media pembelajaran APLIKASI PENDETEKSI OBJEK BERSEGL.
2. Anda diharapkan memilih salah satu pilihan jawaban pada setiap pernyataan yang tersedia dengan memberikan TANDA SILANG (X) pada kolom jawaban.

NO	PERNYATAAN	JAWABAN			
1	Desain tata letak komponen pada GUI sudah rapi.	1	2	X	4

3. Jika anda ingin mengubah jawaban, maka berikan tanda SAMA DENGAN (=) pada pilihan jawaban yang akan diganti dan memberikan TANDA SILANG (X) pada kolom penggantinya.

NO	PERNYATAAN	JAWABAN			
1	Desain tata letak komponen pada GUI sudah rapi.	1	=	X	4

4. Keterangan jawaban:

- 1 = Sangat Tidak Setuju / Sangat Tidak Sesuai / Sangat Tidak Baik
 2 = Tidak Setuju / Tidak Sesuai / Tidak Baik
 3 = Setuju / Sesuai / Baik
 4 = Sangat Setuju / Sangat Sesuai / Sangat Baik

5. Komentar atau saran anda mohon ditulis pada lembar yang telah disediakan. Atas kesediaan anda untuk mengisi angket ini, saya ucapkan terima kasih

B. Angket Penilaian

NO	PERNYATAAN	JAWABAN			
1	Media pembelajaran yang dibuat sesuai dengan materi pembelajaran yang diajarkan.	1	2	3	☒
2	Materi yang ada dalam modul mudah untuk dipahami.	1	2	☒	4
3	Contoh program pada modul mudah untuk dipahami.	1	2	☒	4
4	Materi yang disajikan sesuai dengan mata kuliah praktik penginderaan visual robot.	1	2	3	☒
5	Materi yang disajikan berisi kompetensi yang dibutuhkan.	1	2	☒	4
6	Ilustrasi dalam modul mempermudah dalam melaksanakan kegiatan praktikum.	1	2	☒	4
7	Langkah kerja dalam modul mudah untuk diikuti.	1	2	☒	4
8	Komponen pada GUI tidak membuat bingung.	1	2	3	☒
9	Pengoperasian media pembelajaran aplikasi pendeteksi objek bersegi dapat dilakukan dengan mudah.	1	2	☒	4
10	GUI sebagai hasil <i>output</i> data berfungsi dengan baik dan akurat.	1	2	☒	4
11	Desain media pembelajaran menarik.	1	☒	3	4
12	Media pembelajaran aplikasi pendeteksi objek bersegi memberi tambahan pengetahuan dalam pemrograman pada mata kuliah praktik penginderaan visual robot.	1	2	3	☒
13	Media pembelajaran aplikasi pendeteksi objek bersegi memberi tambahan pengetahuan tentang penginderaan visual robot yang lebih kompleks.	1	2	3	☒
14	Media pembelajaran aplikasi pendeteksi objek bersegi dapat membantu dalam memahami materi pada mata kuliah lain.	1	2	☒	4
15	Media pembelajaran aplikasi pendeteksi objek bersegi membantu mempelajari pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	☒	4

16	Media pembelajaran aplikasi pendeteksi objek bersegi dapat memotivasi untuk belajar pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	3	4
17	Media pembelajaran aplikasi pendeteksi objek bersegi menambah kompetensi dalam pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	3	4
18	Media pembelajaran aplikasi pendeteksi objek bersegi dapat meningkatkan keaktifan belajar saat pembelajaran praktik penginderaan visual robot.	1	2	3	4
19	Media pembelajaran aplikasi pendeteksi objek bersegi efektif dalam meningkatkan kemampuan belajar peserta didik praktik penginderaan visual robot.	1	2	3	4
20	Media pembelajaran aplikasi pendeteksi objek bersegi secara efisien dapat meningkatkan kemampuan belajar peserta didik.	1	2	3	4

atau tidak hanya
4 menggunakan warna
oren saja

C. Komentar dan Saran Umum

- untuk media pembelajaran lebih bisa menggunakan banyak warna/supaya lebih menarik, karena isi dan konten materinya sudah sangat bagus
- untuk tulisan pada media pembelajaran lebih disesuaikan, tidak terlalu kecil

Yogyakarta, 12-12-2018.



Ruti Rammawati

(.....)

ANGKET PENILAIAN
PENGEMBANGAN APLIKASI PENDETEKSI OBJEK BERSEGI
MENGUNAKAN METODE *CONTOUR*
SEBAGAI MEDIA PEMBELAJARAN
PRAKTIK PENGINDERAAN VISUAL ROBOT



IDENTITAS PESERTA DIDIK

Nama : I Made Kana Panyahsa
NIM : 16518291009

PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2018

Angket Penilaian Media

Hal : Pengisian Angket Penilaian
Kepada : Peserta didik Program Studi Pendidikan Teknik Mekatronika

Assalamu'alaikum Wr.Wb.

Dengan hormat,

Mohon kesediaan dan bantuan saudara/i untuk meluangkan waktu guna mengisi angket ini. Angket ini berguna untuk mengumpulkan data terkait dengan "Pengembangan Aplikasi Pendeteksi Objek Bersegi Menggunakan Metode *Contour* Sebagai Media Pembelajaran Praktik Penginderaan Visual Robot".

Angket ini bukan merupakan tes, sehingga jawaban yang anda berikan tidak akan memengaruhi nilai mata kuliah. Jawaban yang baik adalah jawaban yang sesuai dengan kenyataan dan diisi berdasarkan hati nurani saudara/i, serta akan kami jamin kerahasiaannya. Kejujuran saudara/i dalam menjawab angket ini sangat diharapkan demi mendapatkan hasil penelitian yang maksimal.

Atas bantuan dan kerjasama dari saudara/i, saya ucapkan terimakasih.

Wassalamu'alaikum Wr.Wb.

Yogyakarta, 12 Oktober 2018

Hormat saya,

Peneliti

A. Petunjuk Pengisian Angket

1. Angket ini dimaksudkan untuk mengetahui pendapat/penilaian anda sebagai pengguna media pembelajaran APLIKASI PENDETEKSI OBJEK BERSEGI.
2. Anda diharapkan memilih salah satu pilihan jawaban pada setiap pernyataan yang tersedia dengan memberikan TANDA SILANG (X) pada kolom jawaban.

NO	PERNYATAAN	JAWABAN			
1	Desain tata letak komponen pada GUI sudah rapi.	1	2	X	4

3. Jika anda ingin mengubah jawaban, maka berikan tanda SAMA DENGAN (=) pada pilihan jawaban yang akan diganti dan memberikan TANDA SILANG (X) pada kolom penggantinya.

NO	PERNYATAAN	JAWABAN			
1	Desain tata letak komponen pada GUI sudah rapi.	1	=	X	4

4. Keterangan jawaban:

- 1 = Sangat Tidak Setuju / Sangat Tidak Sesuai / Sangat Tidak Baik
2 = Tidak Setuju / Tidak Sesuai / Tidak Baik
3 = Setuju / Sesuai / Baik
4 = Sangat Setuju / Sangat Sesuai / Sangat Baik

5. Komentar atau saran anda mohon ditulis pada lembar yang telah disediakan. Atas kesediaan anda untuk mengisi angket ini, saya ucapkan terima kasih

B. Angket Penilaian

NO	PERNYATAAN	JAWABAN			
		1	2	3	4
1	Media pembelajaran yang dibuat sesuai dengan materi pembelajaran yang diajarkan.	1	2	3	X
2	Materi yang ada dalam modul mudah untuk dipahami.	1	2	3	X
3	Contoh program pada modul mudah untuk dipahami.	1	2	3	X
4	Materi yang disajikan sesuai dengan mata kuliah praktik penginderaan visual robot.	1	2	3	X
5	Materi yang disajikan berisi kompetensi yang dibutuhkan.	1	2	3	X
6	Ilustrasi dalam modul mempermudah dalam melaksanakan kegiatan praktikum.	1	2	3	X
7	Langkah kerja dalam modul mudah untuk diikuti.	1	2	3	X
8	Komponen pada GUI tidak membuat bingung.	1	2	3	X
9	Pengoperasian media pembelajaran aplikasi pendeteksi objek bersegi dapat dilakukan dengan mudah.	1	2	3	X
10	GUI sebagai hasil <i>output</i> data berfungsi dengan baik dan akurat.	1	2	3	X
11	Desain media pembelajaran menarik.	1	X	3	4
12	Media pembelajaran aplikasi pendeteksi objek bersegi memberi tambahan pengetahuan dalam pemrograman pada mata kuliah praktik penginderaan visual robot.	1	2	3	X
13	Media pembelajaran aplikasi pendeteksi objek bersegi memberi tambahan pengetahuan tentang penginderaan visual robot yang lebih kompleks.	1	2	3	X
14	Media pembelajaran aplikasi pendeteksi objek bersegi dapat membantu dalam memahami materi pada mata kuliah lain.	1	2	X	4
15	Media pembelajaran aplikasi pendeteksi objek bersegi membantu mempelajari pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	X	4

16	Media pembelajaran aplikasi pendeteksi objek bersegi dapat memotivasi untuk belajar pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	3	X
17	Media pembelajaran aplikasi pendeteksi objek bersegi menambah kompetensi dalam pemrograman pada pembelajaran praktik penginderaan visual robot.	1	2	X	4
18	Media pembelajaran aplikasi pendeteksi objek bersegi dapat meningkatkan keaktifan belajar saat pembelajaran praktik penginderaan visual robot.	1	2	3	X
19	Media pembelajaran aplikasi pendeteksi objek bersegi efektif dalam meningkatkan kemampuan belajar peserta didik praktik penginderaan visual robot.	1	2	3	X
20	Media pembelajaran aplikasi pendeteksi objek bersegi secara efisien dapat meningkatkan kemampuan belajar peserta didik.	1	2	3	X

C. Komentor dan Saran Umum

.....

Yogyakarta, 12 Desember 2018.



(I Made Yana Pangkajene.....)

Lampiran 6 Analisis Data

Lampiran 6.1. Analisis Data Ahli Media

Lampiran 6.2. Analisis Data Ahli Materi

Lampiran 6.3. Analisis Data Uji Pengguna

Lampiran 6.4. Analisis Data Realibilitas Instrumen

Lampiran 6.1. Analisis Data Ahli Media

Ahli Media	Konsensus dengan																							
	Tujuan				Jml	Klgr	Karakteristik				Jml	Klgr	Materi				Jml	Klgr	Teori				Jml	Klgr
	1	2	3	4			5	6	7	8			9	10	11	12			13	14	15	16		
Asdik Aznura, M.Pd	2	2	3	3	10	CL	3	3	3	3	12	L	3	3	3	3	12	L	3	3	3	3	12	L
Artadila Chandra Nugraha, S.T., M.T.	4	4	4	4	16	SL	4	4	3	3	14	SL	4	4	4	4	16	SL	3	3	4	4	14	SL
Jumlah					26						26						26						26	
Rerata					13	L					13	L					14	SL					13	L
Persentase					81,25%						81,25%						87,50%						81,25%	
Jumlah Butir					4						4						4						4	
Skor Maks					16						16						16						16	
Skor Min					4						4						4						4	
Berada Ideal					10						10						10						10	
Simpangan Ideal					2						2						2						2	

Kategori Kelengkapan	Interval Konsensus dengan																								Keseluruhan	Ket					
	Tujuan				Karakteristik				Materi				Teori				Gaya Belajar				Kondisi										
Sangat Layak	13,6	S	X	S	10	13,6	S	X	S	16	13,6	S	X	S	16	13,6	S	X	S	8	6,8	S	X	S	8	68	S	X	S	80	36
Layak	11,2	S	X	S	13,6	11,2	S	X	S	13,6	11,2	S	X	S	13,6	11,2	S	X	S	6,8	5,6	S	X	S	6,8	56	S	X	S	64	32
Cukup Layak	8,8	S	X	S	11,2	8,8	S	X	S	11,2	8,8	S	X	S	11,2	8,8	S	X	S	4,4	4,4	S	X	S	4,4	44	S	X	S	48	24
Kurang Layak	6,4	S	X	S	8,8	6,4	S	X	S	8,8	6,4	S	X	S	8,8	6,4	S	X	S	2,2	2,2	S	X	S	2,2	22	S	X	S	24	12
Tidak Layak	4	S	X	S	6,4	4	S	X	S	6,4	4	S	X	S	6,4	4	S	X	S	2,2	2	S	X	S	2,2	20	S	X	S	20	10

Lampiran 6.2. Analisis Data Ahli Materi

Atrib	Relevansi												Kriteria				Bahasa				Geografikan				Jml	Kgr								
	1	2	3	4	5	6	7	8	9	10	11	12	Jml	Kgr	13	14	15	16	Jml	Kgr	17	18	19	20			Jml	Kgr	21	22	23	24	Jml	Kgr
Moh. Khairuddin M.T., Ph.D.	4	4	3	3	4	4	4	4	4	4	4	4	47	SL	3	3	3	3	4	SL	4	3	3	3	16	SL	4	4	4	4	16	SL	54	SL
Sigit Yatsmos, S.T., M.T.	4	4	3	4	3	4	4	4	3	4	4	4	45	SL	4	3	3	4	14	SL	4	3	3	3	14	SL	3	4	4	4	15	SL	68	SL
Jumlah													92						29						30						31		181	
Rata-rata													46	SL					14,5	SL					15	SL					15,5	SL	71	SL
Persentase													95,83%						90,63%						93,75%							96,88%		94,79%
Jumlah Butir													12																		4		24	
Skor Maks													48						16						16						16		56	
Skor Min													4						4						4						4		24	
Rerata Ideal													10						10						10						10		40	
Simpangan Ideal													4						2						2						2		12	

Kategori Kalayakan	Relevansi				Interval Kriteria								Keseluruhan				Ket				
	40,8	≤	X	≤	48	13,6	≤	X	≤	16	13,6	≤	X	≤	16	81,6		≤	X	≤	96
Sangat Layak	33,6	≤	X	≤	40,8	11,2	≤	X	≤	13,6	11,2	≤	X	≤	13,6	67,2	≤	X	≤	81,6	SL
Layak	26,4	≤	X	≤	33,6	8,8	≤	X	≤	11,2	8,8	≤	X	≤	11,2	54,4	≤	X	≤	67,2	SL
Cukup Layak	19,2	≤	X	≤	26,4	6,4	≤	X	≤	8,8	6,4	≤	X	≤	8,8	41,6	≤	X	≤	54,4	SL
Kurang Layak	15,2	≤	X	≤	19,2	4,8	≤	X	≤	6,4	4,8	≤	X	≤	6,4	38,4	≤	X	≤	52,8	SL
Tidak Layak	12	≤	X	≤	15,2	4	≤	X	≤	6,4	4	≤	X	≤	6,4	34	≤	X	≤	48	CS

Lampiran 6.3. Analisis Data Uji Pengguna

NIM	Kriteria																												Jml	Ktgr					
	Isi dan Tujuan						Jml	Ktgr	Instruksional								Jml	Ktgr	Teknik					Jml	Ktgr	Efektif dan Efisien		Jml			Ktgr				
																										19	20								
	1	2	3	4	5	7			6	12	13	14	15	16	17	18			8	9	10	11													
16518241001	4	3	3	4	4	3	21	SL	4	4	3	3	4	4	3	3	28	SL	3	3	4	4	14	SL	4	3	7	SL	70	SL					
16518241002	4	3	3	4	3	3	20	L	3	4	3	3	3	2	3	2	23	L	3	4	3	3	13	L	3	3	6	L	62	L					
16518241003	4	4	4	4	4	4	24	SL	4	4	4	4	4	4	4	4	32	SL	4	3	4	2	13	L	4	4	8	SL	77	SL					
16518241004	4	4	3	4	4	3	22	SL	4	4	4	3	4	4	4	4	31	SL	4	4	3	4	15	SL	4	4	8	SL	76	SL					
16518241006	4	3	3	4	3	3	20	L	3	4	4	3	3	3	3	3	26	L	4	3	3	2	12	L	4	4	8	SL	66	L					
16518241007	4	4	3	4	4	4	23	SL	4	4	4	4	4	4	4	4	32	SL	4	4	4	3	15	SL	4	4	8	SL	78	SL					
16518241009	4	4	4	4	4	4	24	SL	4	4	4	3	3	4	3	4	29	SL	4	4	4	2	14	SL	4	4	8	SL	75	SL					
16518241010	3	3	3	4	3	3	19	L	3	4	4	2	3	3	4	2	25	L	2	4	3	3	12	L	3	3	6	L	62	L					
16518241011	3	4	3	3	3	3	19	L	4	3	3	3	4	3	3	3	26	L	3	3	4	2	12	L	3	3	6	L	63	L					
16518241012	4	3	3	4	3	3	20	L	3	4	3	3	3	3	3	3	25	L	3	4	3	3	13	L	3	3	6	L	64	L					
16518241013	4	4	4	4	4	4	24	SL	4	4	4	4	4	4	4	3	31	SL	3	4	4	3	14	SL	4	3	7	SL	76	SL					
16518241014	4	3	3	4	4	3	21	SL	3	4	4	4	4	3	3	3	28	SL	3	3	4	3	13	L	4	4	8	SL	70	SL					
16518241015	4	4	4	4	4	4	24	SL	4	4	4	4	4	4	4	3	31	SL	4	4	4	4	16	SL	4	3	7	SL	78	SL					
16518241016	4	4	4	4	3	4	23	SL	4	3	3	3	3	4	3	4	27	L	4	4	3	4	15	SL	4	3	7	SL	72	SL					
16518241018	4	4	4	4	4	3	23	SL	4	4	4	4	4	3	4	3	30	SL	3	4	4	4	15	SL	4	4	8	SL	76	SL					
16518241020	4	3	3	4	4	3	21	SL	3	4	4	3	3	4	4	4	29	SL	4	3	3	4	14	SL	4	4	8	SL	72	SL					
16518241021	4	4	4	4	4	4	24	SL	4	4	4	4	4	4	4	3	31	SL	3	4	3	3	13	L	4	4	8	SL	76	SL					
16518241022	3	4	4	3	4	4	22	SL	3	3	4	3	3	4	3	4	27	L	4	4	4	4	16	SL	4	3	7	SL	72	SL					
16518241023	4	4	3	3	4	3	21	SL	4	3	3	3	4	3	4	3	27	L	4	3	4	3	14	SL	3	4	7	SL	69	SL					
16518241024	4	4	4	3	4	3	22	SL	4	4	3	4	4	4	3	3	29	SL	3	4	3	4	14	SL	4	4	8	SL	73	SL					
16518241025	4	4	4	4	3	3	22	SL	4	4	4	4	3	3	4	3	29	SL	3	3	4	3	13	L	3	3	6	L	70	SL					
16518241026	4	4	4	4	4	4	24	SL	4	4	4	3	4	4	4	3	30	SL	3	4	4	4	15	SL	3	3	6	L	75	SL					
16518241027	4	3	4	4	3	4	22	SL	4	4	4	3	3	3	4	3	28	SL	3	4	3	4	14	SL	4	4	8	SL	72	SL					
16518241028	4	3	4	4	3	4	22	SL	4	4	3	4	4	4	4	3	30	SL	4	3	3	3	13	L	4	4	8	SL	73	SL					
16518241029	4	4	3	4	4	3	22	SL	3	4	4	2	4	4	3	3	27	L	3	3	3	3	12	L	3	3	6	L	67	L					
16518241030	4	3	3	4	4	4	22	SL	4	4	4	4	4	4	4	4	32	SL	3	4	3	3	13	L	4	4	8	SL	75	SL					
16518241031	4	3	3	4	4	3	21	SL	3	4	4	3	4	4	4	3	29	SL	4	4	3	4	15	SL	4	4	8	SL	73	SL					
16518241032	4	3	3	4	4	3	21	SL	4	4	3	3	3	4	4	4	29	SL	3	3	3	4	13	L	4	4	8	SL	71	SL					
16518241033	4	4	4	4	4	4	24	SL	4	4	3	3	3	4	3	3	27	L	4	4	3	3	14	SL	3	3	6	L	71	SL					
16518241034	4	4	3	4	3	3	21	SL	4	4	4	3	4	4	3	4	30	SL	3	4	3	3	13	L	4	4	8	SL	72	SL					
16518241035	4	4	4	4	3	4	23	SL	4	4	4	4	4	4	3	4	31	SL	3	3	4	4	14	SL	4	3	7	SL	75	SL					
16518241037	3	4	3	3	4	4	21	SL	3	3	3	4	3	4	4	4	28	SL	3	3	4	4	14	SL	3	3	6	L	69	SL					
16518241038	4	3	4	4	4	3	22	SL	3	4	4	4	4	4	4	4	31	SL	3	3	3	3	12	L	3	4	7	SL	72	SL					
16518241040	4	3	3	3	4	3	20	L	3	4	3	3	3	3	3	2	24	L	3	3	3	3	12	L	3	3	6	L	62	L					
Jumlah							744										972						464			244		2424							
Rata-rata							21,88	SL									28,59	SL					13,65	SL			7,18	SL	71,29	SL					
Persentase							91,18%										89,34%						85,29%			89,71%		89,12%							
Jumlah Butir							6										8						4			2		20							
Skor Maks							24										32						16			8		80							
Skor Min							6										8						4			2		20							
Rerata Ideal							15										20						10			5		50							
Simpangan Ideal							3										4						2			1		10							

Kategori Kelayakan	Kriteria															Keseluruhan	Ket									
	Isi dan Tujuan					Instruksional					Teknik							Efektif dan Efisien								
Sangat Layak	20,4	≤	X	≤	24	27,2	≤	X	≤	32	13,6	≤	X	≤	16	6,8	≤	X	≤	8	68	≤	X	≤	80	SL
Layak	16,8	≤	X	<	20,4	22,4	≤	X	<	27,2	11,2	≤	X	<	13,6	5,6	≤	X	<	6,8	56	≤	X	<	68	L
Cukup Layak	13,2	≤	X	<	16,8	17,6	≤	X	<	22,4	8,8	≤	X	<	11,2	4,4	≤	X	<	5,6	44	≤	X	<	56	CL
Kurang Layak	9,6	≤	X	<	13,2	12,8	≤	X	<	17,6	6,4	≤	X	<	8,8	3,2	≤	X	<	4,4	32	≤	X	<	44	KL
Tidak Layak	6	≤	X	<	9,6	8	≤	X	<	12,8	4	≤	X	<	6,4	2	≤	X	<	3,2	20	≤	X	<	32	TL

Lampiran 6.4. Analisis Data Reliabilitas Instrumen

NIM Responden	Butir																				Total	y1	y2	d
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20				
16518241001	4	3	3	4	4	4	3	3	3	4	4	4	3	3	4	4	3	3	4	3	70	35	35	0
16518241002	4	3	3	4	3	3	3	3	4	3	3	4	3	3	3	2	3	2	3	3	62	32	30	2
16518241003	4	4	4	4	4	4	4	4	3	4	2	4	4	4	4	4	4	4	4	4	77	37	40	-3
16518241004	4	4	3	4	4	4	3	4	4	3	4	4	4	3	4	4	4	4	4	4	76	38	38	0
16518241006	4	3	3	4	3	3	3	4	3	3	2	4	4	3	3	3	3	3	4	4	66	32	34	-2
16518241007	4	4	3	4	4	4	4	4	4	4	3	4	4	4	4	4	4	4	4	4	78	38	40	-2
16518241009	4	4	4	4	4	4	4	4	4	4	2	4	4	3	3	4	3	4	4	4	75	36	39	-3
16518241010	3	3	3	4	3	3	3	2	4	3	3	4	4	2	3	3	4	2	3	3	62	33	29	4
16518241011	3	4	3	3	3	4	3	3	3	4	2	3	3	3	4	3	3	3	3	3	63	30	33	-3
16518241012	4	3	3	4	3	3	3	3	4	3	3	4	3	3	3	3	3	3	3	3	64	32	32	0
16518241013	4	4	4	4	4	4	4	3	4	4	3	4	4	4	4	4	4	3	4	3	76	39	37	2
16518241014	4	3	3	4	4	3	3	3	3	4	3	4	4	4	4	3	3	3	4	4	70	35	35	0
16518241015	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	3	4	3	78	40	38	2
16518241016	4	4	4	4	3	4	4	4	4	3	4	3	3	3	3	4	3	4	4	3	72	36	36	0
16518241018	4	4	4	4	4	4	3	3	4	4	4	4	4	4	4	3	4	3	4	4	76	39	37	2
16518241020	4	3	3	4	4	3	3	4	3	3	4	4	4	3	3	4	4	4	4	4	72	36	36	0
16518241021	4	4	4	4	4	4	4	3	4	3	3	4	4	4	4	4	4	3	4	4	76	39	37	2
16518241022	3	4	4	3	4	3	4	4	4	4	4	3	4	3	3	4	3	4	4	3	72	37	35	2
16518241023	4	4	3	3	4	4	3	4	3	4	3	3	3	3	4	3	4	3	3	4	69	34	35	-1
16518241024	4	4	4	3	4	4	3	3	4	3	4	4	3	4	4	4	3	3	4	4	73	37	36	1
16518241025	4	4	4	4	3	4	3	3	3	4	3	4	4	4	3	3	4	3	3	3	70	34	36	-2
16518241026	4	4	4	4	4	4	4	3	4	4	4	4	4	3	4	4	4	3	3	3	75	39	36	3
16518241027	4	3	4	4	3	4	4	3	4	3	4	4	4	3	3	3	4	3	4	4	72	38	34	4
16518241028	4	3	4	4	3	4	4	4	3	3	3	4	3	4	4	4	4	3	4	4	73	36	37	-1
16518241029	4	4	3	4	4	3	3	3	3	3	3	4	4	2	4	4	3	3	3	3	67	34	33	1
16518241030	4	3	3	4	4	4	4	3	4	3	3	4	4	4	4	4	4	4	4	4	75	38	37	1
16518241031	4	3	3	4	4	3	3	4	4	3	4	4	4	3	4	4	4	3	4	4	73	38	35	3
16518241032	4	3	3	4	4	4	3	3	3	3	4	4	3	3	3	4	4	4	4	4	71	35	36	-1
16518241033	4	4	4	4	4	4	4	4	4	3	3	4	3	3	3	4	3	3	3	3	71	35	36	-1
16518241034	4	4	3	4	3	4	3	3	4	3	3	4	4	3	4	4	3	4	4	4	72	35	37	-2
16518241035	4	4	4	4	3	4	4	3	3	4	4	4	4	4	4	4	3	4	4	3	75	37	38	-1
16518241037	3	4	3	3	4	3	4	3	3	4	4	3	3	4	3	4	4	4	3	3	69	34	35	-1
16518241038	4	3	4	4	4	3	3	3	3	3	3	4	4	4	4	4	4	4	3	4	72	36	36	0
16518241040	4	3	3	3	4	3	3	3	3	3	3	4	3	3	3	3	3	2	3	3	62	32	30	2

Alpha Belah Dua	Varian	6,19	6,68
	Varian Total	21,85	
	r11	0,82232	
	Realibilitas	Sangat Tinggi	

Spearman Brown	Korelasi	0,70
	r11	0,82268
	Realibilitas	Sangat Tinggi

Rulon	Varian Bada	3,88
	Varian Total	21,85
	r11	0,82232
	Realibilitas	Sangat Tinggi

Sukiman (2012: 184)					
Interval					Tingkat
0	≤	X	<	0,19	Sangat Rendah
0,2	≤	X	<	0,39	Rendah
0,4	≤	X	<	0,59	Cukup
0,6	≤	X	<	0,79	Tinggi
0,8	≤	X	≤	1	Sangat Tinggi

Lampiran 7 Berkas Penelitian

Lampiran 7.1. Surat Keputusan Pelaksanaan Tugas Akhir Skripsi

Lampiran 7.1. Surat Keputusan Pelaksanaan Tugas Akhir Skripsi

**KEPUTUSAN DEKAN FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
NOMOR : 183/PMK/PB/XI/2018**

**TENTANG
PENGANGKATAN DOSEN PEMBIMBING TUGAS AKHIR SKRIPSI (TAS) MAHASISWA
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

DEKAN FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

Menimbang : a. bahwa untuk kelancaran pelaksanaan kegiatan Tugas Akhir Skripsi (TAS) mahasiswa, dipandang perlu mengangkat dosen pembimbingnya;
b. bahwa untuk keperluan sebagaimana dimaksud pada huruf a perlu menetapkan Keputusan Dekan Tentang Pengangkatan Dosen Pembimbing Tugas Akhir Skripsi (TAS) Fakultas Teknik Universitas Negeri Yogyakarta.

Mengingat : 1. Undang-undang RI Nomor 20 Tahun 2003 Tentang Sistem Pendidikan Nasional (Lembaran Negara Tahun 2003 Nomor 78, Tambahan Lembaran Negara Nomor 4301);
2. Peraturan Pemerintah Republik Indonesia Nomor 4 Tahun 2014 Tentang Penyelenggaraan Pendidikan Tinggi dan Pengelolaan Perguruan Tinggi (Lembaran Negara Tahun 2014 Nomor 16, Tambahan Lembaran Negara Republik Indonesia Nomor 5500);
3. Keputusan Presiden Republik Indonesia Nomor 93 Tahun 1999 Tentang Perubahan Institut Keguruan dan Ilmu Pendidikan menjadi Universitas;
4. Peraturan Mendiknas RI Nomor 23 Tahun 2011 Tentang Organisasi dan Tata Kerja Universitas Negeri Yogyakarta;
5. Peraturan Mendiknas RI Nomor 34 Tahun 2011 Tentang Statuta Universitas Negeri Yogyakarta;
6. Keputusan Menteri Pendidikan dan Kebudayaan RI Nomor 98/MPK.A4/KP/2013 Tentang Pengangkatan Rektor Universitas Negeri Yogyakarta;
7. Peraturan Rektor Nomor 2 Tahun 2014 tentang Peraturan Akademik;
8. Keputusan Rektor Nomor 800/UN.34/KP/2016 tahun 2016 tentang Pengangkatan Dekan Fakultas Teknik Universitas Negeri Yogyakarta.

MEMUTUSKAN

Menetapkan : **KEPUTUSAN DEKAN TENTANG PENGANGKATAN DOSEN PEMBIMBING TUGAS AKHIR SKRIPSI (TAS) FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA.**

PERTAMA : Mengangkat Saudara :

Nama	: Hcrlambang Sigit Pramono, S.T., M.Cs
NIP	: 19650829 199903 1 001
Pangkat/Golongan	: Penata Tk.I, III/d
Jabatan Akademik	: Lektor

sebagai Dosen Pembimbing Untuk mahasiswa penyusun Tugas Akhir Skripsi (TAS) :

Nama	: Ihsan Syarifuddin
NIM	: 14518244016
Prodi Studi	: Pend. Teknik Mekatronika - S1
Judul Skripsi/TA	: PENGEMBANGAN APLIKASI PENDETEKSI OBJEK BERSEGI MENGGUNAKAN METODE CONTOUR SEBAGAI MEDIA PEMBELAJARAN PENGINDERAAN VISUAL ROBOT

- KEDUA : Dosen Pembimbing sebagaimana dimaksud dalam Diktum PERTAMA bertugas merencanakan, mempersiapkan, melaksanakan, dan mempertanggungjawabkan pelaksanaan kegiatan bimbingan terhadap mahasiswa sebagaimana dimaksud dalam Diktum PERTAMA sampai mahasiswa dimaksud dinyatakan lulus.
- KETIGA : Biaya yang diperlukan dengan adanya Keputusan ini dibebankan pada Anggaran DIPA Universitas Negeri Yogyakarta Tahun 2018.
- KEEMPAT : Keputusan ini berlaku sejak tanggal 7 November 2018.

Tembusan Keputusan Dekan ini disampaikan kepada :

1. Para Wakil Dekan Fakultas Teknik;
 2. Kepala Bagian Tata Usaha Fakultas Teknik;
 3. Kepala Subbagian Keuangan dan Akuntansi Fakultas Teknik;
 4. Kepala Subbagian Pendidikan Fakultas Teknik;
 5. Ketua Jurusan Pendidikan Teknik Elektro Fakultas Teknik;
 6. Mahasiswa yang bersangkutan;
- Universitas Negeri Yogyakarta.

Ditetapkan di : Yogyakarta
Pada tanggal : 7 November 2018

DEKAN FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA,



Dr. Ir. Drs. WIDARTO, M.Pd.
NIP. 19631230 198812 1 001

Lampiran 8 Program Aplikasi Pendeteksi Objek Bersegi

Lampiran 8.1. *Splash Screen*

Lampiran 8.2. Tampilan Utama

Lampiran 8.3. Pengaturan

Lampiran 8.4. Tujuan

Lampiran 8.5. Tentang

Lampiran 8.6. Bantuan

Lampiran 8.1. *Splash Screen*



splashScreen.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Skripsi
{
    public partial class splashScreen : Form
    {
        Random rnd = new Random();
        Main main = new Main();
        int value = 0;
        int splash = 0;
        public splashScreen()
        {
            InitializeComponent();
        }

        private void timer1_Tick(object sender, EventArgs e)
        {
            if (splash == 0)
            {
                int val = rnd.Next(10);
```

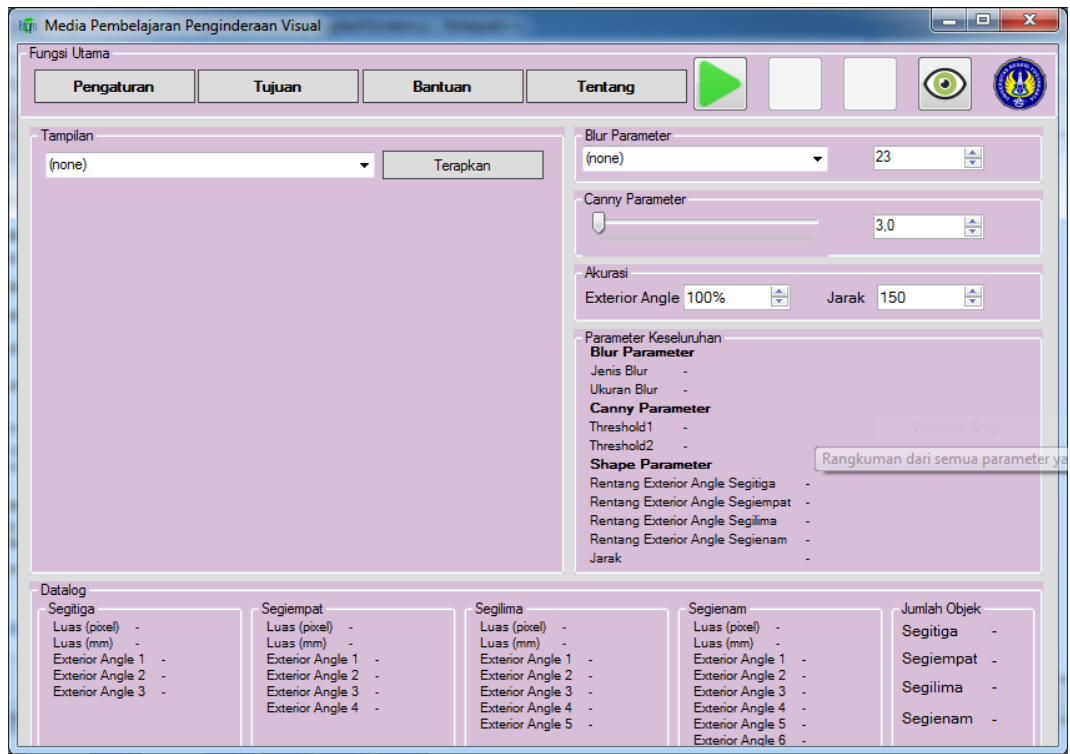
```

        value += val;
        if (value > 100)
        {
            value = 100;
        }
        progressBar1.Value = value;
        if(progressBar1.Value == 100)
        {
            splash = 1;
        }
    }

    if(splash == 1)
    {
        splash = 2;
        main.Show();
        this.Hide();
    }
}
}
}
}

```

Lampiran 8.2. Tampilan Utama



Main.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Diagnostics;
using Microsoft.Win32;
using Emgu.CV;
using Emgu.CV.CvEnum;
using Emgu.CV.Structure;
using Emgu.CV.Util;
using System.IO;
using System.Xml.Serialization;

namespace Skripsi
{
    public partial class Main : Form
```

```

{
    Configuration config = new Configuration();
    About about = new About();
    tujuan tujuan = new tujuan();
    Capture cap = null;
    Mat frame = new Mat();
    Mat blur = new Mat();
    Mat cannyIn = new Mat();
    Mat cannyOut = new Mat();
    Mat output = new Mat();

    int cameraIndex, outputIndex;
    bool segitiga, segiempat, segilima, segienam;
    int blurSize, indexBlur;
    double ratioCanny, threshold1, threshold2;
    int segitigaAngle = 120, luasAsli3 = 2305;
    int segiempatAngle = 90, luasAsli4 = 2500;
    int segilimaAngle = 72, luasAsli5 = 2632;
    int segienamAngle = 60, luasAsli6 = 2330;
    int accVal;
    int range;
    int jarak;
    int jmlh3, jmlh4, jmlh5, jmlh6;
    double segitigaMin, segitigaMax;
    double segiempatMin, segiempatMax;
    double segilimaMin, segilimaMax;
    double segienamMin, segienamMax;

    double[] angle3 = new double[3];
    double[] angle4 = new double[4];
    double[] angle5 = new double[5];
    double[] angle6 = new double[6];
    bool bool3 = false;
    bool bool4 = false;
    bool bool5 = false;
    bool bool6 = false;

    public Main()
    {
        InitializeComponent();
    }

    private void Main_FormClosed(object sender,
FormClosedEventArgs e)
    {
        Application.Exit();
        try
        {
            File.Delete("cameraConfig.xml");

```

```

    }
    catch (Exception ex)
    { }
}

private void button1_Click(object sender, EventArgs e)
{
    config.Show();
    this.Hide();
}

private void Main_Load(object sender, EventArgs e)
{
    for(int x = 100; x >= 0; x--)
    {
        domainUpDown3.Items.Add(x.ToString() + "%");
    }
    domainUpDown4.SelectedIndex = 0;
    domainUpDown3.SelectedItem = "100%";
    domainUpDown1.SelectedIndex = 0;
    domainUpDown2.SelectedIndex = 0;
    comboBox1.SelectedIndex = 0;
    comboBox2.SelectedIndex = 0;

    if(File.Exists("cameraConfig.xml"))
    {
        XmlSerializer rd = new
        XmlSerializer(typeof(Information));
        FileStream read = new
        FileStream("cameraConfig.xml", FileMode.Open, FileAccess.Read,
        FileShare.Read);
        Information info =
        (Information)rd.Deserialize(read);
        cameraIndex = info.CameraIndex;
        segitiga = info.Segitiga;
        segiempat = info.Segiempat;
        segilima = info.Segilima;
        segienam = info.Segienam;
        read.Dispose();
        read.Close();

        button5.Enabled = true;
        button5.BackgroundImage =
        global::Skripsi.Properties.Resources.Custom_Icon_Design_Flatastic
        _9_Start;
    }
    else
    {
        button5.Enabled = false;
    }
}

```



```

        button5.BackgroundImage = null;
    }

    button6.Enabled = false;
    button6.BackgroundImage = null;
    button7.Enabled = false;
    button7.BackgroundImage = null;
}

private void button6_Click(object sender, EventArgs e)
{
    cap.Pause();
    button5.Enabled = true;
    button5.BackgroundImage =
global::Skripsi.Properties.Resources.Custom_Icon_Design_Flatatic
_9_Start;
    button6.Enabled = false;
    button6.BackgroundImage = null;
    button7.Enabled = false;
    button7.BackgroundImage = null;

}

private void button7_Click(object sender, EventArgs e)
{
    Application.Restart();
}

private void button5_Click(object sender, EventArgs e)
{
    cap = new Capture(cameraIndex);
    cap.ImageGrabbed += processFrame;
    cap.Start();
    timer1.Start();
    button5.Enabled = false;
    button5.BackgroundImage = null;
    button6.Enabled = true;
    button6.BackgroundImage =
global::Skripsi.Properties.Resources.multimedia;
    button7.Enabled = true;
    button7.BackgroundImage =
global::Skripsi.Properties.Resources.stop_flat;
}

private void processFrame(object sender, EventArgs e)
{
    #region initialize
    segitigaMin = (segitigaAngle * range) / 100;
    segitigaMin = Math.Round(segitigaMin, 2);

```

```

segitigaMax = (segitigaAngle * 2) - segitigaMin;
segitigaMax = Math.Round(segitigaMax, 2);

segiempatMin = (segiempatAngle * range) / 100;
segiempatMin = Math.Round(segiempatMin, 2);
segiempatMax = (segiempatAngle * 2) - segiempatMin;
segiempatMax = Math.Round(segiempatMax, 2);

segilimaMin = (segilimaAngle * range) / 100;
segilimaMin = Math.Round(segilimaMin, 2);
segilimaMax = (segilimaAngle * 2) - segilimaMin;
segilimaMax = Math.Round(segilimaMax, 2);

segienamMin = (segienamAngle * range) / 100;
segienamMin = Math.Round(segienamMin, 2);
segienamMax = (segienamAngle * 2) - segienamMin;
segienamMax = Math.Round(segienamMax, 2);

bool3 = false;
bool4 = false;
bool5 = false;
bool6 = false;
angle3.Initialize();
angle4.Initialize();
angle5.Initialize();
angle6.Initialize();
#endregion

#region gambar asli
cap.Retrieve(frame, 0);
output = frame.Clone();
#endregion

#region blur
if (indexBlur == 1)
{
    CvInvoke.Blur(frame, blur, new Size(blurSize,
blurSize), new Point(-1, -1), BorderType.Default);
}
else if(indexBlur == 2)
{
    CvInvoke.GaussianBlur(frame, blur, new
Size(blurSize, blurSize), 3, 0, BorderType.Default);
}
else if(indexBlur == 3)
{
    CvInvoke.MedianBlur(frame, blur, blurSize);
}
else

```

```

        {
            blur = frame.Clone();
        }
    #endregion

    #region canny
    CvInvoke.CvtColor(blur, cannyIn,
ColorConversion.Bgr2Gray);
    CvInvoke.Canny(cannyIn, cannyOut, threshold1,
threshold2);
    #endregion

    #region bentuk
    jmlh3 = 0;
    jmlh4 = 0;
    jmlh5 = 0;
    jmlh6 = 0;
    using(VectorOfVectorOfPoint contours = new
VectorOfVectorOfPoint())
    {
        CvInvoke.FindContours(cannyOut, contours, null,
RetrType.External, ChainApproxMethod.ChainApproxSimple);
        int count = contours.Size;
        for(int i = 0; i < count; i++)
        {
            using (VectorOfPoint contour = contours[i])
            using (VectorOfPoint approx = new
VectorOfPoint())
            {
                CvInvoke.ApproxPolyDP(contour, approx,
CvInvoke.ArcLength(contour, true) * 0.02, true);
                Point[] pts = approx.ToArray();
                double multiply = 0.0004762867 *
Math.Pow(jarak, 2.712263);
                if(CvInvoke.ContourArea(approx, false) >
500)
                {
                    #region cek segtiga
                    if (approx.Size == 3 && segitiga ==
true)
                    {
                        Point[] pt = approx.ToArray();
                        LineSegment2D[] edges =
PointCollection.PolyLine(pt, true);

                        for (int j = 0; j < edges.Length;
j++)
                        {

```

```

double angle =
Math.Abs(edges[(j + 1) %
edges.Length].GetExteriorAngleDegree(edges[j]));
if(angle < segitigaMin ||
angle > segitigaMax)
{
    bool3 = false;
    break;
}
else
{
    angle3[j] = angle;
    bool3 = true;
}
}

if (bool3 == true)
{
    CvInvoke.Polylines(output,
pt, true, new MCvScalar(0, 0, 255), 3, LineType.EightConnected, 0);
    angle31Lbl.Invoke(new
Action(() => angle31Lbl.Text = Math.Round(angle3[0],
2).ToString()));
    angle32Lbl.Invoke(new
Action(() => angle32Lbl.Text = Math.Round(angle3[1],
2).ToString()));
    angle33Lbl.Invoke(new
Action(() => angle33Lbl.Text = Math.Round(angle3[2],
2).ToString()));

    double luasPX =
CvInvoke.ContourArea(approx, false);
    luasPx3Lbl.Invoke(new
Action(() => luasPx3Lbl.Text = Math.Round(luasPX, 2).ToString()));

    double luasMM = luasPX *
(multiply / jarak);
    luasMm3Lbl.Invoke(new
Action(() => luasMm3Lbl.Text = Math.Round(luasMM, 2).ToString()));
    jmlh3++;
}
}
#endregion

#region cek segiempat
if (approx.Size == 4 && segiempat ==
true)
{
    Point[] pt = approx.ToArray();

```

```

LineSegment2D[] edges =
PointCollection.PolyLine(pt, true);

for (int j = 0; j < edges.Length;
j++)
{
    double angle =
Math.Abs(edges[(j + 1) %
edges.Length].GetExteriorAngleDegree(edges[j]));
    Console.WriteLine(angle);
    if (angle < segiempatMin ||
angle > segiempatMax)
    {
        bool4 = false;
        break;
    }
    else
    {
        angle4[j] = angle;
        bool4 = true;
    }
}

if (bool4 == true)
{
    CvInvoke.PolyLines(output,
pt, true, new MCvScalar(0, 255, 0), 3, LineType.EightConnected, 0);
    angle41Lbl.Invoke(new
Action(() => angle41Lbl.Text = Math.Round(angle4[0],
2).ToString()));
    angle42Lbl.Invoke(new
Action(() => angle42Lbl.Text = Math.Round(angle4[1],
2).ToString()));
    angle43Lbl.Invoke(new
Action(() => angle43Lbl.Text = Math.Round(angle4[2],
2).ToString()));
    angle44Lbl.Invoke(new
Action(() => angle44Lbl.Text = Math.Round(angle4[3],
2).ToString()));

    double luasPX =
CvInvoke.ContourArea(approx, false);
    luasPx4Lbl.Invoke(new
Action(() => luasPx4Lbl.Text = Math.Round(luasPX, 2).ToString()));

    double luasMM = luasPX *
(multiply / jarak);
    luasMm4Lbl.Invoke(new
Action(() => luasMm4Lbl.Text = Math.Round(luasMM, 2).ToString()));

```

```

        jmlh4++;
    }
}
#endregion

#region cek segilima
if (approx.Size == 5 && segilima ==
true)
{
    Point[] pt = approx.ToArray();
    LineSegment2D[] edges =
PointCollection.PolyLine(pt, true);

    for (int j = 0; j < edges.Length;
j++)
    {
        double angle =
Math.Abs(edges[(j + 1) %
edges.Length].GetExteriorAngleDegree(edges[j]));
        Console.WriteLine(angle);
        if (angle < segilimaMin ||
angle > segilimaMax)
        {
            bool5 = false;
            break;
        }
        else
        {
            angle5[j] = angle;
            bool5 = true;
        }
    }

    if (bool5 == true)
    {
        CvInvoke.PolyLines(output,
pt, true, new MCvScalar(255, 0, 0), 3, LineType.EightConnected, 0);
        angle51Lbl.Invoke(new
Action(() => angle51Lbl.Text = Math.Round(angle5[0],
2).ToString()));
        angle52Lbl.Invoke(new
Action(() => angle52Lbl.Text = Math.Round(angle5[1],
2).ToString()));
        angle53Lbl.Invoke(new
Action(() => angle53Lbl.Text = Math.Round(angle5[2],
2).ToString()));
        angle54Lbl.Invoke(new
Action(() => angle54Lbl.Text = Math.Round(angle5[3],
2).ToString()));
    }
}

```

```

        angle55Lb1.Invoke(new
Action(() => angle55Lb1.Text = Math.Round(angle5[4],
2).ToString()));

        double luasPX =
CvInvoke.ContourArea(approx, false);
        luasPx5Lb1.Invoke(new
Action(() => luasPx5Lb1.Text = Math.Round(luasPX, 2).ToString()));

        double luasMM = luasPX *
(multiply / jarak);
        luasMm5Lb1.Invoke(new
Action(() => luasMm5Lb1.Text = Math.Round(luasMM, 2).ToString()));
        jmlh5++;
    }
}
#endregion

#region cek segienam
if (approx.Size == 6 && segienam ==
true)
{
    Point[] pt = approx.ToArray();
    LineSegment2D[] edges =
PointCollection.PolyLine(pt, true);

    for (int j = 0; j < edges.Length;
j++)
    {
        double angle =
Math.Abs(edges[(j + 1) %
edges.Length].GetExteriorAngleDegree(edges[j]));
        Console.WriteLine(angle);
        if (angle < segienamMin ||
angle > segienamMax)
        {
            bool6 = false;
            break;
        }
        else
        {
            angle6[j] = angle;
            bool6 = true;
        }
    }

    if (bool6 == true)
    {

```

```

CvInvoke.Polylines(output,
pt, true, new MCvScalar(150, 0, 150), 3, LineType.EightConnected,
0);

angle61Lbl.Invoke(new
Action(() => angle61Lbl.Text = Math.Round(angle6[0],
2).ToString()));

angle62Lbl.Invoke(new
Action(() => angle62Lbl.Text = Math.Round(angle6[1],
2).ToString()));

angle63Lbl.Invoke(new
Action(() => angle63Lbl.Text = Math.Round(angle6[2],
2).ToString()));

angle64Lbl.Invoke(new
Action(() => angle64Lbl.Text = Math.Round(angle6[3],
2).ToString()));

angle65Lbl.Invoke(new
Action(() => angle65Lbl.Text = Math.Round(angle6[4],
2).ToString()));

angle66Lbl.Invoke(new
Action(() => angle66Lbl.Text = Math.Round(angle6[5],
2).ToString()));

double luasPX =
CvInvoke.ContourArea(approx, false);
luasPx6Lbl.Invoke(new
Action(() => luasPx6Lbl.Text = Math.Round(luasPX, 2).ToString()));

double luasMM = luasPX *
(multiply / jarak);
luasMm6Lbl.Invoke(new
Action(() => luasMm6Lbl.Text = Math.Round(luasMM, 2).ToString()));
jmlh6++;
}
}
#endregion

#region cek untuk reset nilai label
if (bool3 == false)
{
luasPx3Lbl.Invoke(new Action(() =>
luasPx3Lbl.Text = "-"));
luasMm3Lbl.Invoke(new Action(() =>
luasMm3Lbl.Text = "-"));
angle31Lbl.Invoke(new Action(() =>
angle31Lbl.Text = "-"));
angle32Lbl.Invoke(new Action(() =>
angle32Lbl.Text = "-"));
}
}

```



```

angle33Lb1.Text = "-"));
    }

    if (bool4 == false)
    {
        luasPx4Lb1.Invoke(new Action(() =>
        luasMm4Lb1.Invoke(new Action(() =>
        luasMm4Lb1.Text = "-"));
        angle41Lb1.Invoke(new Action(() =>
        angle41Lb1.Text = "-"));
        angle42Lb1.Invoke(new Action(() =>
        angle42Lb1.Text = "-"));
        angle43Lb1.Invoke(new Action(() =>
        angle43Lb1.Text = "-"));
        angle44Lb1.Invoke(new Action(() =>
        angle44Lb1.Text = "-"));
    }

    if (bool5 == false)
    {
        luasPx5Lb1.Invoke(new Action(() =>
        luasMm5Lb1.Invoke(new Action(() =>
        luasMm5Lb1.Text = "-"));
        angle51Lb1.Invoke(new Action(() =>
        angle51Lb1.Text = "-"));
        angle52Lb1.Invoke(new Action(() =>
        angle52Lb1.Text = "-"));
        angle53Lb1.Invoke(new Action(() =>
        angle53Lb1.Text = "-"));
        angle54Lb1.Invoke(new Action(() =>
        angle54Lb1.Text = "-"));
        angle55Lb1.Invoke(new Action(() =>
        angle55Lb1.Text = "-"));
    }

    if (bool6 == false)
    {
        luasPx6Lb1.Invoke(new Action(() =>
        luasMm6Lb1.Invoke(new Action(() =>
        luasMm6Lb1.Text = "-"));
        angle61Lb1.Invoke(new Action(() =>
        angle61Lb1.Text = "-"));
        angle62Lb1.Invoke(new Action(() =>
        angle62Lb1.Text = "-"));
    }

```

```

angle63Lbl.Text = "-"));
angle64Lbl.Text = "-"));
angle65Lbl.Text = "-"));
angle66Lbl.Text = "-"));
    }
    #endregion
}

#region tampil jumlah
segitigaJmlh.Invoke(new Action(() =>
segitigaJmlh.Text = jmlh3.ToString()));
segiempatJmlh.Invoke(new Action(() =>
segiempatJmlh.Text = jmlh4.ToString()));
segilimaJmlh.Invoke(new Action(() =>
segilimaJmlh.Text = jmlh5.ToString()));
segienamJmlh.Invoke(new Action(() =>
segienamJmlh.Text = jmlh6.ToString()));
#endregion
}
#endregion

#region tampilan
if (outputIndex == 1)
{
    imageBox1.Image = frame;
}
else if (outputIndex == 2)
{
    imageBox1.Image = blur;
}
else if (outputIndex == 3)
{
    imageBox1.Image = cannyOut;
}
else if (outputIndex == 4)
{
    imageBox1.Image = output;
}
else
{
    imageBox1.Image = null;
}
#endregion
}

```

```

private void timer1_Tick(object sender, EventArgs e)
{
    indexBlur = comboBox2.SelectedIndex;
    try
    {
        blurSize = Convert.ToInt32(domainUpDown1.Text);
        ratioCanny = Convert.ToDouble(domainUpDown2.Text);
        trackBar1.Maximum = (int)(255 / ratioCanny);

        threshold1 = trackBar1.Value;
        threshold2 = threshold1 * ratioCanny;
    }
    catch (Exception ex) { }

    label5.Text = ": " + comboBox2.Text;
    label6.Text = ": " + domainUpDown1.Text;

    label8.Text = ": " + threshold1.ToString();
    label7.Text = ": " + threshold2.ToString();

    jarakKamera.Text = domainUpDown4.Text + " cm";
    jarak = Convert.ToInt32(domainUpDown4.Text);
    accVal = domainUpDown3.SelectedIndex;
    range = 100 - accVal;

    if(segitiga == true)
    {
        label17.Text = ": " + segitigaMin.ToString() + " - " + segitigaMax.ToString();
    }
    if (segiempat == true)
    {
        label18.Text = ": " + segiempatMin.ToString() + " - " + segiempatMax.ToString();
    }
    if (segilima == true)
    {
        label19.Text = ": " + segilimaMin.ToString() + " - " + segilimaMax.ToString();
    }
    if (segienam == true)
    {
        label20.Text = ": " + segienamMin.ToString() + " - " + segienamMax.ToString();
    }
}

private void button4_Click(object sender, EventArgs e)

```

```

    {
        outputIndex = comboBox1.SelectedIndex;
    }

private void pictureBox2_Click(object sender, EventArgs e)
{
    cap.Pause();
    timer1.Stop();
}

private void pictureBox3_Click(object sender, EventArgs e)
{
    cap.Stop();
    timer1.Stop();
}

private void button2_Click(object sender, EventArgs e)
{
    Process.Start("help.pdf");
}

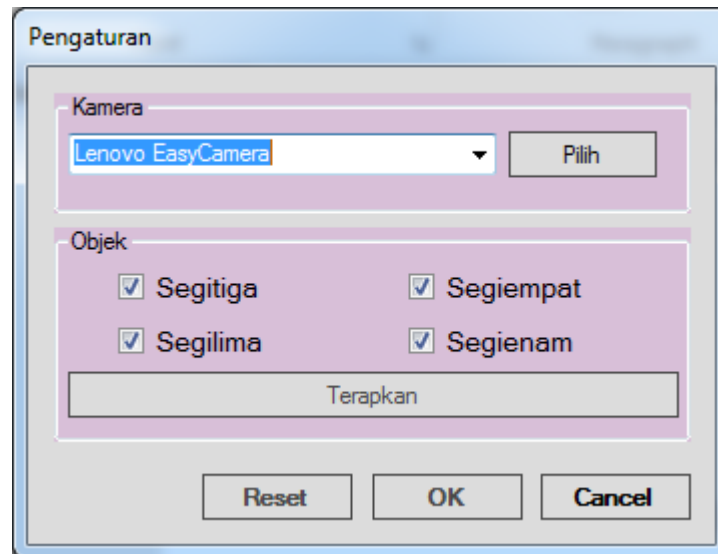
private void button3_Click(object sender, EventArgs e)
{
    about.Show();
    this.Hide();
}

private void button8_Click(object sender, EventArgs e)
{
    try
    {
        Process.Start("notepad++.exe", "code.cs");
    }
    catch (Exception ex)
    {
        Process.Start("npp.7.5.1.Installer.x64.exe");
    }
}

private void button9_Click(object sender, EventArgs e)
{
    tujuan.Show();
    this.Hide();
}
}
}

```

Lampiran 8.3. Pengaturan



Configuration.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using DirectShowLib;
using System.IO;
using System.Xml.Serialization;

namespace Skripsi
{
    public partial class Configuration : Form
    {
        int cameraIndex;
        bool segitiga, segiempat, segilima, segienam;
        public Configuration()
        {
            InitializeComponent();
        }

        public void readIndex()
        {
            List<KeyValuePair<int, string>> listCamera = new
            List<KeyValuePair<int, string>>();
```

```

DsDevice[] systemCameras =
DsDevice.GetDevicesOfCat(FilterCategory.VideoInputDevice);
int deviceIndex = 0;
foreach (DsDevice systemCamera in systemCameras)
{
    listCamera.Add(new KeyValuePair<int,
string>(deviceIndex, systemCamera.Name));
    deviceIndex++;
}
comboBox1.DataSource = null;
comboBox1.Items.Clear();

comboBox1.DataSource = new BindingSource(listCamera,
null);

comboBox1.DisplayMember = "Value";
comboBox1.ValueMember = "Key";
}

private void button3_Click(object sender, EventArgs e)
{
    try
    {
        Information info = new Information();
        info.CameraIndex = cameraIndex;
        info.Segitiga = segitiga;
        info.Segiempat = segiempat;
        info.Segilima = segilima;
        info.Segienam = segienam;
        saveConfig.saveData(info, "cameraConfig.xml");

        Main main = new Main();
        main.Show();
        this.Hide();
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}

private void Configuration_Load(object sender, EventArgs
e)
{
    readIndex();
    if(File.Exists("cameraConfig.xml"))
    {
        XmlSerializer rd = new
XmlSerializer(typeof(Information));

```

```

        FileStream read = new
FileStream("cameraConfig.xml", FileMode.Open, FileAccess.Read,
FileShare.Read);
        Information info =
(Information)rd.Deserialize(read);
        cameraIndex = info.CameraIndex;
        segitiga = info.Segitiga;
        segiempat = info.Segiempat;
        segilima = info.Segilima;
        segienam = info.Segienam;

        try
        {
            comboBox1.SelectedIndex = cameraIndex;
        }
        catch (Exception ex) { }

        checkBox1.Checked = segitiga;
        checkBox2.Checked = segiempat;
        checkBox3.Checked = segilima;
        checkBox4.Checked = segienam;
        read.Dispose();
        read.Close();
    }

    button1.Enabled = true;
    button2.Enabled = false;
    button3.Enabled = false;
    button4.Enabled = false;
}

private void button1_Click(object sender, EventArgs e)
{
    try
    {
        KeyValuePair<int, string> selectedItem =
(KeyValuePair<int, string>)comboBox1.SelectedItem;
        cameraIndex = selectedItem.Key;
    }
    catch (Exception ex) { }

    button1.Enabled = false;
    button2.Enabled = true;
    button3.Enabled = false;
    button4.Enabled = false;
}

private void button2_Click(object sender, EventArgs e)
{

```

```

        try
        {
            segitiga = checkBox1.Checked;
            segiempat = checkBox2.Checked;
            segilima = checkBox3.Checked;
            segienam = checkBox4.Checked;
        }
        catch (Exception ex) { }

        button1.Enabled = false;
        button2.Enabled = false;
        button3.Enabled = true;
        button4.Enabled = true;
    }

    private void button4_Click(object sender, EventArgs e)
    {
        cameraIndex = new int();
        segitiga = false;
        segiempat = false;
        segilima = false;
        segienam = false;

        checkBox1.Checked = segitiga;
        checkBox2.Checked = segiempat;
        checkBox3.Checked = segilima;
        checkBox4.Checked = segienam;

        button1.Enabled = true;
        button2.Enabled = false;
        button3.Enabled = false;
        button4.Enabled = false;
    }

    private void checkBox1_CheckedChanged(object sender,
    EventArgs e)
    {
        segitiga = checkBox1.Checked;
    }

    private void checkBox2_CheckedChanged(object sender,
    EventArgs e)
    {
        segiempat = checkBox2.Checked;
    }

    private void checkBox3_CheckedChanged(object sender,
    EventArgs e)
    {

```



```

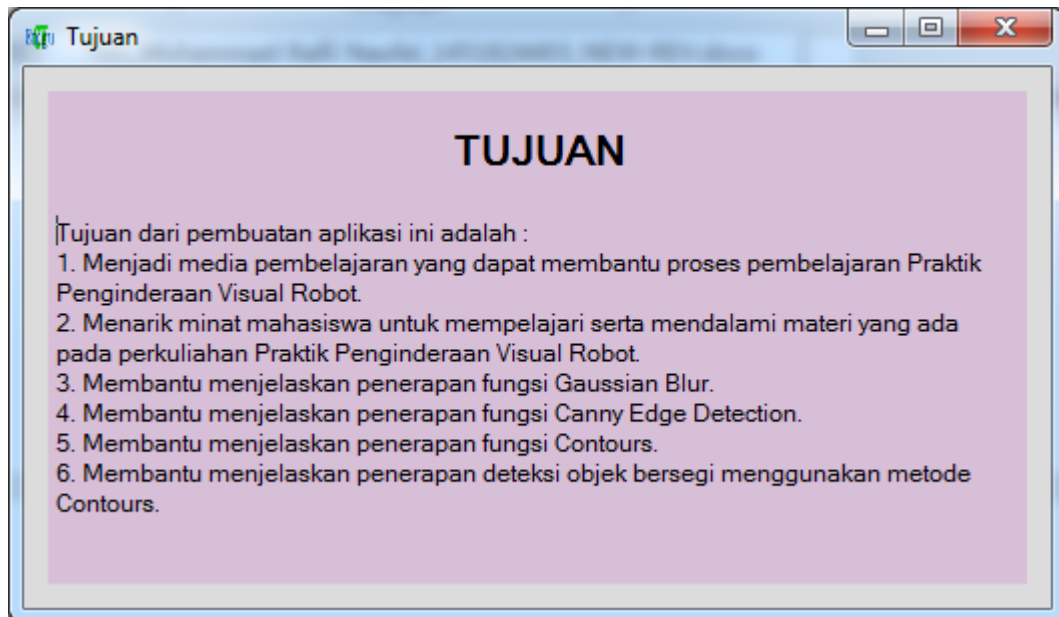
        segilima = checkBox3.Checked;
    }

    private void checkBox4_CheckedChanged(object sender,
EventArgs e)
    {
        segienam = checkBox4.Checked;
    }

    private void button5_Click(object sender, EventArgs e)
    {
        Main main = new Main();
        main.Show();
        this.Hide();
    }
}

```

Lampiran 8.4. Tujuan



tujuan.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
namespace Skripsi
{
    public partial class tujuan : Form
    {
        public tujuan()
        {
            InitializeComponent();
        }

        private void tujuan_FormClosing(object sender,
        FormClosingEventArgs e)
        {
            Main main = new Main();
            main.Show();
        }
    }
}
```

Lampiran 8.5. Tentang



About.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Skripsi
{
    public partial class About : Form
    {
        public About()
        {
            InitializeComponent();
        }

        private void About_FormClosing(object sender,
        FormClosingEventArgs e)
        {
            Main main = new Main();
            main.Show();
        }
    }
}
```

Lampiran 8.6. Bantuan

PETUNJUK PENGGUNAAN MODUL PEMBELAJARAN PENGINDERAAN VISUAL

1. Buka aplikasi Media Pembelajaran Penginderaan Visual.
2. Tunggu beberapa saat sampai muncul tampilan seperti pada Gambar 1.



Gambar 1. Tampilan Jendela Utama Aplikasi

Keterangan:

- 1) **Pengaturan**: untuk mengatur parameter yang akan dijalankan oleh aplikasi.
- 2) **Bantuan**: untuk melihat panduan penggunaan aplikasi
- 3) **Tentang**: untuk melihat profil pembuat aplikasi.

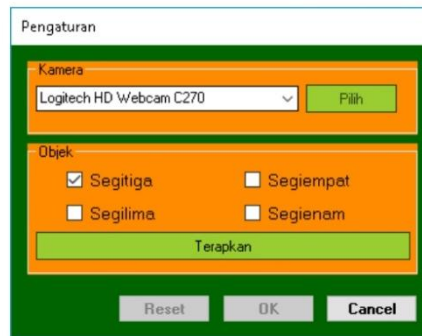
Dikarenakan aplikasi baru pertama dijalankan, maka terlebih dahulu melakukan pengaturan parameter dari aplikasi dengan menekan tombol **Pengaturan**.

3. Akan muncul tampilan jendela Pengaturan yang terlihat seperti pada Gambar 2. Langkah pertama dalam pengaturan adalah mengatur sensor kamera yang digunakan.



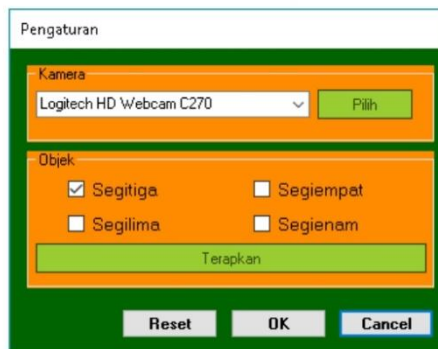
Gambar 2. Pengaturan sensor kamera

4. Pengaturan selanjutnya adalah objek bersegi yang akan dideteksi. Objek yang dipilih minimal 1 jenis dan maksimal boleh semua dipilih. Pada Gambar 3 dipilih objek Segitiga saja.



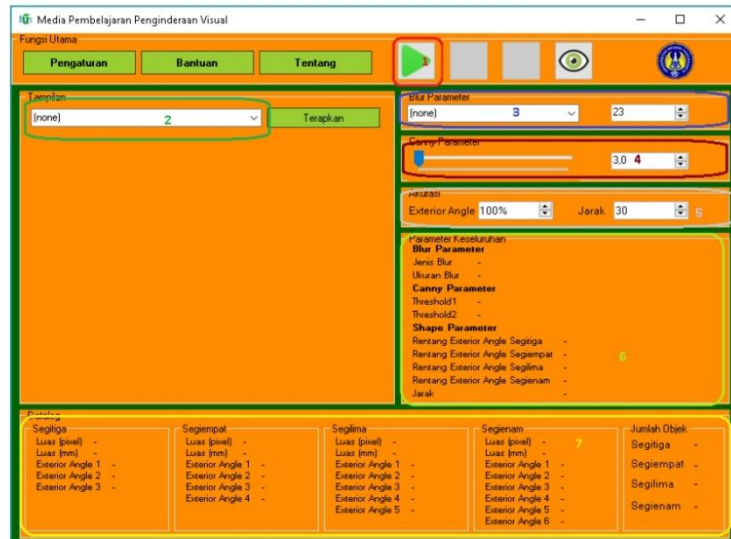
Gambar 3. Pengaturan objek yang ingin dideteksi

5. Pastikan lagi apakah pengaturan sudah sesuai dengan apa yang diinginkan, jika belum maka gunakan tombol *Reset* untuk mengatur ulang semua pengaturan, namun jika sudah maka gunakan tombol *OK* untuk kembali ke Jendela Utama aplikasi.



Gambar 4. Tombol pilihan Ulang dan OK

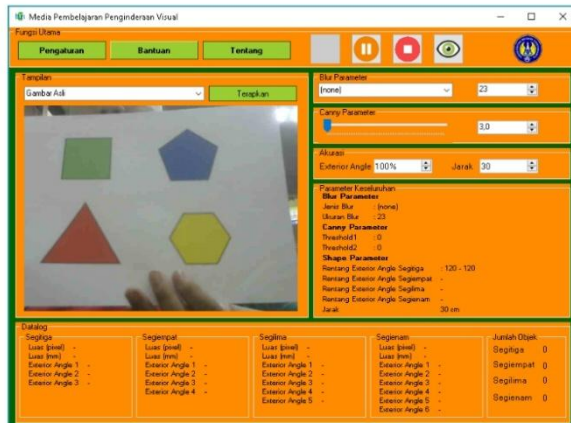
6. Kembali lagi ke tampilan utama aplikasi. Setelah melakukan pengaturan, tombol *Play* pada Fungsi Utama akan aktif seperti yang terlihat pada Gambar 5.



Gambar 5. Tampilan utama setelah melakukan pengaturan

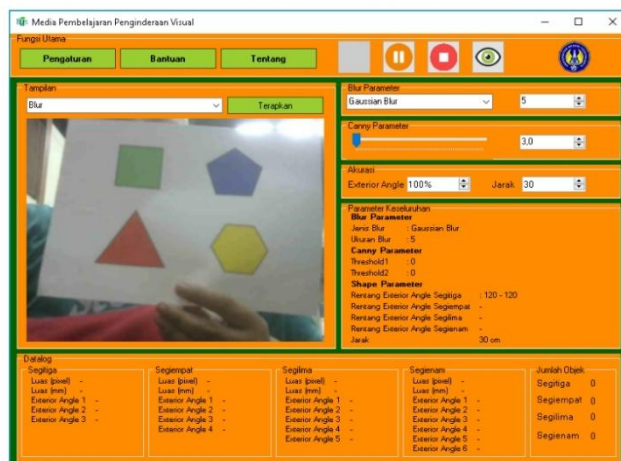
Keterangan:

- 1) **Tombol Play**: untuk memulai pengambilan citra.
 - 2) **Dropdown tampilan**: memilih tampilan citra yang diproses, jangan lupa tekan tombol Terapkan.
 - 3) **Pengaturan Blur**: pengaturan parameter *blur* seperti metode *blur* yang digunakan serta ukuran *blur*-nya.
 - 4) **Pengaturan Canny**: pengaturan parameter *canny*, disini fokus pada pengaturan nilai *Threshold1* dan juga *Threshold2*.
 - 5) **Pengaturan Keakuratan**: pengaturan tingkat keakuratan dalam mendeteksi *Exterior Angle* dari bentuk yang ingin dideteksi serta jarak kamera dengan objek.
 - 6) **Tampilan data pengaturan**: menampilkan ringkasan semua parameter.
 - 7) **Tampilan data dari pengolahan citra**: menampilkan data-data hasil pengolahan citra.
- Tekan tombol *Play* untuk memulai proses pengambilan citra oleh kamera.
7. Pada menu *dropdown* Tampilan, akan memberikan pilihan citra apa yang ingin dilihat.



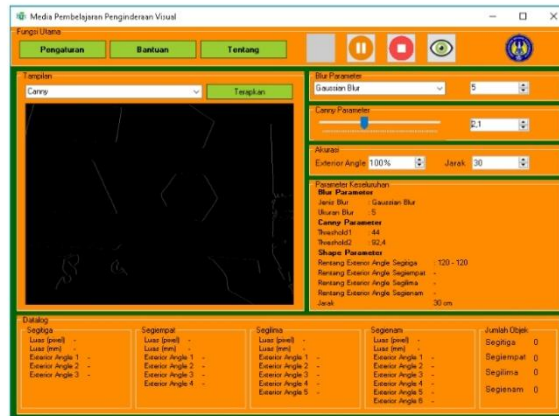
Gambar 6. Pengaturan tampilan citra yang diinginkan

8. Untuk pengaturan *Blur*, gunakan *dropdown* untuk memilih jenis *Blur* yang akan digunakan, lalu ubah nilai pada pilihan *UpDown* untuk memilih nilai *Blur* yang diinginkan.



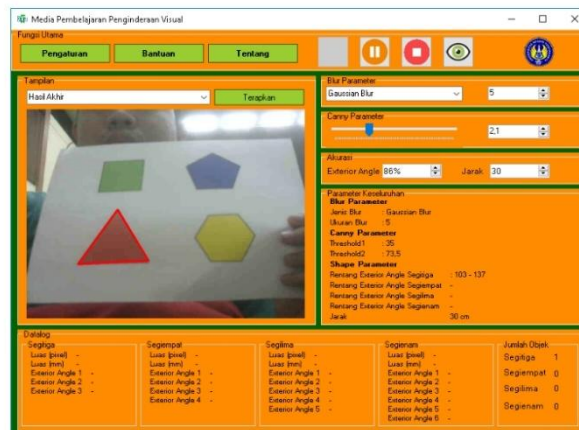
Gambar 7. Pengaturan *Blur* yang akan digunakan

9. Pada saat melakukan pengaturan parameter *Canny*, *trackbar* yang ada mewakili nilai *Threshold1*, sedangkan *UpDown* mewakili nilai pengali untuk mendapatkan nilai *Threshold2*. Contoh pengaturannya bisa dilihat pada Gambar 8.



Gambar 8. Contoh pengaturan parameter Canny

- Setelah semua pengaturan parameter sudah dilakukan, atur *dropdown* pada Tampilan menjadi Hasil Akhir untuk melihat objek yang terdeteksi. Jika objek belum bisa terdeteksi dengan baik, nilai Akurasi bisa dikurangi sampai objek dapat terdeteksi seperti pada Gambar 9. Semua nilai parameter dapat dilihat pada kolom Parameter Keseluruhan.

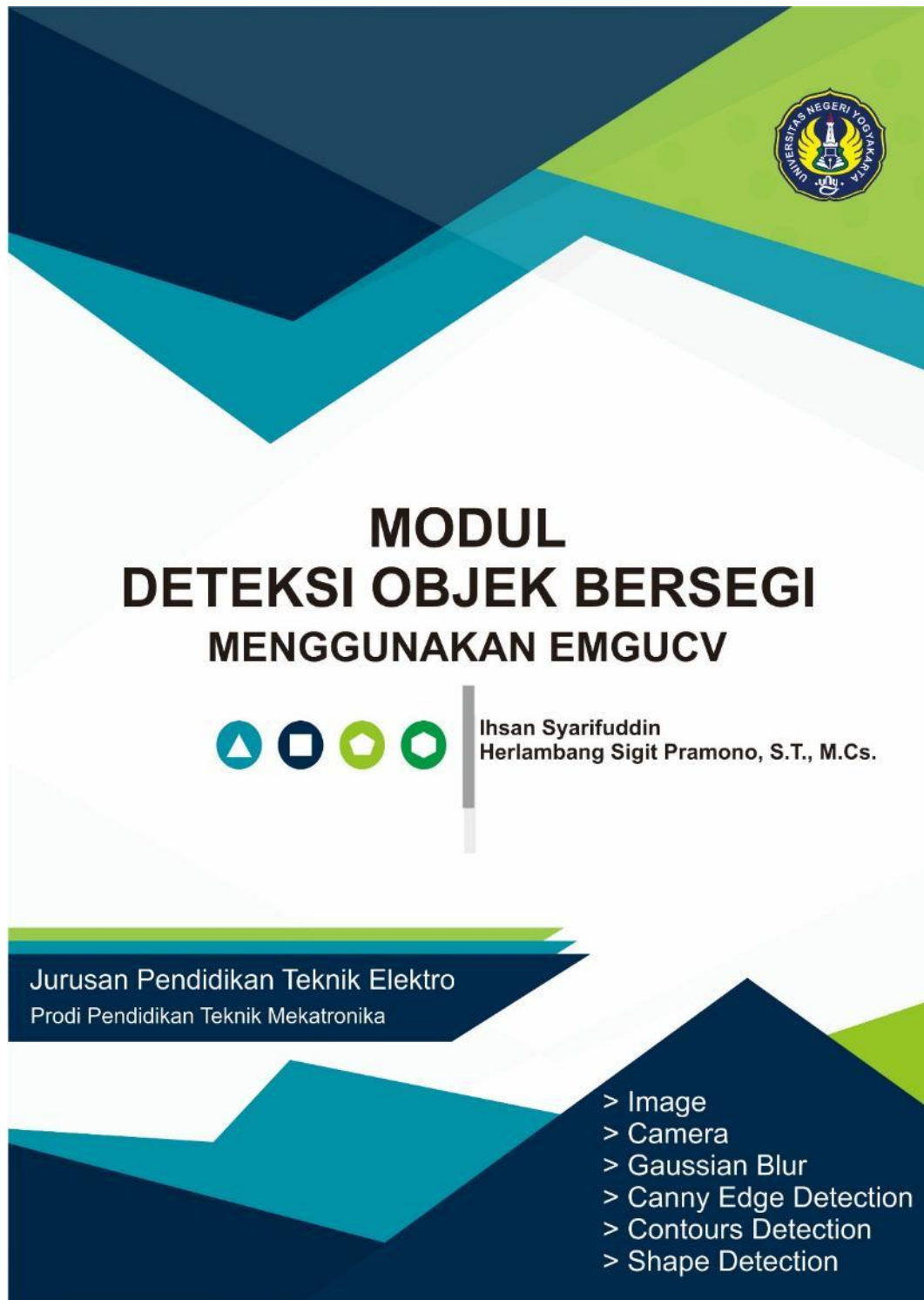


Gambar 9. Menggunakan nilai Accuracy 86%

Lampiran 9 Modul dan *Labsheet*

Lampiran 9.1. Modul *Computer Vision*

Lampiran 9.2. *Labsheet*



KATA PENGANTAR

Visual Robot? Siapa yang tak kenal dengannya. Mungkin beberapa golongan orang awam masih belum mengenalnya. *Visual robot* merupakan salah satu bidang komponen dari dunia robot yang mempelajari bagaimana indera penglihatan dari sebuah robot, layaknya mata dalam sistem penginderaan penglihat dari manusia. Di tengah zaman yang berkembang pesat ini, agar tak ketinggalan zaman, setidaknya kita tahu apa itu *visual robot*, dasar-dasar nya, atau mungkin kita bisa punya kompetensi menguasai pemrogramannya.

Pemrograman dasarnya yang mudah difahami memungkinkan *visual robot* masih mudah untuk dapat dipelajari di berbagai jenjang pendidikan, seperti sekolah menengah, diploma, sarjana, pascasarjana, pekerja, insinyur, profesor, dan masih banyak lagi pembelajaran *visual robot* oleh berbagai jenis kalangan masyarakat lainnya.

Nah, mungkin dengan hadirnya buku **Modul Deteksi Objek Bersegi Menggunakan EmguCV** ini bisa membantu para siswa, mahasiswa, guru, dosen, orang awam lain, dan semua yang mungkin membutuhkan agar bisa tahu apa itu visual robot, cara memrogramnya, dan pengaplikasiannya di dunia robot maupun industri.

Demikian, mudah-mudahan melalui buku ini, penulis dan para pembaca bisa menguasai dasar-dasar ilmu visual robot, mempraktikkan dan menerapkannya dalam kehidupan sehari-hari. Semoga apa yang kita pelajari membawa kebermanfaatan untuk segalanya. Sekian, Wassalamualaikum Warrahmatullahi Wabarakatuh.

Yogyakarta, Februari 2019

Penulis

DAFTAR ISI

KATA PENGANTAR	i
DAFTAR ISI	ii
BAB I	1
Computer Vision	1
1. Pengertian Computer Vision.....	1
2. Penerapan Computer Vision	2
BAB II	7
Instalasi EmguCV	7
1. Instalasi Visual Studio 2012	7
2. Instalasi EmguCV di <i>Windows</i>	9
3. Konfigurasi EmguCV di Visual Studio 2012.....	11
BAB III	16
Dasar-dasar <i>Computer Vision</i>	16
1. <i>Image</i>	16
2. <i>Camera</i>	20
3. <i>Smoothing</i>	22
4. <i>Canny Edge Detection</i>	26
5. <i>Contours</i>	29
BAB IV	34
Deteksi Segitiga, Segiempat, Segilima, dan Segienam	34
1. Aplikasi Pendeteksi Objek Bersegi.....	34
2. Deteksi Bentuk (<i>Shape Detection</i>).....	38

DAFTAR GAMBAR

Gambar 1. Pemeriksaan PCB berbasis computer vision.....	2
Gambar 2. Pemeriksaan label berbasis computer vision.....	3
Gambar 3. Pemeriksaan botol berbasis computer vision	3
Gambar 4. Robot ASIMO dari HONDA	4
Gambar 5. Computer Vision pada UAV	4
Gambar 6. Penglihatan buatan pada tunanetra	5
Gambar 7. Melihat aliran darah dengan Computer Vision	5
Gambar 8. Computer Vision pada traffic light.....	6
Gambar 9. Computer Vision dalam dunia militer.....	6

BAB I

Computer Vision

1. Pengertian Computer Vision

Kenneth Dawson & Howe (2014: 1) menjelaskan bahwa *Computer Vision* adalah sebuah analisis gambar dan video secara otomatis oleh komputer guna mendapatkan suatu pemahaman mengenai dunia. Namun, alasan kenapa kita berpikir bahwa *vision* itu mudah adalah karena kita memiliki sistem visual kita sendiri yang membuat tugas tersebut tampak intuitif bagi pikiran sadar kita. Namun sebenarnya, sistem visual manusia sangatlah kompleks dan bahkan perkiraan mengenai keterlibatan otak yang terlibat dalam pemrosesan visual sangatlah besar, yaitu dari 25% hingga lebih dari 50%.

Secara umum, tahapan dalam proses *Computer Vision* dapat dibagi menjadi 4, yaitu *Image Acquisition*, *Image Processing*, *Image Analysis*, dan *Image Understanding*.

a) *Image Acquisition*

Computer Vision menggunakan sensor visual untuk melakukan penangkapan data citra. Sensor visual yang digunakan pada umumnya adalah kamera. Hal ini sama dengan manusia yang merekam data dengan menggunakan mata, kemudian informasi visual tersebut diterjemahkan dan diolah oleh otak.

b) *Image Processing*

Pengolahan disini khususnya untuk membantu peningkatan dan perbaikan kualitas citra, sehingga dapat dianalisa dan diolah lebih jauh secara lebih efisien. Selain itu, pengolahan data citra akan meningkatkan kualitas data citra yang dapat berupa peningkatan level sinyal-sinyal terhadap *noise*.

c) *Image Analysis*

Dalam proses ini akan dilakukan eksplorasi beberapa karakteristik tertentu dari sebuah objek melalui data citra. Proses yang termasuk *Image Analysis* misalnya, pendeteksian tepi serta batasan objek dalam citra, baik

berdasarkan perbedaan warna, tingkat kecerahan, perbedaan kontras, bentuk, dan sebagainya.

d) *Image Understanding*

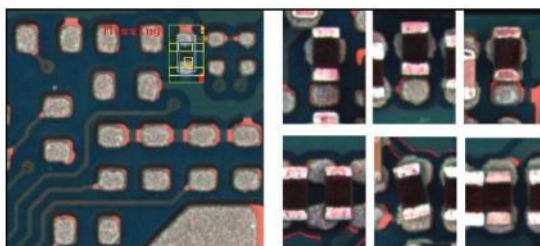
Proses ini merupakan tahapan terakhir dalam *Computer Vision*. Pada umumnya, tahap *Image Understanding* menggunakan teknik kecerdasan buatan misalnya *Artificial Neural Network*, *Fuzzy System*, *Evolutionary Algorithm*, dan yang paling mutakhir adalah *Deep Learning Network*. Selain itu, untuk interpretasi citra, juga bisa melakukan analisa secara stokastik, misalnya dengan menggunakan *Hidden Markov Model (HMM)* atau *Gaussian Mixture Model (GMM)*.

2. Penerapan Computer Vision

Sampai saat ini, dengan meningkatnya kemampuan dan kecepatan komputer komersial, *Computer Vision* sudah mulai diterapkan di berbagai bidang, baik dalam tahap prototipe maupun aplikasi nyata.

a) *Industri*

Computer Vision memiliki banyak sekali penerapan pada dunia industri, terutama pada pemeriksaan otomatis manufaktur suatu barang di setiap sektor produksi. Pada contohnya, *computer vision* digunakan pada pemeriksaan suatu PCB untuk memastikan jalur rangkaian serta komponen, pemeriksaan kualitas cetakan pada suatu label, serta pemeriksaan botol untuk memastikan bahwa semua botol sudah terisi dengan baik.



Gambar 1. Pemeriksaan PCB berbasis *computer vision*

Modul Deteksi Objek Bersegi Menggunakan EmguCV



Gambar 2. Pemeriksaan label berbasis *computer vision*



Gambar 3. Pemeriksaan botol berbasis *computer vision*

b) Robotika

Computer Vision banyak digunakan sebagai sensor visual pada sebuah robot. Robot ASIMO produk dari Honda menggunakan dua kamera sebagai mata yang diletakkan pada bagian kepala robot. ASIMO menggunakan *stereoscopic vision* dan algoritma *Computer Vision* untuk dapat melihat dan mengenali objek dan lingkungan sekitarnya.



Gambar 4. Robot ASIMO dari HONDA

Pengembangan *Computer Vision* di bidang robotika lainnya adalah pemanfaatan *Computer Vision* dalam kendaraan tanpa awak, misalnya kendaraan darat, kapal selam robot, dan pesawat udara tanpa awak. Robot-robot ini yang masuk dalam level otonom secara penuh menggunakan *Computer Vision* untuk melakukan navigasi atau bahkan memetakan lingkungan sekitar.



Gambar 5. *Computer Vision* pada UAV

c) Penglihatan Buatan

Pengembangan penglihatan buatan juga dilakukan dengan memanfaatkan *Computer Vision*. Pengembangan ini dilakukan untuk membantu tunanetra, sehingga mereka merasakan sensasi visual seperti orang normal. Pengembangan ini menggunakan kamera sebagai sensor yang selanjutnya akan diumpankan ke dalam otak melalui elektroda.

Modul Deteksi Objek Bersegi Menggunakan EmguCV

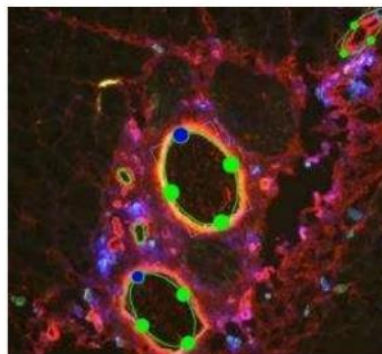
Bagian otak yang diumpangkan adalah otak bagian *visual cortex* yang berada di belakang kepala.



Gambar 6. Penglihatan buatan pada tunanetra

d) Medis dan Kedokteran

Salah satu pemanfaatan Computer Vision yang cukup bermanfaat adalah dalam bidang medis. Proses yang paling banyak digunakan adalah ekstraksi ciri, seperti microscopy images, x-ray images, angiography images, ultrasonic images, dan tomography images. Selain itu, Computer Vision juga digunakan untuk mengukur ukuran organ, aliran darah, dan melihat struktur otak.

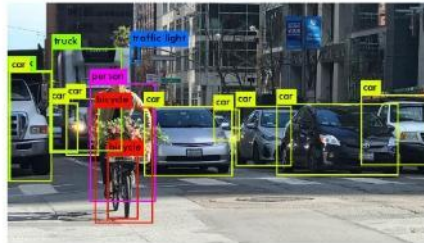


Gambar 7. Melihat aliran darah dengan *Computer Vision*

Modul Deteksi Objek Bersegi Menggunakan EmguCV

e) *Intelligent Transportation System (ITS)*

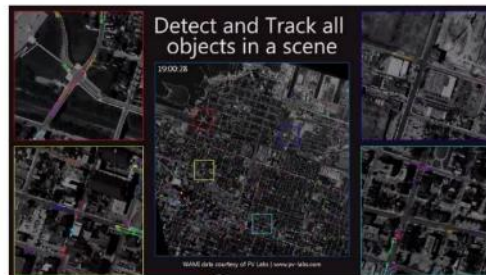
Bidang perlalulintasan mulai menggunakan berkembang teknologi-teknologi yang memanfaatkan teknologi Computer Vision. Alat-alat yang memanfaatkan teknologi Computer Vision antara lain alat pendeteksi kendaraan, penghitung kendaran, dan penghitung kecepatan kendaraan. Pemanfaatan alt-alat ini guna membangun sebuah pengaturan lalu lintas kendaraan secara cerdas.



Gambar 8. *Computer Vision* pada traffic light

f) *Sistem Senjata dan Militer*

Militer merupakan salah satu bidang yang paling banyak melakukan pengembangan teknologi canggih seperti *Computer Vision*. Salah satu penggunaan *Computer Vision* yang paling banyak digunakan di bidang militer adalah sebagai alat mendeteksi musuh serta alat kendali dan navigasi rudal jarak jauh.



Gambar 9. *Computer Vision* dalam dunia militer

BAB II

Instalasi EmguCV

Untuk melakukan praktik pemrograman Computer Vision dengan EmguCV, diperlukan beberapa instalasi dan konfigurasi sehingga memudahkan dalam melakukan praktik. Instalasi dan konfigurasi yang diperlukan diantaranya adalah instalasi Visual Studio 2012, instalasi EmguCV, serta konfigurasi EmguCV pada Visual Studio 2012.

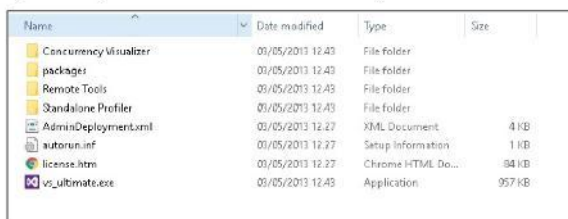
1. Instalasi Visual Studio 2012

a) Mengunduh *installer* Visual Studio 2012

Installer Visual Studio 2012 bisa diunduh melalui situs resmi dari Microsoft. Pilih Visual Studio 2012 versi *Ultimate*, atau bisa langsung diunduh melalui link http://download.microsoft.com/download/D/B/0/DB03922C-FF91-4845-B7F2-FC68595AB730/VS2012_ULT_enu.iso.

b) *Install* aplikasi Visual Studio

Setelah file *installer* sudah terunduh, file di *mount* menggunakan aplikasi UltraISO maupun PowerISO, buka file *vs_ultimate.exe* seperti yang terlihat pada Gambar 10 untuk memulai proses instalasi.



Name	Date modified	Type	Size
Concurrency Visualizer	01/05/2013 12:43	File folder	
packages	01/05/2013 12:43	File folder	
Remote Tools	01/05/2013 12:43	File folder	
Standalone Profiler	01/05/2013 12:43	File folder	
AdminDeployment.xml	01/05/2013 12:27	XML Document	4 KB
autorun.inf	01/05/2013 12:27	Setup Information	1 KB
license.htm	01/05/2013 12:27	Chrome HTML Do...	94 KB
vs_ultimate.exe	01/05/2013 12:43	Application	957 KB

Gambar 10. Isi dari *VS2012_ULT_enu.iso* setelah di *mount*

Pada awal proses instalasi, akan disuguhkan dengan tampilan dengan konten tempat dimana Visual Studio 2012 akan diinstal, dan tentu saja persetujuan penggunaan *software* seperti yang terlihat. Tekan tombol *Next*

Modul Deteksi Objek Bersegi Menggunakan EmguCV

setelah menentukan lokasi dimana *software* akan diinstal dan memberikan tanda *check* pada kotak persetujuan.



Gambar 11. Tampilan saat awal proses instalasi Visual Studio 2012.

Tampilan selanjutnya akan memperlihatkan fitur-fitur apa saja yang ingin diinstal pada Visual Studio 2012. Jika tidak mengetahui fungsi dari tiap fitur yang disediakan, maka lebih baik biarkan seperti *default*-nya dimana semua fiturnya dipilih untuk diinstal. Tekan tombol *Install* untuk memulai proses instalasi.



Gambar 12. Pemilihan fitur untuk Visual Studio 2012.

Modul Deteksi Objek Bersegi Menggunakan EmguCV

Proses instalasi akan memakan waktu sekitar 30 menit sampai 60 menit, tergantung dari spesifikasi dari komputer yang digunakan. Setelah proses instalasi selesai, akan muncul tampilan terakhir yang berisi pemberitahuan bahwa proses instalasi Visual Studio 2012 telah selesai. Disini hanya terdapat satu tombol yang berfungsi untuk langsung menjalankan *software* Visual Studio 2012. Dengan terbukanya *software* Visual Studio 2012, maka berarti proses instalasi telah berhasil dilakukan.



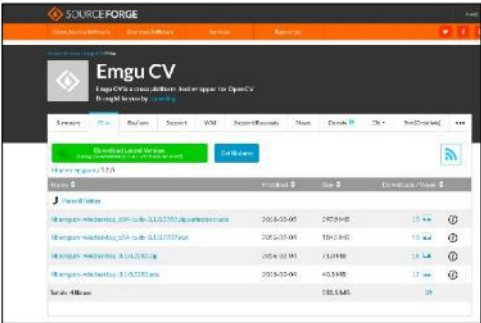
Gambar 13. Tampilan terakhir saat proses instalasi selesai.

2. Instalasi EmguCV di Windows

a) Mengunduh *library* EmguCV

Library bisa diunduh pada laman *Sourceforge* dengan link <https://sourceforge.net/projects/emguvc/files/emguvc/3.1.0/>, lalu pilih file *libemguvc-windesktop-3.1.0.2282.exe*.

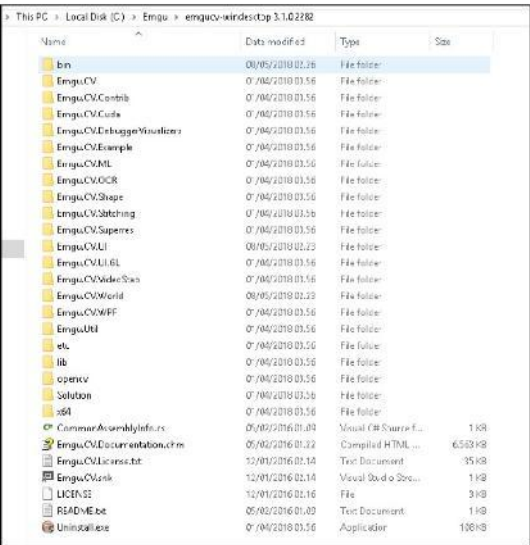
Modul Deteksi Objek Bersegi Menggunakan EmguCV



Gambar 14. Mengunduh *library* EmguC

b) Menginstal EmguCV

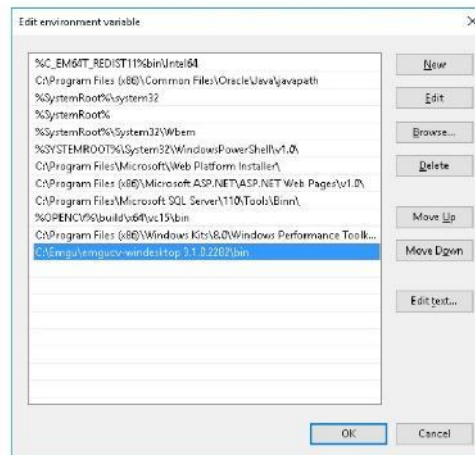
Buka file eksekusi dari *library* EmguCV yang sudah diunduh. Atur agar *library* EmguCV terinstal di partisi C. Setelah instalasi *library* selesai, *folder* dan *file* dari EmguCV dapat terlihat seperti pada Gambar 11.



Gambar 15. *Folder dan file dari library EmguC*

Modul Deteksi Objek Bersegi Menggunakan EmguCV

Setelah EmguCV berhasil terinstal dengan baik, selanjutnya adalah mengatur konfigurasi EmguCV pada sistem agar sistem *Windows* dapat membaca *library* EmguCV dengan baik. Untuk konfigurasi, buka *Windows Explorer* terlebih dahulu, lalu klik kanan pada *This PC* dan pilih *Properties*. Klik pada *Advanced system settings* di panel sebelah kiri. Buka *tab Advanced*, kemudian klik tombol *Environment Variables* di bagian bawah. Tambahkan alamat *folder bin* dari EmguCV ke dalam *System Variables Path*.

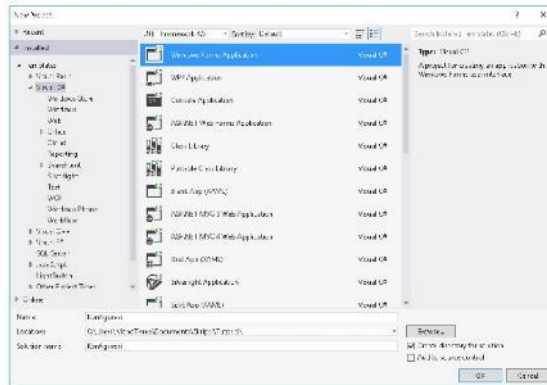


Gambar 16. Konfigurasi *Path* untuk *library* EmguCV

3. Konfigurasi EmguCV di Visual Studio 2012

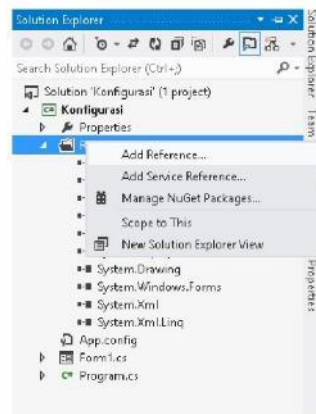
Untuk menggunakan EmguCV pada aplikasi C#, ada beberapa tahapan yang harus dilakukan. Buat *project* baru menggunakan bahasa *Visual C#* dengan jenis aplikasi *Windows Form Application*, lalu beri nama *project* dengan **Konfigurasi**.

Modul Deteksi Objek Bersegi Menggunakan EmguCV



Gambar 17. Pengaturan awal *project* baru

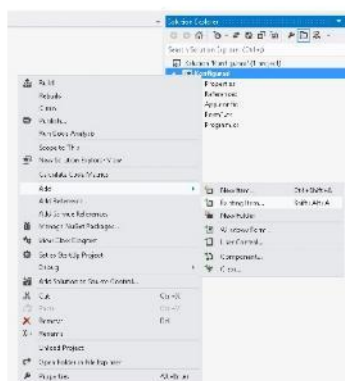
Masukkan beberapa file **.dll** ke dalam *project* sebagai *reference* dengan mengklik kanan **References** dan pilih **Add References**. Selanjutnya, klik **browse** untuk memilih file **.dll** dari EmguCV yang bertempat di **C:\Emgu\emgucv-windesktop 3.1.0.2282\bin**, yaitu **Emgu.CV.UI.dll** dan **Emgu.CV.World.dll**.



Gambar 18. Menambahkan *reference* yang akan digunakan

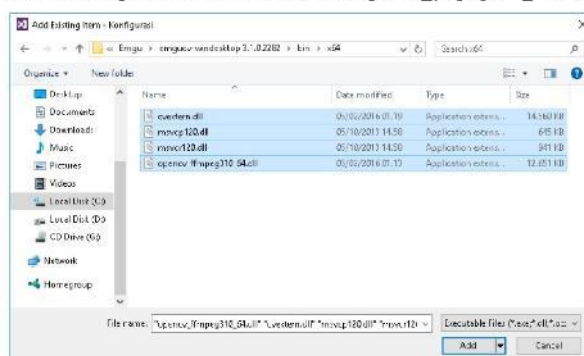
Modul Deteksi Objek Bersegi Menggunakan EmguCV

Masukkan juga beberapa file **.dll** yang berfungsi menjembatan OpenCV dengan **Visual C#**. Klik kanan pada nama *project*, pilih **Add**, lalu **Existing Item**.



Gambar 19. Cara memasukkan file tambahan

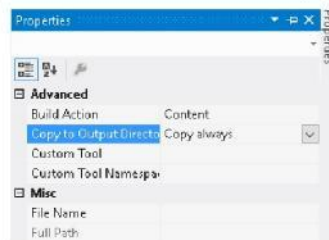
Masuk ke direktori `C:\Emgu\emgucv-windesktop 3.1.0.2282\bin\x64` untuk sistem 64-bit dan `C:\Emgu\emgucv-windesktop 3.1.0.2282\bin\x86` untuk sistem 32-bit. Ubah tipe file menjadi **Executable Files**, lalu pilih file `cvextern.dll`, `msvcp120.dll`, `msvcr120.dll`, dan `opencv_ffmpeg310 64.dll`.



Gambar 20. Menambahkan file *executable* ke *project*

Modul Deteksi Objek Bersegi Menggunakan EmguCV

Pilih semua *file executable* yang baru saja dimasukkan, lalu klik kanan dan pilih **Properties**. Ganti menjadi **Copy always** pada pilihan *Copy to Output Directory*. Hal ini dilakukan agar semua *file executable* tadi dapat langsung tersalin ke dalam *folder* keluaran secara otomatis saat *project* dijalankan.



Gambar 21. Mengganti ke *Copy always*

Ganti tipe *build* aplikasi yang ingin dibuat menjadi sesuai sistem, x86 untuk sistem 32-bit dan x64 untuk sistem 64-bit. Klik kanan pada nama *project*, lalu pilih **Properties**. Pindah ke tab **Build**, lalu ubah sesuai dengan sistem yang dipakai pada menu **Platform target**.

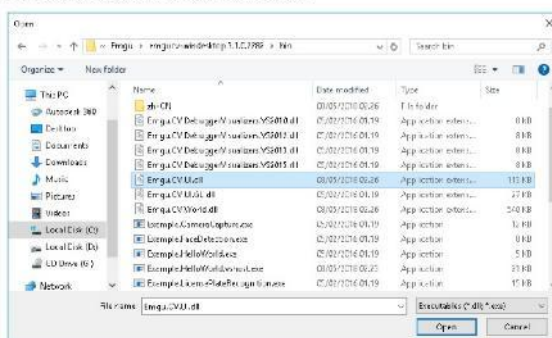


Gambar 22. Mengubah pengaturan *Platform target*

Setelah pengaturan sistem pada *project* sudah dilakukan, maka langkah terakhir yang harus dilakukan adalah menambahkan komponen UI EmguCV kedalam *Toolbox*. Klik kanan pada tab **General** pada *Toolbox*, lalu pilih **Choose Item**. Pada tab **.NET Framework Components**, pilih **Browse** untuk

Modul Deteksi Objek Bersegi Menggunakan EmguCV

mencari *library* UI untuk EmguCV. Masuk ke *C:\Emgu\emgucv-windesktop3.1.0.2282\bin*, lalu pilih *Emgu.CV.UI.dll*.



Gambar 23. Menambahkan komponen UI EmguCV

Setelah komponen berhasil ditambahkan kedalam *Toolbox*, maka akan muncul beberapa komponen UI baru yang berada pada tab **General** seperti *Pointer*, *HistogramBox*, *ImageBox*, *MatrixBox*, dan juga *PanAndZoomPictureBox*. Dari semua komponen tersebut, yang akan sering digunakan adalah *ImageBox*, dimana komponen tersebut digunakan untuk menampilkan gambar digital hasil olahan *computer vision* dari library *EmguCV*.

BAB III

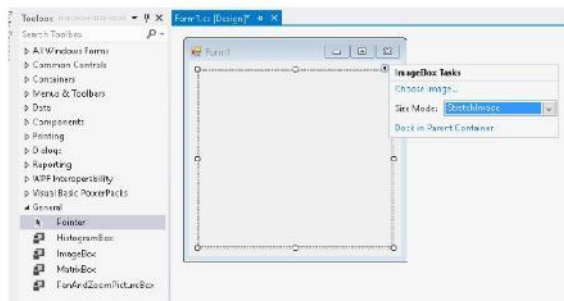
Dasar-dasar *Computer Vision*

Computer vision merupakan ilmu yang meniru sistem penglihatan manusia yang diaplikasikan pada sebuah komputer. Penglihatan manusia yang begitu kompleksnya dapat mengidentifikasi setiap objek yang ada disekitar, hal itulah yang mendasari pengembangan dari sebuah *computer vision*, namun karena hal itu juga yang membuat *computer vision* tidak mudah untuk dikuasai. Untuk memulai mempelajari *computer vision*, perlu diawali dari dasar-dasar dari *computer vision* itu sendiri.

1. *Image*

Menampilkan sebuah *image* atau gambar adalah suatu hal yang paling mendasar yang harus dikuasai ketika belajar *computer vision*. Untuk menampilkan sebuah *image*, maka diperlukan *path* dimana *image* tersebut berada, setelah itu *image* akan ditampilkan melalui *imageBox*.

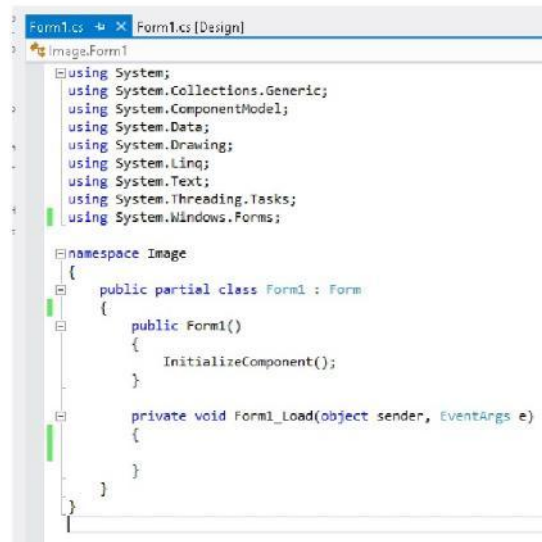
Buat sebuah *project* baru dengan nama *Image*, lalu desain GUI seperti yang terlihat pada Gambar 24 dan konfigurasi EmguCV.



Gambar 24. GUI untuk menampilkan *image*

Atur *Size Mode* dari *imageBox* menjadi *StretchImage* agar ukuran *image* yang akan ditampilkan menyesuaikan ukuran dari *imageBox*. *Double Click* pada *Form* untuk menambahkan *eventHandler Form_Load* serta masuk ke *code panel*.

Modul Deteksi Objek Bersegi Menggunakan EmguCV



Gambar 25. Code panel dan eventHandler Form_Load

Sebelum memulai proses pemrograman, sebaiknya dipahami terlebih dahulu fungsi `CvInvoke.Imread` yang terdapat pada EmguCV berikut:

```
public static Mat Imread(
    string filename,
    LoadImageType loadType
)
```

Listing 1. Fungsi `imread`

Dimana :

`filename` : nama beserta *path* dari *image* yang ingin ditampilkan.

`loadType` : tipe pemuatan *image*.

Variabel `filename` diisikan lengkap sampai ke *path* dari *image* yang dituju, perlu diingat bahwa pada EmguCV pembacaan *folder* menggunakan tanda garis miring kanan (/), bukan garis miring kiri (\). Untuk variabel `loadType`, memiliki beberapa tipe pemuatan yaitu *Unchanged*, *Grayscale*,

Modul Deteksi Objek Bersegi Menggunakan EmguCV

Color, *AnyDepth*, dan juga *AnyColor*. Tipe pemuatan *Unchanged* dan *Grayscale* akan memuat *image* dengan kualitas 8 bit, sedangkan tipe *Color* dan *AnyColor* akan memuat *image* dengan kualitas sama seperti bawaan *image*.

Berikut program lengkap yang digunakan untuk memuat *image* :

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

using Emgu.CV;
using Emgu.CV.CvEnum;
using Emgu.CV.Structure;
using Emgu.CV.Util;
using Emgu.Util;

namespace Image
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            Mat source =
                CvInvoke.Imread("C:/Users/MoboThree/Pictures/4292.jpg", LoadImageType.Color);
            imageBox1.Image = source;
        }
    }
}
```

Listing 2. Program lengkap untuk memuat *image*

Modul Deteksi Objek Bersegi Menggunakan EmguCV

Penjelasan mengenai program *Listing 2* adalah sebagai berikut :

```
using Emgu.CV;  
using Emgu.CV.CvEnum;  
using Emgu.CV.Structure;  
using Emgu.CV.Util;  
using Emgu.Util;
```

Digunakan untuk memasukkan *library* EmguCV kedalam program. Emgu.CV digunakan sebagai *wrapper* atau penerjemah dari fungsi-fungsi pada OpenCV. Emgu.CV.CvEnum berguna untuk enumerasi dari OpenCV. Emgu.CV.Structure berfungsi sebagai *wrapper* yang berisi struktur-struktur yang terdapat pada OpenCV. Emgu.CV.Util berisi kumpulan *utility* dari sistem *computer vision* untuk *project* EmguCV. Emgu.Util berisi kumpulan *utility* dari sistem dasar EmguCV.

```
Mat source =  
CvInvoke.Imread("C:/Users/MoboThree/Pictures/4292.jpg",  
LoadImageType.Color);  
imageBox1.Image = source;
```

Digunakan untuk menyimpan data *image* ke dalam sebuah variabel *source*, dimana *image* tersebut akan ditampilkan melalui *imageBox*. Fungsi *CvInvoke.Imread* berfungsi memuat data *image* dari suatu *path* beserta tipe warna pemuatan *image* yang digunakan, dimana *C:/Users/MoboThree/Pictures/4292.jpg* merupakan *path* dari *image* yang ingin ditampilkan, dan *LoadImageType.Color* merupakan tipe warna yang digunakan.



Gambar 26. Aplikasi penampil *image*

Untuk menjalankan *project*, bisa menggunakan tombol *Play* (▶) atau menekan tombol F5 pada *keyboard*. Jika aplikasi yang muncul nampak seperti Gambar 26, maka *project* untuk menampilkan *image* berhasil dibuat dengan baik.

2. Camera

Pengambilan data *image* dari sebuah kamera diperlukan untuk membuat sebuah aplikasi *computer vision* berbasis *realtime object*. Jenis kamera yang bisa teridentifikasi oleh EmguCV adalah kamera *webcam* bawaan laptop ataupun kamera *webcam* eksternal yang terkoneksi lewat USB. Meski begitu, ada beberapa jenis kamera lain yang dapat teridentifikasi oleh EmguCV melalui *3rd-party library* seperti kamera *kinect* dan juga *ip camera*.

```
namespace Webcam
{
    public partial class Form1 : Form
    {
        Mat source = new Mat();
        Capture cap = null;

        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            cap = new Capture(0);
            cap.ImageGrabbed += cap_ImageGrabbed;
            cap.Start();
        }

        void cap_ImageGrabbed(object sender, EventArgs e)
        {
            cap.Retrieve(source);
            imageBox1.Image = source;
        }
    }
}
```

Listing 3. Code untuk mengambil *image* dari *webcam*

Modul Deteksi Objek Bersegi Menggunakan EmguCV

Langkah-langkah yang dilakukan hampir sama seperti saat membuat project untuk memuat *image*, hanya *code* yang digunakan saja yang berbeda. *Code* yang digunakan untuk mengambil data kamera bisa dilihat pada *Listing 3*.

Penjelasan untuk program *Listing 3* adalah sebagai berikut :

```
Mat source = new Mat();  
Capture cap = null;
```

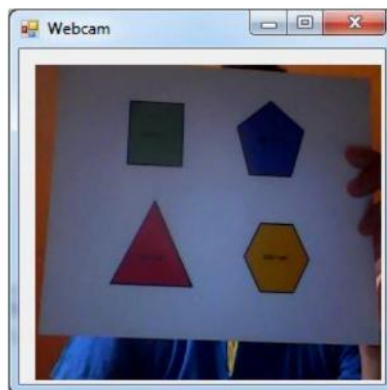
Berisi inisialisasi dari objek *Mat* dan juga *Capture*, dimana objek *Mat* adalah objek yang digunakan untuk tempat penyimpanan data *image*, sedangkan objek *Capture* merupakan objek yang digunakan untuk menangkap data *image* dari *webcam* secara *realtime*. *Mat source = new Mat()* berarti variabel *source* direferensikan sebagai objek *Mat* baru, sedangkan *Capture cap = null* berarti variabel *cap* diinisialisasikan tanpa memiliki referensi dari objek tertentu.

```
private void Form1_Load(object sender, EventArgs e)  
{  
    cap = new Capture(0);  
    cap.ImageGrabbed += cap_ImageGrabbed;  
    cap.Start();  
}
```

Merupakan suatu program yang dijalankan saat *eventHandler* *Form1_Load* terpanggil. Sesuai dengan namanya, *Form1_Load* hanya dijalankan satu kali, yaitu saat pertama kali *Form* dimuat. Program *cap = new Capture(0)* digunakan untuk mereferensikan variabel *cap* sebagai objek *Capture* baru dengan parameter *index 0*. Bagian *cap.ImageGrabbed += cap_ImageGrabbed* digunakan untuk melakukan pengambilan data *image* secara terus-menerus melalui fungsi *cap_ImageGrabbed*. Setelah semuanya sudah sesuai, maka proses pengambilan data *image* dapat dimulai dengan perintah *cap.Start()*. Yang perlu diperhatikan disini adalah jika ada lebih dari satu *webcam* tersambung ke laptop, tidak selamanya *webcam* bawaan berada pada *index 0*.

```
void cap_ImageGrabbed(object sender, EventArgs e)
{
    cap.Retrieve(source);
    imageBox1.Image = source;
}
```

Merupakan suatu program yang berfungsi untuk melakukan pengambilan data *image* secara terus-menerus setelah perintah `cap.Start()` terpanggil. Program `cap.Retrieve(source)` berfungsi untuk melakukan pemindahan data *image* dari objek *Capture* bervariasi `cap` ke objek *Mat* bervariasi `source`. Setelah itu, `source` ditampilkan ke `imageBox1`. Ketika *project* dijalankan, maka keluaran akan tampak seperti pada Gambar 27.



Gambar 27. Aplikasi penampil data *image* kamera

3. *Smoothing*

Smoothing yang dimaksudkan disini adalah proses pemburaman citra dengan tujuan menghilangkan *noise-noise* yang ada. Pemburaman *image* didapatkan dengan cara menggabungkan *image* yang ada dengan kernel *low-pass filter*. Proses pemburaman sebenarnya menghilangkan konten berfrekuensi tinggi dari gambar seperti *noise* dan juga tepian, sehingga akan didapatkan *image* yang sedikit terburamkan.

Gaussian Blur merupakan salah satu metode *smoothing* yang sering digunakan untuk memburamkan *noise* yang ada pada *image*. Fungsi *Gaussian Blur* pada EmguCV bisa dilihat pada *Listing 4*.

```
public static void GaussianBlur(  
    IInputArray src,  
    IOutputArray dst,  
    Size ksize,  
    double sigmaX,  
    double sigmaY = 0,  
    BorderType borderType = BorderType.Reflect101  
)
```

Listing 4. Fungsi GaussianBlur pada EmguCV

Dimana :

src : Masukan *image*

dst : Keluaran *image*

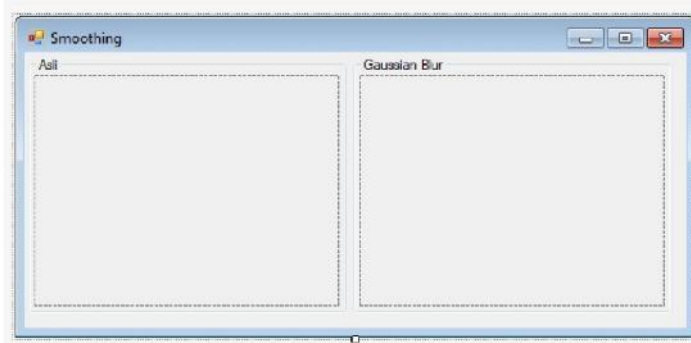
ksize : Ukuran inti *Gaussian*, harus bernilai positif dan juga ganjil

sigmaX : Inti *Gaussian* standar deviasi dalam sumbu X

sigmaY(optional) : Inti *Gaussian* standar deviasi dalam sumbu Y

borderType(optional) : Metode ekstrapolasi piksel

Buat sebuah *project* baru, lalu konfigurasi EmguCV pada *project* yang baru saja dibuat. Buat tampilan GUI sesuai pada Gambar 28.



Gambar 28. Tampilan GUI untuk *smoothing project*

Code yang digunakan untuk melakukan proses *smoothing* menggunakan metode *GaussianBlur* bias dilihat pada *Listing 5*.

```
namespace Smooth
{
    public partial class Form1 : Form
    {
        Mat source = new Mat();
        Mat smooth = new Mat();
        Capture cap = null;

        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            cap = new Capture();
            cap.ImageGrabbed += cap_ImageGrabbed;
            cap.Start();
        }

        void cap_ImageGrabbed(object sender, EventArgs e)
        {
            cap.Retrieve(source);
            CvInvoke.GaussianBlur(source, smooth, new Size(9,
            9), 0, 0);

            imageBox1.Image = source;
            imageBox2.Image = smooth;
        }
    }
}
```

Listing 5. Code penggunaan fungsi GaussianBlur

Untuk program *smoothing* pada *Listing 5* memiliki dasar program berupa program pengambilan data dari kamera secara *realtime*. Program hanya ditambahkan pada bagian *cap_ImageGrabbed*, dimana penambahan hanya sebatas fungsi *CvInvoke.GaussianBlur* yang digunakan untuk memburamkan *image*. Untuk deklarasi variabel juga ditambah dengan *Mat*

Modul Deteksi Objek Bersegi Menggunakan EmguCV

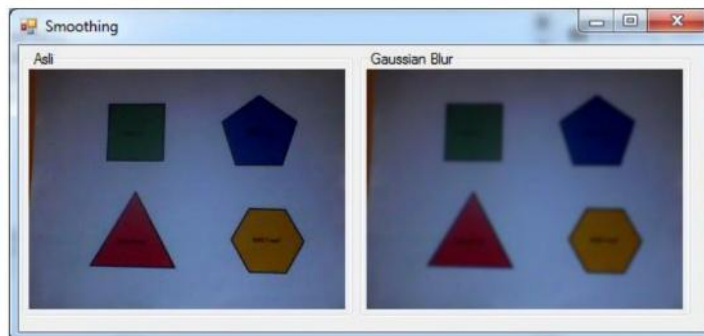
`smooth = new Mat()`, yang digunakan untuk tempat penyimpanan data *image* setelah melalui proses *Gaussian Blur*.

```
CvInvoke.GaussianBlur(source, smooth, new Size(9, 9), 0, 0);
```

Parameter pertama adalah *image* masukan, dan parameter kedua adalah *image* keluaran, serta yang paling penting adalah parameter ketiga, yaitu ukuran kernel yang digunakan saat proses *Gaussian Blur*, dimana nilai kernel disini haruslah ganjil dan minimal bernilai 3. Hasil *smoothing* dengan nilai kernel yang berbeda bisa di lihat pada Gambar 29 dan Gambar 30.



Gambar 29. Hasil *smoothing* dengan nilai kernel 9×9



Gambar 30. Hasil *smoothing* dengan nilai kernel 19×19

4. Canny Edge Detection

Pada tahun 1986, John F.Canny mengembangkan operator pendeteksi tepian. Operator tersebut merupakan algoritma pengolahan *image* secara bertahap yang dapat digunakan dalam pengolahan *image* digital secara luas. Algoritma *Canny* memberikan cara untuk mengolah data piksel menjadi sebuah garis tepian, dimana dapat disebut juga sebagai *edge*. Dua buah ambang batas histeresis digunakan untuk membuat bentuk *edge* dari data piksel yang ada. Fungsi *Canny* pada EmguCV bisa dilihat pada Listing 6.

```
public static void Canny(
    IInputArray image,
    IOutputArray edges,
    double threshold1,
    double threshold2,
    int apertureSize = 3,
    bool L2Gradient = false
)
```

Listing 6. Fungsi *Canny* pada EmguCV

Dimana :

image : Masukan *image*

edges : Keluaran *image*

threshold1 : Ambang batas pertama

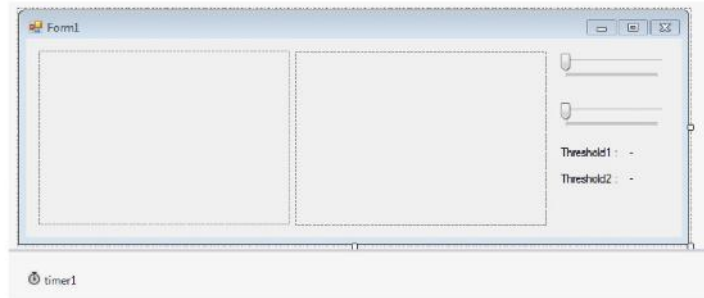
threshold2 : Ambang batas kedua

apertureSize(optional) : Parameter bukaan untuk *Sobel operator*

L2Gradient(optional) : Digunakan untuk menentukan norma yang digunakan

Buat sebuah *project* baru dengan konfigurasi EmguCV, lalu buat tampilan GUI seperti pada Gambar 31. Dua *trackbar* digunakan untuk mengatur nilai *threshold1* dan juga *threshold2*, beserta label yang digunakan untuk menampilkan nilai dari masing-masing *trackbar*. Atur nilai *maximum* semua *trackbar* menjadi 255, lalu tetapkan nilai awal untuk *trackbar1* sebesar 0 dan *trackbar2* sebesar 255. *Timer* digunakan untuk *update* nilai dari semua *trackbar* secara terus-menerus dengan selang waktu yang terdapat pada *Properties*, yaitu 1 ms.

Modul Deteksi Objek Bersegi Menggunakan EmguCV



Gambar 31. Pengaturan GUI untuk *Canny project*

Code yang digunakan untuk algoritma *Canny* adalah sebagai berikut:

```
namespace canny
{
    public partial class Form1 : Form
    {
        Mat source = new Mat();
        Mat canny = new Mat();
        Capture cap = null;
        int threshold1, threshold2;

        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            cap = new Capture();
            cap.ImageGrabbed += cap_ImageGrabbed;
            cap.Start();
        }

        void cap_ImageGrabbed (object sender, EventArgs e)
        {
            cap.Retrieve(source, 0); CvInvoke.Canny(source,
            canny, threshold1, threshold2);
            imageBox1.Image = source;
            imageBox2.Image = canny;
        }
    }
}
```

Listing 7. Code sederhana penggunaan *Canny*


```
private void timer1_Tick(object sender, EventArgs e)
{
    threshold1 = trackBar1.Value;
    threshold2 = trackBar2.Value;

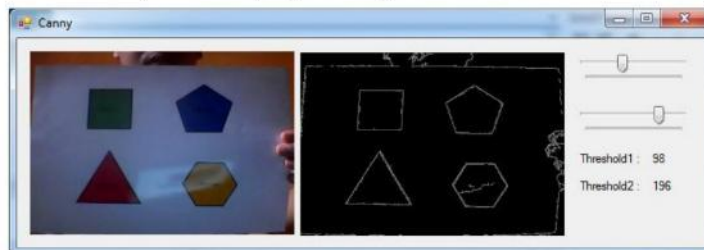
    label3.Text = threshold1.ToString();
    label4.Text = threshold2.ToString();
}
}
```

Listing 7(Lanjutan). Code sederhana penggunaan Canny

Pada Listing 7 di atas, dua hal yang perlu diperhatikan yaitu penggunaan *Timer* untuk meng-*update* nilai variabel *threshold1* yang berperan sebagai *Threshold1* dan juga *threshold2* yang berperan sebagai *Threshold2*, serta penggunaan *CvInvoke.Canny* pada fungsi *cap_ImageGrabbed*.

```
CvInvoke.Canny(source, canny, threshold1, threshold2);
```

Nilai dari *threshold1* dianjurkan memiliki nilai yang lebih kecil daripada nilai *threshold2*, dimana perlu diingat bahwa perbandingan antara *lowThreshold* dengan *highThreshold* adalah antara 1:2 sampai dengan 1:3. Contoh hasil algoritma *Canny* dapat dilihat pada Gambar 32.



Gambar 32. Hasil algoritma *Canny* dengan perbandingan 1:2

5. *Contours*

Pada dasarnya, *contour* digunakan setelah proses pendeteksian tepi berhasil dilakukan. Dengan menggunakan *Canny Edge Detection* didapatkan batas tepian pada tiap *pixel*, namun proses tersebut tidak dapat digunakan untuk mengkombinasikan tiap segmen yang menjadi sebuah bentuk tertentu. Sebelum memulai menggunakan metode *Contour* ini, perlu dipahami terlebih dahulu arti dari *contour* itu sendiri. Arti dari *contour* bisa berbeda tergantung situasi yang ada, namun pengertian umumnya yaitu *contour* merupakan sekumpulan *pixel* atau *point* yang merepresentasikan sebuah garis lengkung,

```
public static void FindContours(
    IInputOutputArray image,
    IOutputArray contours,
    IOutputArray hierarchy,
    RetrType mode,
    ChainApproxMethod method,
    Point offset = null
)
```

dalam artian struktural, pada sebuah *image*. Pada EmguCV, *contour* dapat dicari menggunakan fungsi *FindContours*, dengan rincian seperti yang terlihat pada *Listing 8*.

Listing 8. Fungsi FindContours pada EmguCV

Dimana :

image : Masukan *image* yang berkualitas 8-bit serta berbentuk *binary image*

contours : *Contour* yang terdeteksi dan disimpan dalam bentuk titik vektor

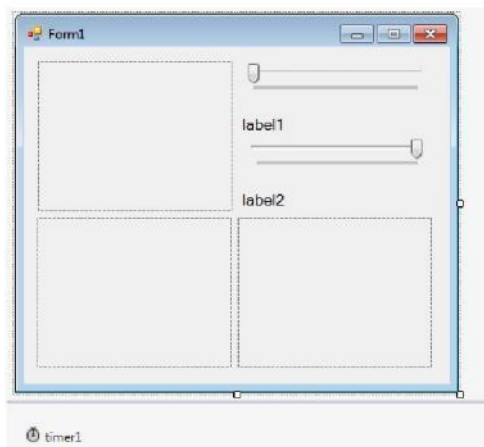
hierarchy : Keluaran vektor tambahan

mode : Mode pengambilan *contour*

method : Metode pendekatan

offset(optional) : Nilai *offset* dari setiap titik *contour* yang telah bergeser

Buat sebuah *project* baru beserta konfigurasi dengan EmguCV, lalu buat GUI seperti pada Gambar 33.



Gambar 33. Tampilan GUI untuk menampilkan *Contour*

Sama seperti materi sebelumnya, terdapat dua *trackbar* yang digunakan untuk mengatur parameter dari fungsi *canny*. Disini digunakan tiga buah *imageBox* dimana tiap *imageBox* digunakan untuk menampilkan *image* dari *webcam*, hasil dari proses *canny*, dan yang terakhir adalah menampilkan hasil dari proses metode *contour* yang paling sederhana, yaitu menampilkan tiap *contour* yang terdeteksi. *Timer* digunakan untuk memperbarui nilai dari tiap *trackbar* yang kemudian ditampilkan melalui dua buah komponen *label*. Berikut *code* yang digunakan untuk mencari dan menampilkan *contour*:

```
namespace contour
{
    public partial class Form1 : Form
    {
        Mat source = new Mat();
        Mat canny = new Mat();
        Mat output = new Mat();

        Capture cap = null;
        int threshold1, threshold2;
    }
}
```

Listing 9. *Code* untuk mendeteksi serta menampilkan *contour*

```

public Form1()
{
    InitializeComponent();
}

private void Form1_Load(object sender, EventArgs e)
{
    cap = new Capture();
    cap.ImageGrabbed += cap_ImageGrabbed;
    cap.Start();
}

private void timer1_Tick(object sender, EventArgs e)
{
    threshold1 = trackBar1.Value;
    threshold2 = trackBar2.Value;
    label1.Text = threshold1.ToString();
    label2.Text = threshold2.ToString();
}

void cap_ImageGrabbed(object sender, EventArgs e)
{
    cap.Retrieve(source, 0);
    output = source.Clone();

    CvInvoke.Canny(source, canny, threshold1,
        threshold2);

    using (VectorOfVectorOfPoint contours = new
        VectorOfVectorOfPoint())
    {
        CvInvoke.FindContours(canny, contours, null,
            RetrType.External,
            ChainApproxMethod.ChainApproxSimple);
        int count = contours.Size;
        for (int i = 0; i < count; i++)
        {
            CvInvoke.DrawContours(output, contours, i, new
                Bgr(Color.Red).MCvScalar, 3,
                LineType.EightConnected, null, 0, new Point(0,
                    0));
        }
    }

    imageBox1.Image = source;
    imageBox2.Image = canny;
    imageBox3.Image = output;
}

```

Listing 9(Lanjutan). Code untuk mendeteksi serta menampilkan *contour*

Modul Deteksi Objek Bersegi Menggunakan EmguCV

Bagian yang perlu diperhatikan adalah proses yang dilakukan setelah fungsi `CvInvoke.Canny`, dimana bagian tersebut merupakan awal dari proses pendeteksian *contour*.

```
using (VectorOfVectorOfPoint contours = new  
VectorOfVectorOfPoint())
```

Digunakan untuk menginisialisasi variabel `contours` yang direferensikan sebagai objek `VectorOfVectorOfPoint` baru, dan variabel tersebut digunakan selama proses pendeteksian *contour* saja.

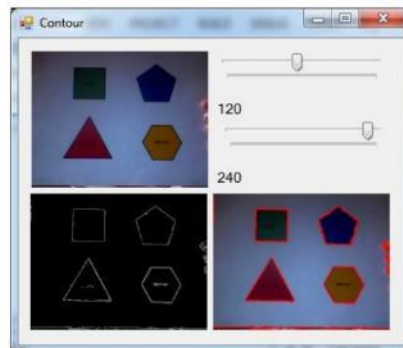
```
CvInvoke.FindContours(canny, contours, null,  
RetrType.External, ChainApproxMethod.ChainApproxSimple)
```

Digunakan untuk menemukan semua *contour* yang ada pada masukan `canny`, yang kemudian keluarannya akan disimpan dalam variabel `contours`. Parameter ketiga merupakan tempat dimana semua hirarki dari setiap *contour* yang terdeteksi akan disimpan, dapat bernilai `null`, dimana berarti hirarki tidak akan disimpan di variabel apapun. `RetrType` adalah parameter yang digunakan untuk menentukan metode pendeteksian *contour* yang ada, sedangkan `ChainApproxMethod` adalah parameter untuk menentukan metode pengambilan titik-titik dari *contour* yang sudah terdeteksi.

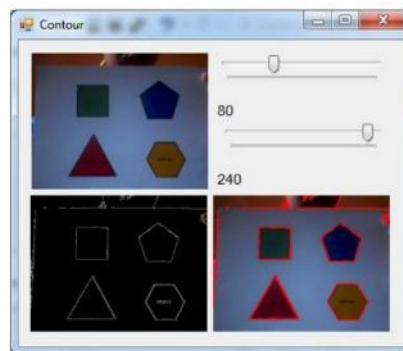
```
CvInvoke.DrawContours(output, contours, i, new  
Bgr(Color.Red).MCvScalar, 3, LineType.EightConnected, null,  
0, new Point(0, 0))
```

Merupakan program yang digunakan untuk menampilkan tiap *contour* yang terdeteksi. Parameter pertama adalah variabel dimana setiap *contour* akan ditampilkan. Variabel `contours` mewakili semua *contour* yang terdeteksi, sedangkan `i` mewakili *index* dari tiap *contour* yang ingin ditampilkan secara bertahap. Pada `new Bgr(Color.Red).MCvScalar`, berguna untuk mengatur warna yang akan digunakan, dan parameter selanjutnya berguna sebagai pengatur ketebalan dari garis yang digunakan untuk menggambar semua *contour* yang ada. Pada Gambar 34 dan Gambar 35, bisa dilihat bahwa *contour*

yang dideteksi akan menyesuaikan dari hasil olahan *Canny Edge Detection*, dan ini menjadi salah satu bukti bahwa pengolahan *contour* merupakan salah satu proses yang dapat digunakan untuk menindaklanjuti hasil dari proses *Canny Edge Detection*. Pada Gambar 34, fungsi *FindContours* dengan menggunakan keluaran *Canny Edge Detection* dengan perbandingan 1:2, sedangkan pada Gambar 35, *FindContours* menggunakan keluaran *Canny Edge Detection* dengan perbandingan 1:3.



Gambar 34. Hasil *contour* dari proses *Canny Edge Detection* 1:2



Gambar 35. Hasil *contour* dari proses *Canny Edge Detection* 1:3

BAB IV

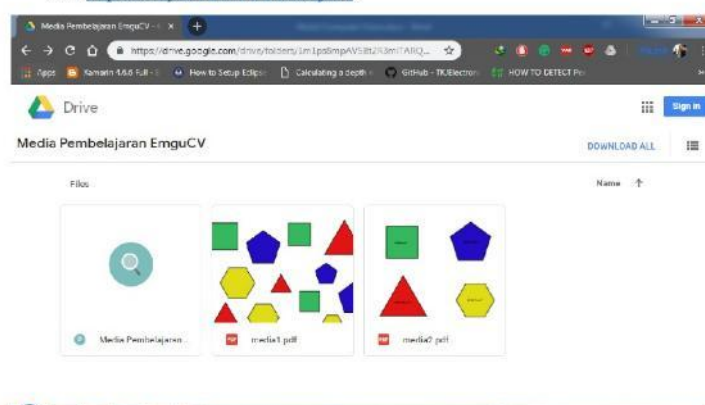
Deteksi Segitiga, Segiempat, Segilima, dan Segienam

Sistem penglihatan manusia tampaknya menggunakan berbagai jenis sumber informasi untuk mendeteksi dan mengenali sebuah objek. Pada tingkatan terendah pada suatu pengenalan objek, informasi mengenai tepian dan suatu bagian tertentu digunakan untuk mengekstraksi sebuah unit perseptual. Pada beberapa kasus, informasi dari batasan bentuk, seperti bentuk persegi kendaraan dalam suatu citra udara, pengenalan objek memiliki suatu peranan yang sangatlah penting.

1. Aplikasi Pendeteksi Objek Bersegi

Untuk mempermudah dalam memahami langkah-langkah untuk mendeteksi objek bersegi, seperti segitiga, segiempat, segilima, dan segienam, penulis sudah menyediakan media pembelajaran berupa aplikasi pendeteksi objek bersegi yang dapat diunduh melalui *Google Drive*.

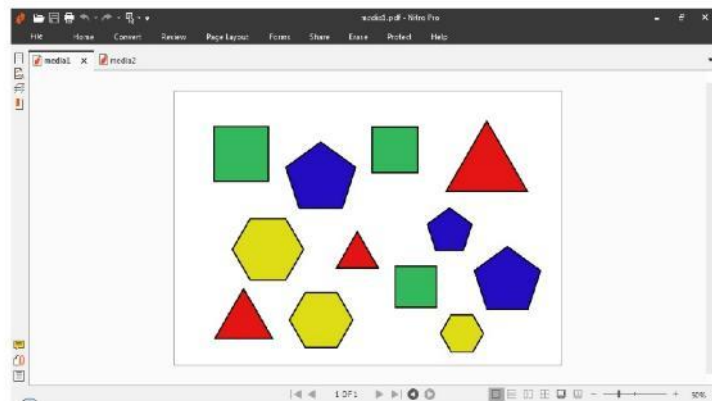
- a. Untuk media pembelajaran beserta objek yang dideteksi dapat diunduh pada link <https://tinyurl.com/sidedObject>.



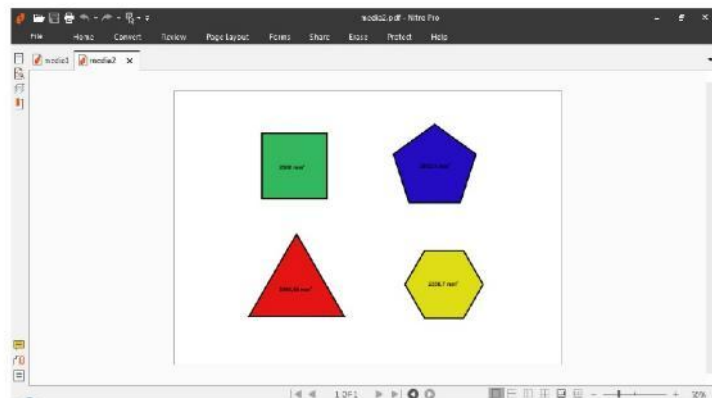
Gambar 36. Media pembelajaran yang ada di Google Drive

Modul Deteksi Objek Bersegi Menggunakan EmguCV

- b. Cetak terlebih dahulu objek yang akan dideteksi dalam *file media1.pdf* dan *media2.pdf* di kertas ivory 230 ukuran A4.



Gambar 37. Isi dari *file media1.pdf*



Gambar 38. Isi dari *file media2.pdf*

- c. Jalankan *installer* aplikasi dengan nama *file Media Pembelajaran EmguCV.exe*.

Modul Deteksi Objek Bersegi Menggunakan EmguCV

Folder	Name	Date modified	Type	Size
Desktop	db12-uninstall.exe	12/01/2018 17:09	Application	214,720 KB
Downloads	db12-uninstall.exe	27/09/2018 20:53	Application	154,384 KB
Recent Places	DirectXWebSetupTool64-7.5-Setup.exe	26/07/2018 23:19	Application	387,873 KB
Google Drive	DirectX64-7.5-Setup.exe	27/09/2018 23:18	Application	227,805 KB
	Line_Coder_Perc_4400_Full.exe	01/09/2018 15:52	Application	47,008 KB
Libraries				
Documents	DirectX-uninstall-3.1.0.2202.exe	27/09/2018 15:43	Application	18,322 KB
Music	Visual Basic	29/09/2018 20:00	Application	2,900 KB
Pictures	Media Burn.bundle Emagic V2.exe	01/05/2018 20:52	Application	77,889 KB
Videos	MGInstall.exe	09/02/2018 21:23	Application	59,320 KB
	MGInstall.exe	12/04/2018 16:30	Application	14,713 KB
Computers	MyCam_V2.0.exe	01/05/2018 20:48	Application	55,487 KB
Local Disk (C:)	NCP46-38108346-468-468-AIOS-FULL.exe	01/05/2018 05:55	Application	61,031 KB
Local Disk (D:)	NCPC46-Temping-Pack-468355555.exe	05/05/2018 20:30	Application	13,219 KB
CD Drive (G:)	NCPC46-Temping-Pack-468-468-AIOS-FULL.exe	14/03/2018 23:30	Application	68,259 KB
	NCPC46-Temping-Pack-468355555-FULL.exe	05/05/2018 21:04	Application	90,763 KB
Network	NCPC46-48301416-468-468-AIOS-FULL.exe	06/05/2018 13:22	Application	66,505 KB

Gambar 39. *File instalasi aplikasi*

- d. Langkah pertama proses instalasi adalah menginstal *library* EmguCV, hal ini guna mengatasi jika pada Laptop atau PC belum ada *library* EmguCV, dimana *library* tersebut sangat dibutuhkan oleh aplikasi agar dapat berjalan dengan baik.



Gambar 40. Proses instalasi *library* EmguCV

- e. Langkah selanjutnya adalah proses instalasi aplikasi pada sistem.



Gambar 41. Proses instalasi aplikasi Media Pembelajaran EmguCV

- f. Setelah proses instalasi selesai, maka pada tampilan *desktop* akan muncul *shortcut* untuk menjalankan aplikasi. Pastikan aplikasi dijalankan dengan pengaturan *Run As Administrator*.

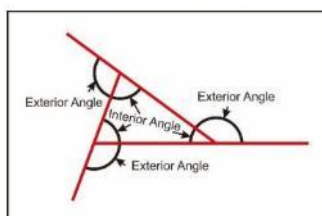


Gambar 42. *Run As Administrator* saat menjalankan aplikasi

2. Deteksi Bentuk (*Shape Detection*)

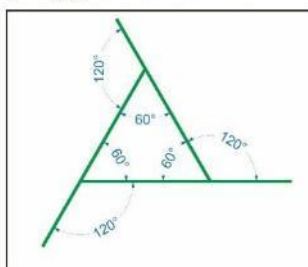
Mendeteksi suatu bentuk merupakan salah satu kegunaan dari metode *contour* yang sudah dipelajari sebelumnya. Cara yang paling sederhana untuk mendeteksi bentuk adalah dengan mencari sudut *exterior* antar garis yang bersinggungan. Garis-garis tersebut dapat dicari dengan menerapkan metode *contour*, yang sebelumnya telah dilakukan proses pengurangan *noise* menggunakan *GaussianBlur* dan juga metode *canny* untuk mencari garis tepi dari objek yang ingin dideteksi.

Sudut *exterior* merupakan sudut terluar dari setiap sisi pada bidang *polygon* terhadap perpanjangan sisi lain yang berdekatan, seperti yang terlihat pada Gambar 43.

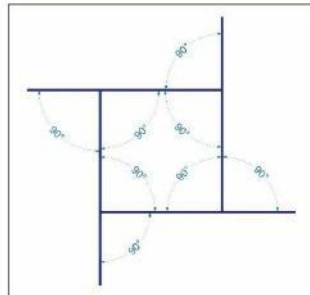


Gambar 43. Sudut *exterior* dan sudut *interior*

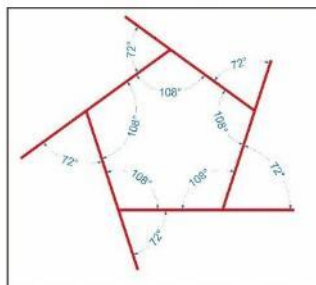
Untuk mendeteksi objek bersegi seperti segitiga, segiempat, segilima, dan juga segienam, maka harus diketahui terlebih dahulu besarnya sudut *exterior* dari masing-masing objek. Jika diasumsikan setiap sisi dari objek memiliki panjang yang sama, maka besarnya sudut *exterior* akan terlihat seperti pada Gambar 44, Gambar 45, Gambar 46, dan juga Gambar 47.



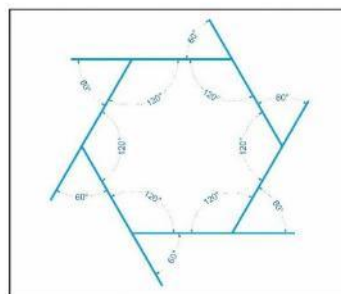
Gambar 44. Sudut *exterior* pada segitiga



Gambar 45. Sudut *exterior* pada segiempat



Gambar 46. Sudut *exterior* pada segilima



Gambar 47. Sudut *exterior* pada segienam

Modul Deteksi Objek Bersegi Menggunakan EmguCV

EmguCV menyediakan fungsi *GetExteriorAngleDegree* yang bisa digunakan untuk mencari sudut *exterior* dari dua garis yang bersinggungan. Fungsi tersebut berada pada *namespace Emgu.CV.Structure* dan pada *class LineSegment2D*. Penjelasan fungsi *GetExteriorAngleDegree* dapat dilihat pada *Listing 10* dan untuk penggunaannya dapat dilihat pada *Listing 11*.

```
public double GetExteriorAngleDegree(  
    LineSegment2D otherLine  
)
```

Listing 10. Fungsi GetExteriorAngleDegree pada EmguCV

Dimana :

otherLine : Garis lain yang bersinggungan

```
double angle = Math.Abs(line.GetExteriorAngleDegree(otherLine))
```

Listing 11. Penggunaan GetExteriorAngleDegree

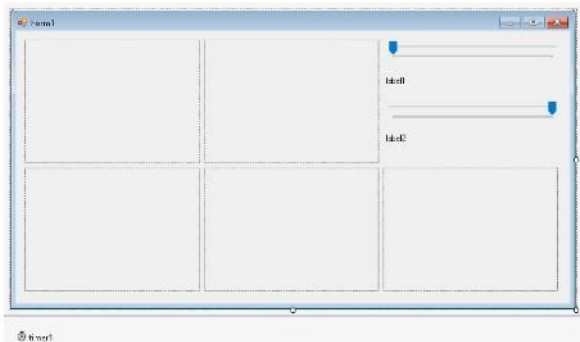
Dimana :

angle : Variabel untuk menyimpan hasil perhitungan sudut *exterior*

line : Garis pertama yang digunakan untuk mencari sudut *exterior*

otherLine : Garis lain yang bersinggungan dengan garis pertama

Buat sebuah *project* baru berbasis C#, buat GUI seperti yang terlihat pada Gambar 48.



Gambar 48. GUI untuk aplikasi deteksi objek bersegi

Modul Deteksi Objek Bersegi Menggunakan EmguCV

Terdapat 5 *imageBox* dimana masing-masing *imageBox* digunakan untuk menampilkan hasil dari tiap proses dalam pendeteksian objek bersegi, mulai dari *image* awala, *image* hasil *GaussianBlur*, *image* hasil *Canny Edge Detection*, *image* hasil *Contours*, dan tentu saja *image* hasil deteksi objek bersegi. Dua *trackbar* digunakan untuk mengatur nilai parameter *Threshold1* dan *Threshold2* untuk digunakan pada proses *Canny Edge Detection*.

Berikut *code* yang digunakan untuk pendeteksian objek bersegi:

```
public Form1()
{
    InitializeComponent();
}

private void Form1_Load(object sender, EventArgs e)
{
    cap = new Capture(0);
    cap.ImageGrabbed += cap_ImageGrabbed;
    cap.Start();
}

void cap_ImageGrabbed(object sender, EventArgs e)
{
    cap.Retrieve(source);
    draw = source.Clone();
    output = source.Clone();
    CvInvoke.GaussianBlur(source, blur, new Size(9, 9), 0);
    CvInvoke.Canny(blur, canny, threshold1, threshold2);
    using (VectorOfVectorOfPoint contours = new
    VectorOfVectorOfPoint())
    {
        CvInvoke.FindContours(canny, contours, null,
        RetrType.External, ChainApproxMethod.ChainApproxSimple);
        int count = contours.Size;
        for(int i = 0; i < count; i++)
        {
            CvInvoke.DrawContours(draw, contours, i, new
            Bgr(Color.Red).MCvScalar, 3, LineType.EightConnected, null,
            0, new Point(0, 0));
            using (VectorOfPoint contour = contours[i])
```

Listing 12. Code yang digunakan untuk pendeteksian objek bersegi

```

using (VectorOfPoint approx = new VectorOfPoint())
{
    CvInvoke.ApproxPolyDP(contour, approx,
        CvInvoke.ArcLength(contour, true) * 0.02, true);
    Point[] pts = approx.ToArray();
    if (CvInvoke.ContourArea(approx, false) > 500)
    {
        if (approx.Size == 4)
        {
            LineSegment2D[] edges = PointCollection.PolyLine(pts,
                true);
            bool shape = false;
            for (int j = 0; j < edges.Length; j++)
            {
                double angle = Math.Abs(edges[(j + 1) %
                    edges.Length].GetExteriorAngleDegree(edges[j]));
                if (angle < 80 || angle > 100)
                {
                    shape = false;
                    break;
                }
                else
                {
                    shape = true;
                }
            }

            if (shape == true)
            {
                CvInvoke.PolyLines(output, pts, true, new
                    Bgr(Color.Blue).MCvScalar, 3, LineType.EightConnected,
                    0);
            }
        }
    }
}

imageBox1.Image = source;
imageBox2.Image = blur;
imageBox3.Image = canny;
imageBox4.Image = draw;
imageBox5.Image = output;
}

```

Listing 12(Lanjutan). Code yang digunakan untuk pendeteksian objek bersegi

Modul Deteksi Objek Bersegi Menggunakan EmguCV

Penjelasan untuk *Listing 12* adalah sebagai berikut :

```
cap.Retrieve(source);  
draw = source.Clone();  
output = source.Clone();
```

Variabel *source* digunakan untuk menampung data *image* asli yang berasal dari *webcam*. Variabel *draw* dan *output* merupakan data *image* hasil duplikasi dari variabel *source*, dimana *draw* nantinya akan digunakan untuk menampilkan data *contour* yang terdeteksi, sedangkan *output* digunakan untuk menampilkan data dari objek bersegi yang terdeteksi. Data *image* dari variabel *source* diduplikasi agar nilai dari *source* tetap murni, tanpa olahan apapun, sehingga bisa dijadikan acuan ketika aplikasi dijalankan.

```
CvInvoke.GaussianBlur(source, blur, new Size(9, 9), 0);  
CvInvoke.Canny(blur, canny, threshold1, threshold2);
```

Sebelum masuk ke proses deteksi objek bersegi, terlebih dahulu harus melalui beberapa proses yang nantinya berguna untuk membantu proses deteksi objek menjadi lebih akurat. Untuk langkah sederhana yang bisa dilakukan adalah dengan mengaplikasikan *Gaussian Blur* untuk mengurangi *noise* tepian pada *image*, dan *Canny Edge Detection* untuk pendeteksian tepian pada *image*.

```
CvInvoke.FindContours(canny, contours, null, RetrType.External,  
ChainApproxMethod.ChainApproxSimple);
```

Merupakan program yang digunakan untuk mendapatkan tiap *contour* yang terdeteksi. Parameter pertama adalah variabel dimana setiap *contour* akan ditampilkan. Variabel *contours* mewakili semua *contour* yang terdeteksi, sedangkan *i* mewakili *index* dari tiap *contour* yang ingin ditampilkan secara bertahap. Pada `new Bgr(Color.Red).MCvScalar`, berguna untuk mengatur warna yang akan digunakan, dan parameter selanjutnya berguna sebagai pengatur ketebalan dari garis yang digunakan untuk menggambar semua *contour* yang ada.

```
CvInvoke.ApproxPolyDP(contour, approx, CvInvoke.ArcLength  
(contour, true) * 0.02, true);
```

Digunakan untuk mendapatkan kurva poligon dengan nilai keakuratan tertentu. Parameter pertama merujuk pada masukan vektor yang akan dideteksi garis kurvanya, sedangkan parameter kedua sebagai keluaran garis kurva yang sudah

Modul Deteksi Objek Bersegi Menggunakan EmguCV

terdeteksi. Parameter ketiga biasa disebut epsilon, digunakan untuk mengatur nilai akurasi dari pendeteksian, semakin kecil nilainya maka semakin banyak garis kurva yang bisa terdeteksi. Parameter terakhir pada fungsi `CvInvoke.ApproxPolyDP` memiliki kegunaan yang sama seperti parameter kedua dari fungsi `CvInvoke.ArcLength`, yaitu untuk menentukan apakah kurva yang dideteksi tertutup atau tidak.

```
if(CvInvoke.ContourArea(approx, false) > 500)
```

Digunakan untuk mencari luas dari suatu kurva. Parameter pertama merupakan masukan kurva yang ingin dihitung luasnya. Parameter kedua merujuk pada nilai keluaran akan berisi nilai asli, yang berarti bisa positif ataupun negative, atau akan bernilai mutlak positif. *Code* diatas bertujuan untuk memilah luas kurva tertentu, dimana jika luas kurva bernilai lebih dari 500 piksel, maka akan diproses lebih lanjut, jika tidak maka proses pengolahan *contour* akan berakhir.

```
double angle = Math.Abs(edges[(j + 1) % edges.Length].GetExteriorAngleDegree(edges[j]));
```

Digunakan untuk menghitung besarnya nilai sudut *exterior* dari tiap garis vektor yang terdeteksi, dimana hasil dari perhitungan akan dimasukkan ke dalam variabel *angle*. Variabel `edges[(j + 1) % edges.Length]` berisi garis vektor pada *index* selanjutnya, sedangkan `edges[j]` adalah garis vektor yang berada pada *index* saat fungsi dipanggil.

```
if (angle < 80 || angle > 100)
{
    shape = false;
    break;
}
else
{
    shape = true;
}
```

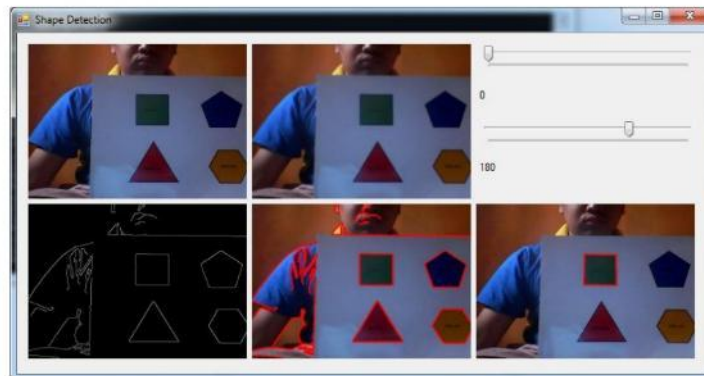
Merupakan sebuah klarifikasi apakah sudut *exterior* sudah sesuai dengan perhitungan ditambah dengan toleransi. Dikarenakan objek yang diinginkan adalah objek segiempat, dimana pengaturannya berada pada `approx.Size == 4` di proses sebelumnya, maka sudut *exterior* yang diinginkan adalah 90° ditambah dengan

Modul Deteksi Objek Bersegi Menggunakan EmguCV

toleransi 10° , maka didapat nilai minimal sudut *exterior* adalah 80° dan nilai maksimal sudut *exterior* adalah 100° . Jika salah satu sudut *exterior* tidak memenuhi syarat tersebut, maka *contour* tidak akan dianggap sebagai objek segiempat. Proses klarifikasi ini akan menghasilkan nilai boolean dimana *shape* = *false* jika objek bukan objek segiempat, dan *shape* = *true* jika objek dianggap objek segiempat.

```
if (shape == true)
{
    CvInvoke.Polylines(output, pts, true, new
    Bgr(Color.Blue).MCvScalar, 3, LineType.EightConnected, 0);
}
```

Digunakan untuk menggambar garis vektor yang terdeteksi sebagai objek segiempat jika syarat terpenuhi. Parameter pertama adalah data *image* yang akan digunakan sebagai keluaran. Parameter kedua merupakan data titik-titik koordinat dari objek yang terdeteksi. Parameter ketiga digunakan untuk membuat garis kurva yang digambarkan akan berupa garis kurva tertutup atau tidak. Parameter keempat digunakan untuk mengatur warna dari garis keluaran yang dihasilkan. Parameter kelima berfungsi untuk mengatur ketebalan dari garis kurva. Parameter keenam digunakan untuk mengatur tipe garis yang digunakan, dan parameter terakhir adalah untuk mengatur nilai pergeseran *bits*. Keluaran dari *code* dari *Listing 12* dapat dilihat pada Gambar 49.



Gambar 49. Hasil deteksi objek segiempat

Lampiran 9.2. *Labsheet*



	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Memuat Citra Digital	No. Job : 1
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

A. Kompetensi

Memahami cara kerja/prinsip memuat citra digital menggunakan EmguCV.

B. Indikator

1. Mahasiswa dapat membuat GUI untuk memuat citra digital.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk memuat citra digital.
3. Mahasiswa dapat memuat berbagai jenis citra digital menggunakan EmguCV.

C. Tujuan

1. Mahasiswa dapat membuat GUI untuk memuat citra digital dengan baik dalam waktu yang singkat.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk memuat citra digital dengan baik dan benar dalam waktu yang singkat.
3. Mahasiswa dapat memuat berbagai jenis citra digital menggunakan EmguCV dengan baik dan benar dalam waktu yang singkat.

D. Dasar Teori

1. Citra Digital

Menurut Darma Putra (2010: 19), citra digital merupakan sebuah larik (array) yang berisi nilai-nilai real maupun kompleks yang direpresentasikan dengan deretan bit tertentu. Suatu citra dapat didefinisikan sebagai fungsi $f(x,y)$ berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial, dan amplitudo f di titik koordinat (x,y) , ini dinamakan intensitas atau tingkat keabuan dari citra pada titik tersebut.

2. File Formats

a) JPEG

JPEG merupakan singkatan dari *Joint Photographic Experts Group*, dan menggunakan *file extension* berupa **.jpg**. Mayoritas penggunaan *file format* untuk citra digital adalah JPEG, hal ini dikarenakan *format* tersebut memiliki kompresi data yang sangat tinggi sehingga banyak digunakan untuk suatu postingan pada berbagai jenis

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Memuat Citra Digital	No. Job : 1
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

social media. Namun, ketika suatu citra digital diterapkan kompresi yang terlalu tinggi, maka kualitas yang dihasilkanpun akan berkurang drastis.

b) *PNG*

PNG merupakan singkatan dari *Portable Network Graphics*, dan merupakan sebuah *file format* yang sangat penting dalam lingkup *editing*. *PNG* dapat digunakan untuk membuat suatu citra digital yang memiliki latar belakang transparan sepenuhnya atau transparan sebagian. *Format* ini tidak akan mengurangi kualitas serta detail dari suatu citra digital secara drastis. Resolusi *file* yang dihasilkan juga relatif lebih besar, sehingga *PNG* sangat cocok untuk *image* yang berukuran kecil namun dengan detail yang cukup tinggi.

c) *GIF*

GIF merupakan singkatan dari *Graphics Interchange Format*, dan memiliki kualitas yang sangat mirip dengan *PNG*. Dengan *format* ini, seseorang bisa membuat sebuah animasi singkat untuk sebuah *website*. *Format* ini memiliki rentang warna yang sangat sedikit, oleh karena itu *format* ini tidak selalu cocok dengan berbagai jenis citra digital. Seperti *PNG*, *GIF* juga dapat membuat latar belakang menjadi transparan, namun *GIF* tidak mendukung untuk latar belakang transparan sebagian.

d) *BMP*

BMP merupakan suatu citra digital yang terbentuk dari sekumpulan piksel, oleh karena itu semakin besar ukuran citra digital semakin banyak penyimpanan yang dibutuhkan. Sebuah *BMP* memuat sebuah *file header*, yang didalamnya mengandung pengenalan *bitmap* (*bitmap identifier*), ukuran *file* (*file size*), lebar (*width*), tinggi (*height*), pilihan warna (*color options*), dan titik mulai data *bitmap* (*bitmap data starting point*). Setelah *file header*, juga terdapat data piksel *bitmap*, semuanya terdiri dari berbagai warna yang berbeda. Jika ada citra digital yang memiliki resolusi 640×480 piksel, maka data yang harus disimpan oleh sistem sebanyak 307.200 piksel, dan jika citra digital dengan resolusi 2072×2048 memiliki 6.291.456 piksel data yang harus disimpan.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Memuat Citra Digital	No. Job : 1
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

e) *TIFF*

TIFF merupakan singkatan dari *Tagged Image File Format*, dimana cukup terkenal kalangan fotografer dan juga desainer grafis. Data citra digital yang disimpan dalam *format TIFF* sangatlah bagus untuk pengolahan *post-processing* dimana data yang dihasilkan sama sekali tidak terkompresi. *TIFF* dapat digunakan untuk menghasilkan citra digital dalam *format* yang lain.

3. Memuat Citra Digital menggunakan EmguCV

Memuat sebuah citra digital adalah suatu hal yang paling mendasar yang harus dikuasai ketika belajar *computer vision*. Untuk menampilkan sebuah citra digital, maka diperlukan *path* dimana citra digital tersebut berada, setelah itu citra digital akan ditampilkan melalui *imageBox*. Untuk memuat sebuah citra digital, EmguCV menyediakan fungsi `CvInvoke.Imread`, berikut penggunaan dari fungsi `CvInvoke.Imread` :

```
public static Mat Imread(
    string filename,
    LoadImageType loadType
)
```

Dimana :

`filename` : nama beserta *path* dari citra digital yang ingin ditampilkan
`loadType` : tipe pemuatan citra digital

Variabel `filename` diisikan lengkap sampai ke *path* dari citra digital yang dituju, perlu diingat bahwa pada EmguCV pembacaan *folder* menggunakan tanda garis miring kanan (`/`), bukan garis miring kiri (`\`). Untuk variabel `loadType`, memiliki beberapa tipe pemuatan yaitu *Unchanged*, *Grayscale*, *Color*, *AnyDepth*, dan juga *AnyColor*. Tipe pemuatan *Unchanged* dan *Grayscale* akan memuat citra digital dengan kualitas 8 bit, sedangkan tipe *Color* dan *AnyColor* akan memuat citra digital dengan kualitas sama seperti bawaan citra digital.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Memuat Citra Digital	No. Job : 1
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

E. Alat dan Bahan

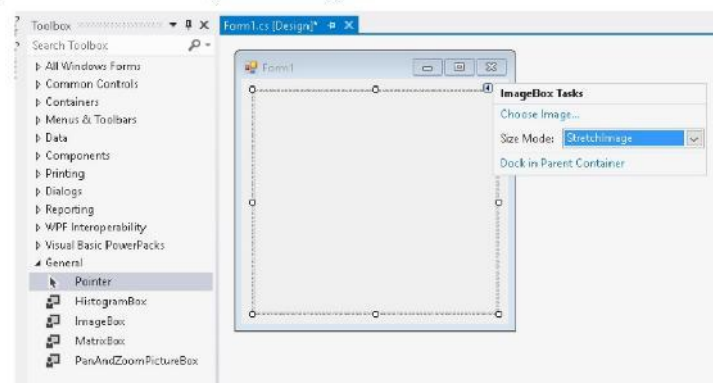
1. Laptop / Desktop PC.
2. Library EmguCV versi 3.1.0.2282.
3. Software Microsoft Visual Studio 2012.
4. Citra digital yang ingin dimuat.

F. Keselamatan Kerja

1. Jauhkan peralatan yang tidak dibutuhkan dari meja kerja.
2. Ikuti prosedur yang ada dengan teliti sesuai dengan langkah kerja yang tersedia.
3. Penggunaan laptop / desktop PC dengan langkah-langkah yang benar dan baik pada saat menyalakan ataupun mematikannya.

G. Langkah Kerja

Buat sebuah *project* baru dengan nama *Image*, lalu desain GUI seperti yang terlihat pada Gambar 1 dan konfigurasi EmguCV.



Gambar 1. GUI untuk menampilkan citra digital

Atur **Size Mode** dari *imageBox* menjadi **StretchImage** agar ukuran citra digital yang akan ditampilkan menyesuaikan ukuran dari *imageBox*. *Double Click* pada **Form** untuk menambahkan *eventHandler Form_Load* serta masuk ke *code panel*.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Memuat Citra Digital	No. Job : 1
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

Masukkan *code* berikut untuk memuat sebuah citra digital:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

using Emgu.CV;
using Emgu.CV.CvEnum;
using Emgu.CV.Structure;
using Emgu.CV.Util;
using Emgu.Util;

namespace Image
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            Mat source =
                CvInvoke.Imread("C:/Users/MoboThree/Pictures/4292.jpg",
                    LoadImageType.Color);
            imageBox1.Image = source;
        }
    }
}
```

Listing 1. Code untuk memuat sebuah citra digital

Untuk penjelasan dari *Listing 1*, silahkan dilihat pada Modul Deteksi Objek Bersegi Menggunakan EmguCV pada Bab 3 Subbab *Image*.

H. Pengujian

1. Pengujian dilakukan dengan menjalankan *project* yang telah dibuat.
2. Amati perbedaan serta persamaan dari tiap tipe *LoadImageType*.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Memuat Citra Digital	No. Job : 1
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

3. Isikan hasil pengamatan dengan ketentuan memuat 5 citra digital yang berbeda ke dalam tabel berikut.

Tabel pemuatan citra digital dengan berbagai tipe *LoadImageType*.

No	Citra Asli	<i>LoadImageType</i>				
		<i>Unchanged</i>	<i>Grayscale</i>	<i>Color</i>	<i>AnyColor</i>	<i>AnyDepth</i>
1						
2						
3						
4						
5						
Kesimpulan :						

I. Tugas

Muatlah 5 jenis citra digital dengan 5 jenis *file format* yang berbeda, serta tiap satu citra dimuat dengan *LoadImageType* yang berbeda pula. Amati persamaan serta perbedaan yang terjadi terhadap pemuatan 5 jenis *file format* citra digital. Isikan hasil yang keluar dengan ketentuan sesuai pada tabel berikut:

	<i>JPEG</i>	<i>PNG</i>	<i>BMP</i>	<i>GIF</i>	<i>TIFF</i>
<i>Unchanged</i>					
<i>Grayscale</i>					
<i>Color</i>					
<i>AnyColor</i>					
<i>AnyDepth</i>					
Kesimpulan :					

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Mengakses Webcam	No. Job : 2
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

A. Kompetensi

Memahami cara kerja/prinsip mengakses *webcam* menggunakan EmguCV.

B. Indikator

1. Mahasiswa dapat membuat GUI untuk mengakses *webcam*.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk mengakses *webcam*.
3. Mahasiswa dapat mengakses *webcam* dengan resolusi yang berbeda menggunakan EmguCV.

C. Tujuan

1. Mahasiswa dapat membuat GUI untuk mengakses *webcam* dengan baik dalam waktu yang singkat.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk mengakses *webcam* dengan baik dan benar dalam waktu yang singkat.
3. Mahasiswa dapat mengakses *webcam* dengan resolusi yang berbeda menggunakan EmguCV dengan baik dan benar dalam waktu yang singkat.

D. Dasar Teori

1. Kamera Webcam

Kamera *webcam* adalah kamera perekam citra yang menggunakan teknologi USB 2.0 atau USB 3.0 untuk transfer data citra. Kamera *webcam* didesain agar mempermudah antarmuka dengan suatu sistem komputer dengan menggunakan teknologi USB yang sama seperti yang terdapat pada komputer pada umumnya. Kecepatan akses teknologi USB yang ada pada sistem komputer saat ini adalah 480 Mb/s untuk USB 2.0 dan pada USB 3.0 memiliki kecepatan transfer 5 Gb/s.

2. Webcam Resolution

Kamera *webcam* dikenal karena biaya pembuatannya yang rendah serta fleksibilitasnya yang tinggi, membuat *webcam* menjadi alat yang paling murah untuk digunakan pada *videoconference*. Meskipun memiliki harga yang murah, resolusi yang ditawarkan cukuplah mengesankan, pada *webcam* yang termasuk kategori *low-end*

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Mengakses Webcam	No. Job : 2
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

menawarkan resolusi 320×240 , resolusi 640×480 pada webcam kategori sedang, dan pada webcam kategori *high-end* menawarkan resolusi 1280×720 bahkan ada yang sampai 1920×1080 .

3. Mengakses Webcam Menggunakan EmguCV

Pengambilan data citra digital dari sebuah kamera diperlukan untuk membuat sebuah aplikasi *computer vision* berbasis *realtime object*. Jenis kamera yang bisa teridentifikasi oleh EmguCV adalah kamera webcam bawaan laptop ataupun kamera webcam eksternal yang terkoneksi lewat USB. Meski begitu, ada beberapa jenis kamera lain yang dapat teridentifikasi oleh EmguCV melalui *3rd-party library* seperti kamera *kinect* dan juga *ip camera*. Untuk mengakses webcam, EmguCV menyediakan *class* yang bisa digunakan untuk melakukan pengambilan data *image* secara *realtime*, dan *class* tersebut adalah *Capture*. Namun meskipun data citra digital sudah didapatkan, citra yang terekam tidak bisa ditampilkan secara langsung pada *imageBox*, namun harus menggunakan fungsi *Retrieve* terlebih dahulu. Berikut adalah penggunaan pada *class* *Capture* dan juga penggunaan untuk fungsi *Retrieve*:

```
public Capture(
    int camIndex
)
```

Dimana :

`camIndex` : indeks kamera webcam yang digunakan

```
public virtual bool Retrieve(
    IOutputArray image,
    int flag = 0
)
```

Dimana :

`image` : keluaran citra digital

`flag(optional)`: tipe pengubahan

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Mengakses <i>Webcam</i>	No. Job : 2
	Revisi : 01	Menggunakan <i>EmguCV</i>	4 × 50 menit

Resolusi citra digital yang dihasilkan dari *class* adalah resolusi standar bawaan kamera. Untuk mengubah resolusi dari citra digital yang terekam *webcam*, maka bisa menggunakan metode *SetCaptureProperty*. Metode *SetCaptureProperty* digunakan sebelum proses pengambilan citra dari *class* *Capture* dimulai.

```
cap.SetCaptureProperty(CapProp.FrameWidth, width);
cap.SetCaptureProperty(CapProp.FrameHeight, height);
```

Dimana :

width : lebar resolusi yang diinginkan

height : tinggi resolusi yang diinginkan

Perlu diperhatikan, bahwa resolusi maksimal yang dapat digunakan pada setiap kamera *webcam* adalah berbeda. Jika maksimal resolusi kamera *webcam* yang digunakan adalah 640×480 , maka resolusi yang digunakan untuk pengambilan data citra digital tidak akan lebih dari resolusi tersebut.

E. Alat dan Bahan

1. Laptop / Desktop PC.
2. Kamera *webcam*.
3. Library *EmguCV* versi 3.1.0.2282.
4. Software Microsoft Visual Studio 2012.

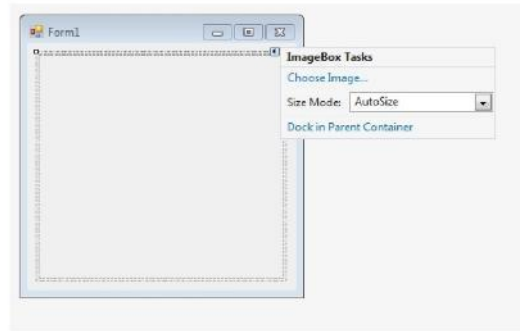
F. Keselamatan Kerja

1. Jauhkan peralatan yang tidak dibutuhkan dari meja kerja.
2. Ikuti prosedur yang ada dengan teliti sesuai dengan langkah kerja yang tersedia.
3. Penggunaan laptop / desktop PC dengan langkah-langkah yang benar dan baik pada saat menyalakan ataupun mematikannya.

G. Langkah Kerja

Buat sebuah *project* baru dengan nama *Webcam*, lalu desain GUI seperti yang terlihat pada Gambar 1 dan konfigurasi *EmguCV*.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Mengakses <i>Webcam</i>	No. Job : 2
	Revisi : 01	Menggunakan <i>EmguCV</i>	4 × 50 menit



Gambar 1. GUI untuk menampilkan citra dari *webcam*

Atur **Size Mode** dari *imageBox* menjadi **AutoSize** agar ukuran citra digital yang akan ditampilkan menyesuaikan ukuran dari *imageBox*. **Double Click** pada **Form** untuk menambahkan **eventHandler Form_Load** serta masuk ke **code panel**.

Masukkan **code** berikut untuk mengakses citra digital dari *webcam* :

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using Emgu.CV;
using Emgu.CV.CvEnum;
using Emgu.CV.Structure;
using Emgu.CV.Util;
using Emgu.Util;

namespace Webcam
{
    public partial class Form1 : Form
    {
        Mat source = new Mat();
        Capture cap = null;
    }
}
```

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Mengakses <i>Webcam</i>	No. Job : 2
	Revisi : 01	Menggunakan <i>EmguCV</i>	4 × 50 menit

```

int width = 640;
int height = 480;
public Form1()
{
    InitializeComponent();
}
private void Form1_Load(object sender, EventArgs e)
{
    cap = new Capture();
    cap.ImageGrabbed += cap_ImageGrabbed;
    cap.SetCaptureProperty(CapProp.FrameWidth, width);
    cap.SetCaptureProperty(CapProp.FrameHeight, height);
    cap.Start();
}
void cap_ImageGrabbed(object sender, EventArgs e)
{
    cap.Retrieve(source);
    imageBox1.Image = source;
}
}
}

```

Listing 1. Code untuk memuat sebuah citra digital

Untuk penjelasan dari *Listing 1*, silahkan dilihat pada Modul Deteksi Objek Bersegi Menggunakan *EmguCV* pada Bab 3 Subbab *Camera*.

H. Pengujian

1. Pengujian dilakukan dengan menjalankan *project* yang telah dibuat.
2. Amati hasil pemuatan setelah *project* dijalankan.
3. Isikan hasil pengamatan dengan ketentuan pengaturan resolusi kamera *webcam* yang berbeda.

Tabel kamera *webcam* dengan berbagai ukuran resolusi.

No	Resolusi	Hasil
1		
2		
3		
4		
Kesimpulan :		

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Mengakses <i>Webcam</i>	No. Job : 2
	Revisi : 01	Menggunakan <i>EmguCV</i>	4 × 50 menit

I. Tugas

Ubahlah ukuran dari *imageBox* dan juga *Form* menjadi ukuran yang tetap (disarankan menggunakan ukuran yang besar, misalnya 1280×720), lalu jalankan *project* dengan pengaturan resolusi yang berbeda. Amati hasil keluaran lalu isikan data pada tabel berikut :

No	Ukuran <i>imageBox</i>	Resolusi	Hasil
1			
2			
3			
4			
Kesimpulan :			

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Gaussian Blur</i>	No. Job : 3
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

A. Kompetensi

Memahami cara kerja/prinsip *Gaussian Blur* menggunakan EmguCV.

B. Indikator

1. Mahasiswa dapat membuat GUI untuk menerapkan *Gaussian Blur*.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk menerapkan *Gaussian Blur*.
3. Mahasiswa dapat menerapkan *Gaussian Blur* dengan nilai pemburaman yang berbeda menggunakan EmguCV.

C. Tujuan

1. Mahasiswa dapat membuat GUI untuk menerapkan *Gaussian Blur* dengan baik dalam waktu yang singkat.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk menerapkan *Gaussian Blur* dengan baik dan benar dalam waktu yang singkat.
3. Mahasiswa dapat menerapkan *Gaussian Blur* dengan nilai pemburaman yang berbeda menggunakan EmguCV dengan baik dan benar dalam waktu yang singkat.

D. Dasar Teori

1. *Gaussian Blur*

Dalam pengolahan citra digital, *Gaussian Blur* adalah sebuah hasil pemburaman citra digital oleh sebuah fungsi Gaussian, yang ditemukan oleh matematikawan dan ilmuwan Carl Friedrich Gauss. *Gaussian Blur* adalah efek yang banyak digunakan dalam perangkat lunak pengolah grafis, biasanya digunakan untuk mengurangi gangguan pada serta mengurangi detail pada citra digital. Efek visual dari teknik pemburaman ini adalah pemburaman halus yang menyerupai tampilan citra yang melalui layar tembus cahaya, sangat berbeda dengan efek bokeh yang dihasilkan oleh lensa yang tidak fokus. *Gaussian Blur* juga digunakan sebagai tahap pra-pemrosesan dalam algoritma *computer vision* untuk meningkatkan struktur gambar pada skala yang berbeda.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Gaussian Blur</i> Menggunakan EmguCV	No. Job : 3
	Revisi : 01		4 × 50 menit



Gambar 1. Hasil *Gaussian Blur* dengan berbagai nilai *Gaussian*

2. *Gaussian Blur* menggunakan EmguCV

Gaussian Blur merupakan salah satu metode *smoothing* yang sering digunakan untuk memburamkan *noise* yang ada pada citra digital. Fungsi *Gaussian Blur* pada EmguCV adalah sebagai berikut :

```
public static void GaussianBlur(
    IInputArray src,
    IOutputArray dst,
    Size ksize,
    double sigmaX,
    double sigmaY = 0,
    BorderType borderType = BorderType.Reflect101
)
```

Dimana :

src : Masukan citra digital

dst : Keluaran citra digital

ksize : Ukuran inti *Gaussian*, harus bernilai positif dan juga ganjil

sigmaX : Inti *Gaussian* standar deviasi dalam sumbu X

sigmaY(optional) : Inti *Gaussian* standar deviasi dalam sumbu Y

borderType(optional) : Metode ekstrapolasi piksel

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Gaussian Blur</i>	No. Job : 3
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

E. Alat dan Bahan

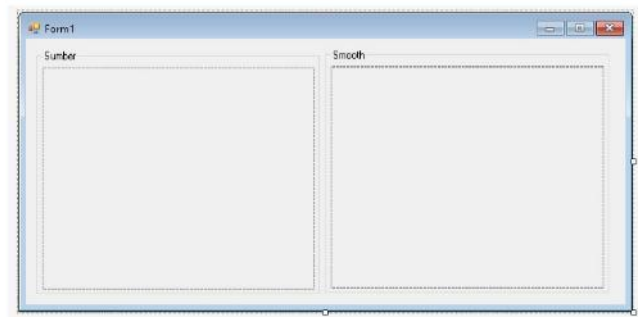
1. Laptop / Desktop PC.
2. Kamera webcam.
3. Library EmguCV versi 3.1.0.2282.
4. Software Microsoft Visual Studio 2012.
5. Media deteksi media1.pdf.

F. Keselamatan Kerja

1. Jauhkan peralatan yang tidak dibutuhkan dari meja kerja.
2. Ikuti prosedur yang ada dengan teliti sesuai dengan langkah kerja yang tersedia.
3. Penggunaan laptop / desktop PC dengan langkah-langkah yang benar dan baik pada saat menyalakan ataupun mematikannya.

G. Langkah Kerja

Buat sebuah project baru dengan nama *Smooth*, lalu desain GUI seperti yang terlihat pada Gambar 1 dan konfigurasi EmguCV.



Gambar 2. GUI untuk menampilkan citra *Gaussian Blur*

Atur *Size Mode* dari semua *imageBox* menjadi *AutoSize* agar ukuran citra digital yang akan ditampilkan menyesuaikan ukuran dari *imageBox*. *Double Click* pada **Form** untuk menambahkan *eventHandler Form_Load* serta masuk ke *code panel*.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Gaussian Blur</i>	No. Job : 3
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

Masukkan *code* berikut untuk menerapkan *Gaussian Blur* pada citra digital :

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using Emgu.CV;
using Emgu.CV.CvEnum;
using Emgu.CV.Structure;
using Emgu.CV.Util;
using Emgu.Util;

namespace Smooth
{
    public partial class Form1 : Form
    {
        Mat source = new Mat();
        Mat smooth = new Mat();
        Capture cap = null;

        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            cap = new Capture();
            cap.ImageGrabbed += cap_ImageGrabbed;
            cap.Start();
        }

        void cap_ImageGrabbed(object sender, EventArgs e)
        {
            cap.Retrieve(source);
            CvInvoke.GaussianBlur(source, smooth, new Size(9, 9), 0, 0);

            imageBox1.Image = source;
            imageBox2.Image = smooth;
        }
    }
}
```

Listing 1. *Code* untuk menerapkan *Gaussian Blur*

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Gaussian Blur</i> Menggunakan EmguCV	No. Job : 3
	Revisi : 01		4 × 50 menit

Untuk penjelasan dari *Listing 1*, silahkan dilihat pada Modul Deteksi Objek Bersegi Menggunakan EmguCV pada Bab 3 Subbab *Smoothing*.

H. Pengujian

1. Pengujian dilakukan dengan menjalankan *project* yang telah dibuat.
2. Amati hasil pemuatan setelah *project* dijalankan.
3. Isikan hasil pengamatan dengan ketentuan ukuran inti *Gaussian* yang berbeda.

Tabel *Gaussian Blur* dengan berbagai ukuran inti *Gaussian*.

No	Inti <i>Gaussian</i>	Hasil
1		
2		
3		
4		
5		
6		
7		
Kesimpulan :		

I. Tugas

Ubahlah ukuran dari *imageBox* dan juga *Form* menjadi ukuran yang tetap (disarankan menggunakan ukuran yang besar, misalnya 1280×720), lalu jalankan *project* dengan pengaturan resolusi dan pengaturan inti *Gaussian* yang berbeda. Amati hasil keluaran lalu isikan data pada table berikut :

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Gaussian Blur</i> Menggunakan EmguCV	No. Job : 3
	Revisi : 01		4 × 50 menit

		Resolusi		
		1280 × 720	640 × 480	320 × 240
Inti Gaussian	3			
	7			
	11			
	15			
	19			
Kesimpulan :				

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Canny Edge Detection</i>	No. Job : 4
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

A. Kompetensi

Memahami cara kerja/prinsip *Canny Edge Detection* menggunakan EmguCV.

B. Indikator

1. Mahasiswa dapat membuat GUI untuk menerapkan *Canny Edge Detection*.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk menerapkan *Canny Edge Detection*.
3. Mahasiswa dapat menerapkan *Canny Edge Detection* dengan nilai parameter yang berbeda menggunakan EmguCV.

C. Tujuan

1. Mahasiswa dapat membuat GUI untuk menerapkan *Canny Edge Detection* dengan baik dalam waktu yang singkat.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk menerapkan *Canny Edge Detection* dengan baik dan benar dalam waktu yang singkat.
3. Mahasiswa dapat menerapkan *Canny Edge Detection* dengan nilai parameter yang berbeda menggunakan EmguCV dengan baik dan benar dalam waktu yang singkat.

D. Dasar Teori

1. *Canny Edge Detection*

Canny Edge Detection adalah operator pendeteksi tepian yang menggunakan algoritma bertahap untuk mendeteksi berbagai tepian pada sebuah citra digital, ditemukan oleh John F. Canny pada tahun 1986. *Canny Edge Detection* merupakan teknik yang mengekstrak informasi struktural dari berbagai objek yang terlihat dan secara drastis mengurangi jumlah data yang akan diproses. Teknik ini telah banyak diterapkan pada berbagai sistem *computer vision*. John F. Canny mengemukakan bahwa persyaratan untuk penerapan *Canny Edge Detection* pada berbagai sistem *computer vision* relatif sama. Dengan demikian, *Canny Edge Detection* diimplementasikan dalam berbagai situasi yang berbeda.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Canny Edge Detection</i>	No. Job : 4
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

2. *Canny Edge Detection* menggunakan EmguCV

Algoritma Canny memberikan cara untuk mengolah data piksel menjadi sebuah garis tepian, dimana dapat disebut juga sebagai *edge*. Dua buah ambang batas histeresis digunakan untuk membuat bentuk *edge* dari data piksel yang ada. Fungsi Canny pada EmguCV adalah sebagai berikut :

```
public static void Canny(
    IInputArray image,
    IOutputArray edges,
    double threshold1,
    double threshold2,
    int apertureSize = 3,
    bool l2Gradient = false
)
```

Dimana :

image : Masukan citra digital
edges : Keluaran citra digital
threshold1 : Ambang batas pertama
threshold2 : Ambang batas kedua
apertureSize(optional) : Parameter bukaan untuk *Sobel operator*
l2Gradient(optional) : Digunakan untuk menentukan norma yang digunakan

Perlu diperhatikan bahwa perbandingan *threshold1* dengan *threshold2* adalah 1:2 sampai dengan 1:3. Nilai perbandingan tersebut disarankan langsung oleh John F. Canny.

E. Alat dan Bahan

1. Laptop / Desktop PC.
2. Kamera *webcam*.
3. *Library* EmguCV versi 3.1.0.2282.
4. *Software* Microsoft Visual Studio 2012.
5. Media deteksi media1.pdf.

F. Keselamatan Kerja

1. Jauhkan peralatan yang tidak dibutuhkan dari meja kerja.
2. Ikuti prosedur yang ada dengan teliti sesuai dengan langkah kerja yang tersedia.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Canny Edge Detection</i>	No. Job : 4
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

- Pergunakan laptop / desktop PC dengan langkah-langkah yang benar dan baik pada saat menyalakan ataupun mematikannya.

G. Langkah Kerja

Buat sebuah *project* baru dengan nama *Canny*, lalu desain GUI seperti yang terlihat pada Gambar 1 dan konfigurasikan EmguCV. Dua *trackbar* digunakan untuk mengatur nilai *threshold1* dan juga *threshold2*, beserta *label* yang digunakan untuk menampilkan nilai dari masing-masing *trackbar*. Atur nilai *maximum* semua *trackbar* menjadi 255, lalu tetapkan nilai awal untuk *trackbar1* sebesar 0 dan *trackbar2* sebesar 255. *Timer* digunakan untuk *update* nilai dari semua *trackbar* secara terus-menerus dengan selang waktu yang terdapat pada *Properties*, yaitu 1 ms.



Gambar 1. GUI untuk menerapkan *Canny Edge Detection*

Atur **Size Mode** dari semua *imageBox* menjadi **StretchImage** agar ukuran citra digital yang akan ditampilkan menyesuaikan ukuran dari *imageBox*. *Double Click* pada **Form** untuk menambahkan *eventHandler* **Form_Load** serta masuk ke *code panel*. Masukkan *code* berikut untuk menerapkan Gaussian Blur pada citra digital :

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
```


	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Canny Edge Detection</i>	No. Job : 4
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

```

using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using Emgu.CV;
using Emgu.CV.CvEnum;
using Emgu.CV.Structure;
using Emgu.CV.Util;
using Emgu.Util;

namespace canny
{
    public partial class Form1 : Form
    {
        Mat source = new Mat();
        Mat canny = new Mat();
        Capture cap = null;
        int threshold1, threshold2;

        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            cap = new Capture();
            cap.ImageGrabbed += process;
            cap.Start();
        }

        private void process(object sender, EventArgs e)
        {
            cap.Retrieve(source, 0);
            CvInvoke.Canny(source, canny, threshold1, threshold2);
            imageBox1.Image = source;
            imageBox2.Image = canny;
        }

        private void timer1_Tick(object sender, EventArgs e)
        {
            threshold1 = trackBar1.Value;
            threshold2 = trackBar2.Value;

            label3.Text = threshold1.ToString();
            label4.Text = threshold2.ToString();
        }
    }
}

```

Listing 1. Code untuk menerapkan Canny Edge Detection

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Canny Edge Detection</i> Menggunakan EmguCV	No. Job : 4
	Revisi : 01		4 × 50 menit

Untuk penjelasan dari *Listing 1*, silahkan dilihat pada Modul Deteksi Objek Bersegi Menggunakan EmguCV pada Bab 3 Subbab *Canny Edge Detection*.

H. Pengujian

1. Pengujian dilakukan dengan menjalankan *project* yang telah dibuat.
2. Amati pengaruh dari parameter *threshold1* dan *threshold2* ketika berubah nilai.
3. Isikan hasil pengamatan dengan penggunaan parameter *threshold1* dan *threshold2* yang bervariasi.

Tabel *Canny Edge Detection* dengan berbagai nilai parameter *threshold1* dan *threshold2*.

No	<i>Threshold1</i>	<i>Threshold2</i>	Hasil
1			
2			
3			
4			
5			
Kesimpulan :			

I. Tugas

Terapkan *Gaussian Blur* sebelum proses *Canny Edge Detection*. Buatlah dengan menggunakan inti *Gaussian* yang berbeda, serta dengan parameter *threshold1* dan *threshold2* yang juga berbeda. Amati hasil keluaran lalu isikan data pada tabel berikut :

No	Inti <i>Gaussian</i>	<i>Threshold1</i>	<i>Threshold2</i>	Hasil
1				
2				
3				
4				
5				
Kesimpulan :				

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Contours Detection</i>	No. Job : 5
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

A. Kompetensi

Memahami cara kerja/prinsip *Contours Detection* menggunakan EmguCV.

B. Indikator

1. Mahasiswa dapat membuat GUI untuk menerapkan *Contours Detection*.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk menerapkan *Contours Detection*.
3. Mahasiswa dapat menerapkan *Contours Detection* dengan parameter yang berbeda menggunakan EmguCV.

C. Tujuan

1. Mahasiswa dapat membuat GUI untuk menerapkan *Contours Detection* dengan baik dalam waktu yang singkat.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk menerapkan *Contours Detection* dengan baik dan benar dalam waktu yang singkat.
3. Mahasiswa dapat menerapkan *Contours Detection* dengan parameter yang berbeda menggunakan EmguCV dengan baik dan benar dalam waktu yang singkat.

D. Dasar Teori

1. *Contours Detection*

Pada dasarnya, *contour* digunakan setelah proses pendeteksian tepi berhasil dilakukan. Dengan menggunakan *Canny Edge Detection* didapatkan batas tepian pada tiap *pixel*, namun proses tersebut tidak dapat digunakan untuk mengkombinasikan tiap segmen yang menjadi sebuah bentuk tertentu. Sebelum memulai menggunakan metode *Contour* ini, perlu dipahami terlebih dahulu arti dari *contour* itu sendiri. Arti dari *contour* bisa berbeda tergantung situasi yang ada, namun pengertian umumnya yaitu *contour* merupakan sekumpulan *pixel* atau *point* yang merepresentasikan sebuah garis lengkung, dalam artian struktural, pada sebuah citra digital. *Contours Detection* merupakan teknik yang sangat berguna untuk analisis bentuk serta pendeteksian dan pengenalan suatu objek.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Contours Detection</i>	No. Job : 5
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

2. *Contours Detection* menggunakan EmguCV

Pada EmguCV, *contours* dapat dicari menggunakan fungsi *FindContours*, dengan rincian sebagai berikut :

```
public static void FindContours(
    IInputOutputArray image,
    IOutputArray contours,
    IOutputArray hierarchy,
    RetrType mode,
    ChainApproxMethod method,
    Point offset = null
)
```

Dimana :

image : Masukan *image* yang berkualitas 8-bit serta berbentuk *binary image*
contours : *Contour* yang terdeteksi dan disimpan dalam bentuk titik vektor
hierarchy : Keluaran vektor tambahan
mode : Mode pengambilan *contour*
method : Metode pendekatan
offset(optional) : Nilai *offset* dari setiap titik *contour* yang telah bergeser

E. Alat dan Bahan

1. Laptop / Desktop PC.
2. Kamera *webcam*.
3. Library EmguCV versi 3.1.0.2282.
4. Software Microsoft Visual Studio 2012.
5. Media deteksi media2.pdf.

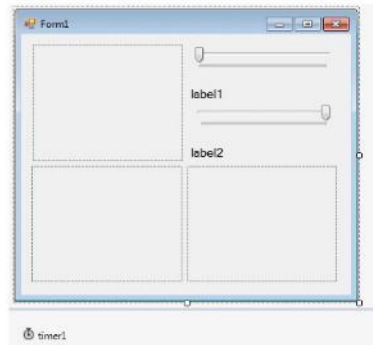
F. Keselamatan Kerja

1. Jauhkan peralatan yang tidak dibutuhkan dari meja kerja.
2. Ikuti prosedur yang ada dengan teliti sesuai dengan langkah kerja yang tersedia.
3. Penggunaan laptop / desktop PC dengan langkah-langkah yang benar dan baik pada saat menyalakan ataupun mematikannya.

G. Langkah Kerja

Buat sebuah *project* baru dengan nama *Contours*, lalu desain GUI seperti yang terlihat pada Gambar 1 dan konfigurasi EmguCV.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Contours Detection</i>	No. Job : 5
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit



Gambar 1. GUI untuk menerapkan *Contours Detection*

Atur **Size Mode** dari semua *imageBox* menjadi **StretchImage** agar ukuran citra digital yang akan ditampilkan menyesuaikan ukuran dari *imageBox*. *Double Click* pada **Form** untuk menambahkan *eventHandler Form_Load* serta masuk ke *code panel*. Masukkan *code* berikut untuk menerapkan *Contours Detection* pada sebuah citra digital:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using Emgu.CV;
using Emgu.CV.CvEnum;
using Emgu.CV.Structure;
using Emgu.CV.Util;
using Emgu.Util;

namespace contour
{
    public partial class Form1 : Form
    {
        Mat source=new Mat(),Mat canny=new Mat(),Mat output=new Mat();
        Capture cap = null;
    }
}
```

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Contours Detection</i>	No. Job : 5
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

```

int threshold1, threshold2;

public Form1()
{
    InitializeComponent();
}

private void Form1_Load(object sender, EventArgs e)
{
    cap = new Capture();
    cap.ImageGrabbed += cap_ImageGrabbed;
    cap.Start();
}

void cap_ImageGrabbed(object sender, EventArgs e)
{
    cap.Retrieve(source);
    output = source.Clone();
    CvInvoke.Canny(source, canny, threshold1, threshold2);
    Mat cClone = canny.Clone();
    using (VectorOfVectorOfPoint contours = new
    VectorOfVectorOfPoint())
    {
        CvInvoke.FindContours(cClone, contours, null,
        RetrType.External, ChainApproxMethod.ChainApproxSimple);
        int count = contours.Size;
        for (int i = 0; i < count; i++)
        {
            CvInvoke.DrawContours(output, contours, i, new
            Bgr(Color.Red).MCvScalar, 3, LineType.EightConnected, null,
            0, new Point(0, 0));
        }
    }
    imageBox1.Image = source;
    imageBox2.Image = canny;
    imageBox3.Image = output;
}

private void timer1_Tick(object sender, EventArgs e)
{
    threshold1 = trackBar1.Value;
    threshold2 = trackBar2.Value;
    label1.Text = threshold1.ToString();
    label2.Text = threshold2.ToString();
}
}

```

Listing 1. Code untuk menerapkan *Contours Detection*

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Contours Detection</i>	No. Job : 5
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

Untuk penjelasan dari *Listing 1*, silahkan dilihat pada Modul Deteksi Objek Bersegi Menggunakan EmguCV pada Bab 3 Subbab *Contours*.

H. Pengujian

1. Pengujian dilakukan dengan menjalankan *project* yang telah dibuat.
2. Amati hasil pemuatan setelah *project* dijalankan.
3. Isikan hasil pengamatan dengan *Contours Detection* dengan masukan hasil *Canny Edge Detection* yang berbeda.

Tabel *Contours Detection* dengan berbagai masukan hasil *Canny Edge Detection*.

No	<i>Canny Edge Detection</i>	Hasil <i>Contours Detection</i>
1		
2		
3		
4		
5		
Kesimpulan :		

I. Tugas

Gunakan hasil *Canny Edge Detection* yang tetap, lalu gunakan parameter *RetrType* dan *ChainApproxMethod* pada fungsi *FindContour*. Amati pengaruh yang dihasilkan pada keluaran *Contour Detection* sesuai pada tabel berikut serta, tampilkan jumlah *contours* yang terdeteksi pada sebuah *label* :

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Contours Detection</i>	No. Job : 5
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

No	RetrType	ChainApproxMethod	Jumlah Contours	Hasil
1	Ccomp	ChainApproxNone		
2		ChainApproxSimple		
3	External	ChainApproxNone		
4		ChainApproxSimple		
5	List	ChainApproxNone		
6		ChainApproxSimple		
7	Tree	ChainApproxNone		
8		ChainApproxSimple		
Kesimpulan :				

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Shape Detection</i>	No. Job : 6
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

A. Kompetensi

Memahami cara kerja/prinsip *Shape Detection* menggunakan EmguCV.

B. Indikator

1. Mahasiswa dapat membuat GUI untuk menerapkan *Shape Detection*.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk menerapkan *Shape Detection*.
3. Mahasiswa dapat menerapkan *Shape Detection* dengan objek yang berbeda menggunakan EmguCV.

C. Tujuan

1. Mahasiswa dapat membuat GUI untuk menerapkan *Shape Detection* dengan baik dalam waktu yang singkat.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk menerapkan *Shape Detection* dengan baik dan benar dalam waktu yang singkat.
3. Mahasiswa dapat menerapkan *Shape Detection* dengan objek yang berbeda menggunakan EmguCV dengan baik dan benar dalam waktu yang singkat.

D. Dasar Teori

1. *Shape Detection*

. Sistem penglihatan manusia tampaknya menggunakan berbagai jenis sumber informasi untuk mendeteksi dan mengenali sebuah objek. Pada tingkatan terendah pada suatu pengenalan objek, informasi mengenai tepian dan suatu bagian tertentu digunakan untuk mengekstraksi sebuah unit perseptual. Pada beberapa kasus, informasi dari batasan bentuk, seperti bentuk persegi kendaraan dalam suatu citra udara, pengenalan objek memiliki suatu peranan yang sangatlah penting.

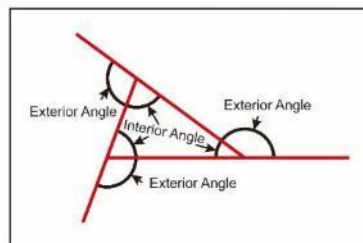
Mendeteksi suatu bentuk merupakan salah satu kegunaan dari metode *contour* yang sudah dipelajari sebelumnya. Cara yang paling sederhana untuk mendeteksi bentuk adalah dengan mencari sudut *exterior* antar garis yang bersinggungan. Garis-garis tersebut dapat dicari dengan menerapkan metode *contour*, yang sebelumnya telah

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Shape Detection</i>	No. Job : 6
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

dilakukan proses pengurangan *noise* menggunakan *GaussianBlur* dan juga metode *canny* untuk mencari garis tepi dari objek yang ingin dideteksi.

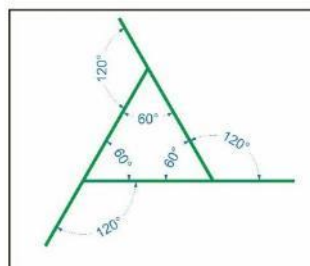
2. Sudut *Exterior*

Sudut *exterior* merupakan sudut terluar dari setiap sisi pada bidang *polygon* terhadap perpanjangan sisi lain yang berdekatan. Ilustrasi dari sudut *exterior* nampak seperti Gambar 1.



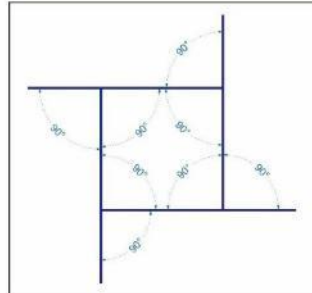
Gambar 1. Sudut *exterior* dan sudut *interior*

Untuk mendeteksi objek bersegi seperti segitiga, segiempat, segilima, dan juga segienam, maka harus diketahui terlebih dahulu besarnya sudut *exterior* dari masing-masing objek. Jika diasumsikan setiap sisi dari objek memiliki panjang yang sama, maka besarnya sudut *exterior* akan terlihat seperti pada Gambar 2, Gambar 3, Gambar 4, dan juga Gambar 5.

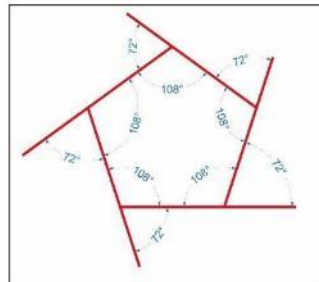


Gambar 2. Sudut *exterior* pada segitiga

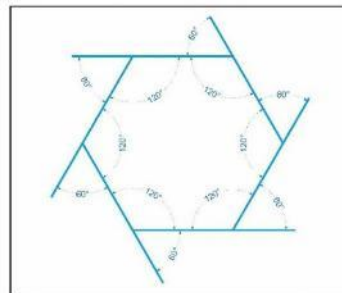
	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Shape Detection</i>	No. Job : 6
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit



Gambar 3. Sudut *exterior* pada segiempat



Gambar 4. Sudut *exterior* pada segilima



Gambar 5. Sudut *exterior* pada segienam

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Shape Detection</i>	No. Job : 6
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

3. *Shape Detection* menggunakan EmguCV

EmguCV menyediakan fungsi *GetExteriorAngleDegree* yang bisa digunakan untuk mencari sudut *exterior* dari dua garis yang bersinggungan. Fungsi tersebut berada pada namespace *Emgu.CV.Structure* dan pada class *LineSegment2D*. Penjelasan fungsi *GetExteriorAngleDegree* dapat dilihat pada Listing 1 dan untuk penggunaannya dapat dilihat pada Listing 2.

```
public double GetExteriorAngleDegree(
    LineSegment2D otherLine
)
```

Listing 1. Fungsi *GetExteriorAngleDegree* pada EmguCV

Dimana :

otherLine : Garis lain yang bersinggungan

```
double angle = Math.Abs(line.GetExteriorAngleDegree(otherLine))
```

Listing 2. Penggunaan *GetExteriorAngleDegree*

Dimana :

angle : Variabel untuk menyimpan hasil perhitungan sudut *exterior*
line : Garis pertama yang digunakan untuk mencari sudut *exterior*
otherLine : Garis lain yang bersinggungan dengan garis pertama

E. Alat dan Bahan

1. Laptop / Desktop PC.
2. Kamera webcam.
3. Media deteksi media2.pdf.
4. Library EmguCV versi 3.1.0.2282.
5. Software Microsoft Visual Studio 2012.

F. Keselamatan Kerja

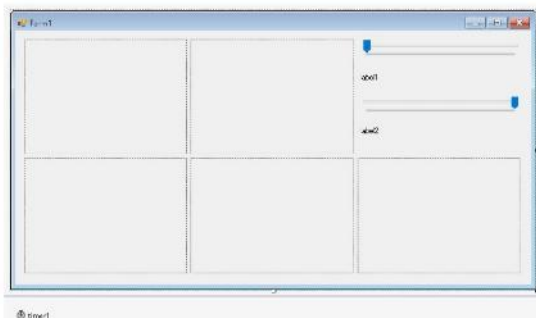
1. Jauhkan peralatan yang tidak dibutuhkan dari meja kerja.
2. Ikuti prosedur yang ada dengan teliti sesuai dengan langkah kerja yang tersedia.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Shape Detection</i>	No. Job : 6
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

- Pergunakan laptop / desktop PC dengan langkah-langkah yang benar dan baik pada saat menyalakan ataupun mematikannya.

G. Langkah Kerja

Buat sebuah *project* baru dengan nama *Shape*, lalu desain GUI seperti yang terlihat pada Gambar 1 dan konfigurasi EmguCV. Dua *trackbar* digunakan untuk mengatur nilai *threshold1* dan juga *threshold2*, beserta *label* yang digunakan untuk menampilkan nilai dari masing-masing *trackbar*. Atur nilai *maximum* semua *trackbar* menjadi 255, lalu tetapkan nilai awal untuk *trackbar1* sebesar 0 dan *trackbar2* sebesar 255. *Timer* digunakan untuk *update* nilai dari semua *trackbar* secara terus-menerus dengan selang waktu yang terdapat pada *Properties*, yaitu 1 ms.



Gambar 1. GUI untuk menampilkan citra digital

Atur **Size Mode** dari semua *imageBox* menjadi **StretchImage** agar ukuran citra digital yang akan ditampilkan menyesuaikan ukuran dari *imageBox*. **Double Click** pada **Form** untuk menambahkan *eventHandler* **Form_Load** serta masuk ke *code panel*.

Masukkan *code* berikut untuk menerapkan *Shape Detection* sebuah citra digital:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
```

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Shape Detection</i>	No. Job : 6
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

```

using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using Emgu.CV;
using Emgu.CV.CvEnum;
using Emgu.CV.Structure;
using Emgu.CV.Util;
using Emgu.Util;

namespace shape
{
    public partial class Form1 : Form
    {
        Capture cap = null;
        Mat source = new Mat();
        Mat blur = new Mat();
        Mat canny = new Mat();
        Mat draw = new Mat();
        Mat output = new Mat();
        double threshold1, threshold2;

        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            cap = new Capture(0);
            cap.ImageGrabbed += cap_ImageGrabbed;
            cap.Start();
        }

        private void timer1_Tick(object sender, EventArgs e)
        {
            threshold1 = trackBar1.Value;
            threshold2 = trackBar2.Value;
            label1.Text = threshold1.ToString();
            label2.Text = threshold2.ToString();
        }

        void cap_ImageGrabbed(object sender, EventArgs e)
        {
            cap.Retrieve(source);
            draw = source.Clone();
            output = source.Clone();
            CvInvoke.GaussianBlur(source, blur, new Size(9, 9), 0);
            CvInvoke.Canny(blur, canny, threshold1, threshold2);
            Mat clone = canny.Clone();

```


	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Shape Detection</i>	No. Job : 6
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

```

using (VectorOfVectorOfPoint contours = new
VectorOfVectorOfPoint())
{
    CvInvoke.FindContours(clone, contours, null,
    RetrType.External, ChainApproxMethod.ChainApproxSimple);
    int count = contours.Size;
    for (int i = 0; i < count; i++)
    {
        CvInvoke.DrawContours(draw, contours, i, new
        Bgr(Color.Red).MCvScalar, 3, LineType.EightConnected,
        null, 0, new Point(0, 0));
        using (VectorOfPoint contour = contours[i])
        using (VectorOfPoint approx = new VectorOfPoint())
        {
            CvInvoke.ApproxPolyDP(contour, approx,
            CvInvoke.ArcLength(contour, true) * 0.02, true);
            Point[] pts = approx.ToArray();
            if (CvInvoke.ContourArea(approx, false) > 500)
            {
                if (approx.Size == 4)
                {
                    LineSegment2D[] edges =
                    PointCollection.PolyLine(pts, true);
                    bool shape = false;
                    for (int j = 0; j < edges.Length; j++)
                    {
                        double angle = Math.Abs(edges[(j + 1) %
                        edges.Length].GetExteriorAngleDegree(edges[j]));
                        if (angle < 80 || angle > 100)
                        {
                            shape = false;
                            break;
                        }
                        else { shape = true; }
                    }
                    if (shape == true)
                    {
                        CvInvoke.PolyLines(output, pts, true, new
                        Bgr(Color.Blue).MCvScalar, 3,
                        LineType.EightConnected, 0);
                    }
                }
            }
        }
        imageBox1.Image = source;
        imageBox2.Image = blur;
        imageBox3.Image = canny;
        imageBox4.Image = draw;
    }
}

```

Listing 1. Code untuk memuat sebuah citra digital

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Shape Detection</i>	No. Job : 6
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

Untuk penjelasan dari *Listing 1*, silahkan dilihat pada Modul Deteksi Objek Bersegi Menggunakan EmguCV pada Bab 4.

H. Pengujian

1. Pengujian dilakukan dengan menjalan *project* yang telah dibuat.
2. Amati hasil pemuatan setelah *project* dijalankan.
3. Isikan hasil pengamatan *Shape Detection* dengan berbagai objek bersegi yang berbeda dilihat dari beberapa sudut kemiringan dengan memiringkan objek secara *horizontal* dan *vertical* terhadap kamera.

Tabel *Shape Detection* dengan berbagai objek yang berbeda.

No	Objek	Kemiringan Objek	Hasil
1	Segitiga		
2			
3			
4			
5	Segiempat		
6			
7			
8			
9	Segilima		
10			
11			
12			
13	Segienam		
14			
15			
16			
Kesimpulan :			

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Menerapkan <i>Shape Detection</i>	No. Job : 6
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

I. Tugas

Buatlah aplikasi deteksi objek bersegi dengan nilai *epsilon* yang berbeda. Amati perbedaan ketika *epsilon* bernilai besar dengan *epsilon* bernilai kecil. Masukkan hasil deteksi ke dalam tabel dan berikan kesimpulannya.

No	Objek	<i>Epsilon</i>	Hasil
1	Segitiga	2% dari <i>Arc Length</i>	
2		7% dari <i>Arc Length</i>	
3		12% dari <i>Arc Length</i>	
4	Segiempat	2% dari <i>Arc Length</i>	
5		7% dari <i>Arc Length</i>	
6		12% dari <i>Arc Length</i>	
7	Segilima	2% dari <i>Arc Length</i>	
8		7% dari <i>Arc Length</i>	
9		12% dari <i>Arc Length</i>	
10	Segienam	2% dari <i>Arc Length</i>	
11		7% dari <i>Arc Length</i>	
12		12% dari <i>Arc Length</i>	
Kesimpulan :			

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan <i>Threshold</i>	No. Job : 7
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

A. Kompetensi

Memahami cara kerja/prinsip *threshold* citra digital menggunakan EmguCV.

B. Indikator

1. Mahasiswa dapat membuat GUI untuk *threshold* citra digital.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk *threshold* citra digital.
3. Mahasiswa dapat menerapkan berbagai jenis tipe *threshold* citra digital menggunakan EmguCV.

C. Tujuan

1. Mahasiswa dapat membuat GUI untuk *threshold* citra digital dengan baik dalam waktu yang singkat.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk *threshold* citra digital dengan baik dan benar dalam waktu yang singkat.
3. Mahasiswa dapat menerapkan berbagai jenis tipe *threshold* citra digital menggunakan EmguCV dengan baik dan benar dalam waktu yang singkat.

D. Dasar Teori

1. *Threshold*

Threshold merupakan cara yang paling sederhana untuk melakukan segmentasi citra digital. Metode *Threshold* paling sederhana melakukan penimpaan pada setiap piksel pada sebuah citra digital, dimana akan menghasilkan piksel hitam jika intensitas warna pada citra digital kurang dari nilai konstanta yang telah ditentukan, dan akan menghasilkan piksel putih jika intensitas warna pada citra digital lebih besar atau sama dengan nilai konstanta.



Gambar 1. Hasil proses *threshold* sederhana

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan <i>Threshold</i>	No. Job : 7
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

2. *Threshold* menggunakan EmguCV

.Untuk melakukan proses *threshold*, citra digital terlebih dahulu harus dirubah ke bentuk *single-channel*. Untuk melakukan proses tersebut, cukup mengubah citra digital ke dalam skala warna *grayscale* dengan menggunakan fungsi `CvInvoke.CvtColor`. Setelah citra digital berhasil dirubah ke bentuk *single-channel*, maka citra digital siap untuk diproses menggunakan fungsi `CvInvoke.Threshold`, berikut penggunaan dari fungsi `CvInvoke.Threshold`:

```
public static double Threshold(
    IInputArray src,
    IOutputArray dst,
    double threshold,
    double maxValue,
    ThresholdType thresholdType
)
```

Dimana :

`src` : masukan citra digital, harus berbentuk *single-channel*.
`dst` : keluaran citra digital, akan berbentuk *single-channel*.
`threshold` : nilai *threshold* yang akan diterapkan.
`maxValue` : nilai maksimal untuk digunakan dengan tipe *thresholding* `CV_THRESH_BINARY` dan `CV_THRESH_BINARY_INV`.
`thresholdType` : tipe *thresholding* yang akan diterapkan.

E. Alat dan Bahan

1. Laptop / Desktop PC.
2. Library EmguCV versi 3.1.0.2282.
3. Software Microsoft Visual Studio 2012.
4. Kamera Webcam.

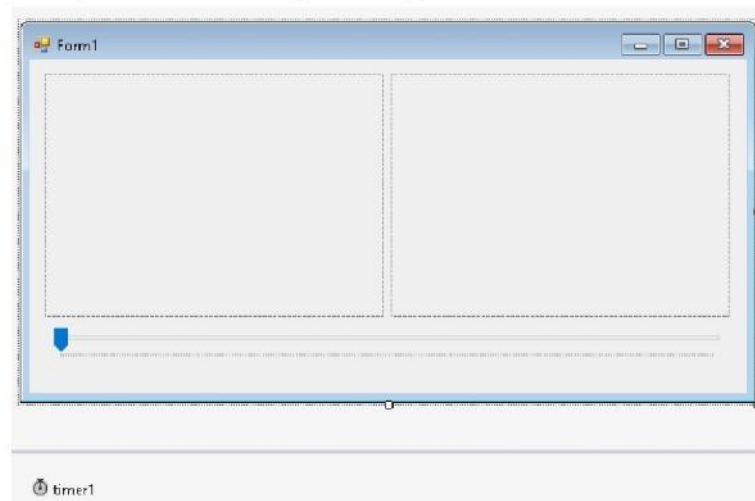
F. Keselamatan Kerja

1. Jauhkan peralatan yang tidak dibutuhkan dari meja kerja.
2. Ikuti prosedur yang ada dengan teliti sesuai dengan langkah kerja yang tersedia.
3. Penggunaan laptop / desktop PC dengan langkah-langkah yang benar dan baik pada saat menyalakan ataupun mematikannya.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan <i>Threshold</i>	No. Job : 7
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

G. Langkah Kerja

Buat sebuah *project* baru dengan nama *Threshold*, lalu desain GUI seperti yang terlihat pada Gambar 2 dan konfigurasi EmguCV.



Gambar 2. GUI untuk menerapkan *threshold*

Atur **Size Mode** dari semua *imageBox* menjadi **StretchImage** agar ukuran citra digital yang akan ditampilkan menyesuaikan ukuran dari *imageBox*. **Double Click** pada **Form** untuk menambahkan *eventHandler* **Form_Load** serta masuk ke *code panel*. Dua *imageBox* digunakan untuk menampilkan citra digital awal sebelum diproses dan untuk menampilkan citra digital hasil *threshold*. *Trackbar* digunakan untuk mengatur nilai *threshold* dengan nilai maksimum *trackbar* adalah 255. *Timer* digunakan untuk meng-*update* data dari *trackbar* secara *real-time*.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan <i>Threshold</i>	No. Job : 7
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

Masukkan *code* berikut untuk menerapkan *threshold*:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using Emgu.CV;
using Emgu.CV.CvEnum;
using Emgu.CV.Structure;
using Emgu.CV.Util;
using Emgu.Util;
namespace threshold
{
    public partial class Form1 : Form
    {
        Capture cap = null;
        Mat source = new Mat();
        Mat threshold = new Mat();
        int thr1;
        public Form1()
        {
            InitializeComponent();
        }
        private void Form1_Load(object sender, EventArgs e)
        {
            cap = new Capture();
            cap.ImageGrabbed += cap_ImageGrabbed;
            cap.Start();
        }
        void cap_ImageGrabbed(object sender, EventArgs e)
        {
            cap.Retrieve(source);
            CvInvoke.CvtColor(source, threshold, ColorConversion.Bgr2Gray);
            CvInvoke.Threshold(source, threshold, thr1, 255,
                ThresholdType.Binary);
            imageBox1.Image = source;
            imageBox2.Image = threshold;
        }
        private void timer1_Tick(object sender, EventArgs e)
        {
            thr1 = trackBar1.Value;
        }
    }
}
```

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan <i>Threshold</i>	No. Job : 7
	Revisi : 01	Menggunakan EmguCV	4 × 50 menit

H. Pengujian

1. Pengujian dilakukan dengan menjalan *project* yang telah dibuat.
2. Amati hasil pemuatan setelah *project* dijalankan.
3. Isikan hasil pengamatan dengan ketentuan nilai *threshold* yang berbeda serta dengan kondisi pencahayaan yang berbeda.

Tabel penerapan nilai *threshold* dengan berbagai kondisi cahaya.

No	Citra Asli	Nilai <i>Threshold</i>	Kondisi Cahaya	
			Redup	Terang
1				
2				
3				
4				
5				
Kesimpulan :				

I. Tugas

Terapkan berbagai jenis tipe *threshold* pada sebuah citra digital dengan kondisi pencahayaan yang berbeda, dengan ketentuan seperti pada tabel berikut:

No	Tipe <i>Threshold</i>	Citra Asli	Kondisi Cahaya	
			Redup	Terang
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
Kesimpulan :				

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan ROI Menggunakan EmguCV	No. Job : 8
	Revisi : 01		4 × 50 menit

A. Kompetensi

Memahami cara kerja/prinsip ROI citra digital menggunakan EmguCV.

B. Indikator

1. Mahasiswa dapat membuat GUI untuk ROI citra digital.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk ROI citra digital.
3. Mahasiswa dapat membuat berbagai ukuran ROI citra digital menggunakan EmguCV.

C. Tujuan

1. Mahasiswa dapat membuat GUI untuk *threshold* citra digital dengan baik dalam waktu yang singkat.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk *threshold* citra digital dengan baik dan benar dalam waktu yang singkat.
3. Mahasiswa dapat membuat berbagai ukuran ROI citra digital menggunakan EmguCV dengan baik dan benar dalam waktu yang singkat.

D. Dasar Teori

1. ROI (*Region Of Interest*)

Region Of Interest adalah suatu bagian dari sebuah citra digital dimana bagian tersebut digunakan untuk proses yang lebih lanjut. Dalam satu citra digital bisa memuat beberapa wilayah ROI, sesuai dengan kebutuhan. Bentuk dari bagian ROI bisa berbasis bentuk garis, lingkaran, ataupun berbagai jenis bentuk poligon.

2. ROI menggunakan EmguCV

Untuk melakukan proses ROI, citra digital harus dirubah dahulu menjadi format `Image<Bgr, Byte>`. Hal ini dikarenakan fungsi ROI yang akan digunakan hanya mendukung format `Image<Bgr, Byte>`. Citra digital akan diambil ROI berdasarkan bentuk yang dibuat, seperti pada saat ini yang menggunakan bentuk segiempat sebagai dasar dari wilayah ROI. Untuk membuat bentuk segiempat, maka digunakan fungsi `Rectangle` yang telah disediakan oleh *library* bawaan dari *OS Windows*.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan ROI Menggunakan EmguCV	No. Job : 8
	Revisi : 01		4 × 50 menit

```
public Rectangle (int x, int y, int width, int height);
```

Dimana :

x : koordinat x dari titik pojok kiri atas segiempat
y : koordinat y dari titik pojok kiri atas segiempat
width : lebar dari segiempat
height : tinggi dari segiempat

Segiempat yang telah dibuat, dimasukkan ke dalam fungsi ROI dengan penjelasan fungsi sebagai berikut:

```
source.ROI = rect;
```

Dimana :

source : citra digital yang ingin di ROI, format Image<Bgr, Byte>
rect : segiempat yang digunakan untuk dasar bentuk ROI

E. Alat dan Bahan

1. Laptop / Desktop PC.
2. Library EmguCV versi 3.1.0.2282.
3. Software Microsoft Visual Studio 2012.
4. Kamera Webcam.

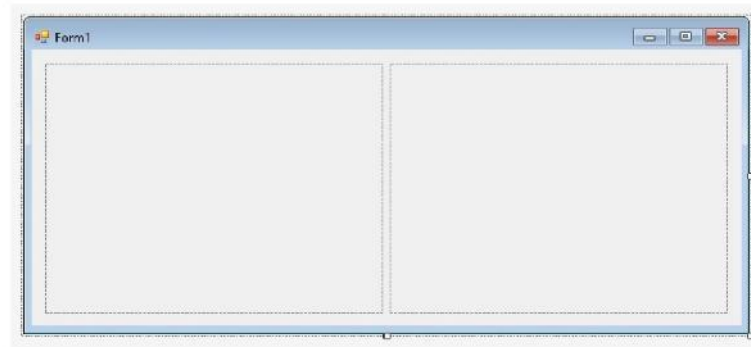
F. Keselamatan Kerja

1. Jauhkan peralatan yang tidak dibutuhkan dari meja kerja.
2. Ikuti prosedur yang ada dengan teliti sesuai dengan langkah kerja yang tersedia.
3. Penggunaan laptop / desktop PC dengan langkah-langkah yang benar dan baik pada saat menyalakan ataupun mematkannya.

G. Langkah Kerja

Buat sebuah *project* baru dengan nama *ROI*, lalu desain GUI seperti yang terlihat pada Gambar 1 dan konfigurasi EmguCV.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan ROI Menggunakan EmguCV	No. Job : 8
	Revisi : 01		4 × 50 menit



Gambar 1. GUI untuk menerapkan ROI

Atur **Size Mode** dari semua *imageBox* menjadi **StretchImage** agar ukuran citra digital yang akan ditampilkan menyesuaikan ukuran dari *imageBox*. **Double Click** pada **Form** untuk menambahkan *eventHandler* **Form_Load** serta masuk ke *code panel*. Dua *imageBox* digunakan untuk menampilkan citra digital awal sebelum diproses dan untuk menampilkan citra digital hasil ROI.

Masukkan *code* berikut untuk menerapkan ROI:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

using Emgu.CV;
using Emgu.CV.CvEnum;
using Emgu.CV.Structure;
using Emgu.CV.Util;
using Emgu.Util;
```

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan ROI Menggunakan EmguCV	No. Job : 8
	Revisi : 01		4 × 50 menit

```

namespace ROI
{
    public partial class Form1 : Form
    {
        Capture cap = null;
        Mat source = new Mat();
        Image<Bgr, Byte> roi;

        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            cap = new Capture();
            cap.ImageGrabbed += cap_ImageGrabbed;
            cap.Start();
        }

        void cap_ImageGrabbed(object sender, EventArgs e)
        {
            cap.Retrieve(source);

            Rectangle rect = new Rectangle(10, 10, 300, 300);
            roi = source.ToImage<Bgr, Byte>();
            roi.ROI = rect;

            imageBox1.Image = source;
            imageBox2.Image = roi;
        }
    }
}

```

H. Pengujian

1. Pengujian dilakukan dengan menjalankan *project* yang telah dibuat.
2. Amati hasil pemuatan setelah *project* dijalankan.
3. Isikan hasil pengamatan dengan ukuran segiempat yang berbeda.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan ROI Menggunakan EmguCV	No. Job : 8
	Revisi : 01		4 × 50 menit

Tabel penerapan ROI dengan ukuran segiempat yang berbeda.

No	Citra Asli	Ukuran Segiempat	Hasil ROI
1			
2			
3			
4			
5			
Kesimpulan :			

I. Tugas

Terapkan berbagai bentuk beserta ukuran bentuk untuk proses ROI, dengan ketentuan tabel seperti berikut :

No	Bentuk ROI	Ukuran	Citra Asli	Hasil ROI
1	Lingkaran			
2				
3	Segitiga			
4				
5	Poligon Acak			
6				
Kesimpulan :				

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan <i>Histogram Equalization</i> Menggunakan EmguCV	No. Job : 9
	Revisi : 01		4 × 50 menit

A. Kompetensi

Memahami cara kerja/prinsip *histrogram equalization* citra digital menggunakan EmguCV.

B. Indikator

1. Mahasiswa dapat membuat GUI untuk *histrogram equalization* citra digital.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk *histrogram equalization* citra digital.
3. Mahasiswa dapat menerapkan *histrogram equalization* citra digital dengan berbagai kondisi cahaya menggunakan EmguCV.

C. Tujuan

1. Mahasiswa dapat membuat GUI untuk *histrogram equalization* citra digital dengan baik dalam waktu yang singkat.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk *histrogram equalization* citra digital dengan baik dan benar dalam waktu yang singkat.
3. Mahasiswa dapat menerapkan *histrogram equalization* citra digital dengan berbagai kondisi cahaya menggunakan EmguCV dengan baik dan benar dalam waktu yang singkat.

D. Dasar Teori

1. *Histogram*

Histogram merupakan representasi grafis dari suatu distribusi tonal dalam sebuah citra digital. Histogram menggambarkan nilai tonal pada setiap piksel pada citra digital. Dengan melihat data histogram pada suatu citra digital, maka dapat digunakan untuk menilai seluruh distribusi tonal secara sekilas.

2. *Histogram Equalization* menggunakan EmguCV

Histogram Equalization merupakan suatu metode yang digunakan untuk meningkatkan kontras dari citra digital, dengan cara melebarkan rentang dari nilai intensitas.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan <i>Histogram Equalization</i> Menggunakan EmguCV	No. Job : 9
	Revisi : 01		4 × 50 menit

Fungsi *histogram equalization* yang terdapat pada EmguCV harus memiliki masukan citra yang berupa *single-channel*. Oleh karena itu, citra digital yang ingin diterapkan fungsi *histogram equalization* harus dirubah terlebih dahulu ke format *grayscale* dengan menggunakan fungsi `CvInvoke.CvtColor`. Berikut penjelasan lengkap dari fungsi `CvInvoke.EqualizeHist` pada EmguCV :

```
public static void EqualizeHist(
    IInputArray src,
    IOutputArray dst
)
```

Dimana :

`src` : masukan citra digital *single-channel*
`dst` : keluaran citra digital dengan format *single-channel*

E. Alat dan Bahan

1. Laptop / Desktop PC.
2. Library EmguCV versi 3.1.0.2282.
3. Software Microsoft Visual Studio 2012.
4. Kamera Webcam.

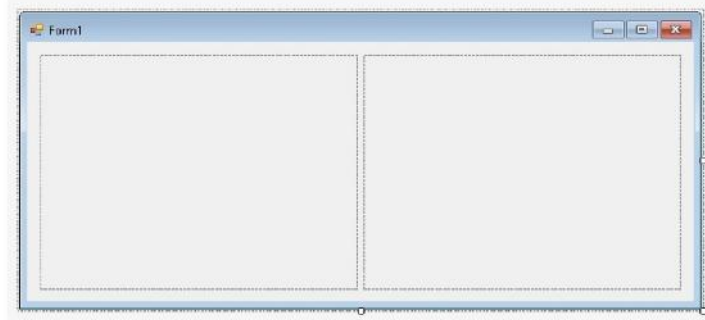
F. Keselamatan Kerja

1. Jauhkan peralatan yang tidak dibutuhkan dari meja kerja.
2. Ikuti prosedur yang ada dengan teliti sesuai dengan langkah kerja yang tersedia.
3. Penggunaan laptop / desktop PC dengan langkah-langkah yang benar dan baik pada saat menyalakan ataupun mematkannya.

G. Langkah Kerja

Buat sebuah *project* baru dengan nama *Histogram*, lalu desain GUI seperti yang terlihat pada Gambar 1 dan konfigurasi EmguCV.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan <i>Histogram</i>	No. Job : 9
	Revisi : 01	<i>Equalization</i> Menggunakan EmguCV	4 × 50 menit



Gambar 1. GUI untuk menerapkan *histogram equalization*

Atur **Size Mode** dari semua *imageBox* menjadi **StretchImage** agar ukuran citra digital yang akan ditampilkan menyesuaikan ukuran dari *imageBox*. **Double Click** pada **Form** untuk menambahkan *eventHandler Form_Load* serta masuk ke *code panel*. Dua *imageBox* digunakan untuk menampilkan citra digital awal sebelum diproses dan untuk menampilkan citra digital hasil *histogram equalization*.

Masukkan *code* berikut untuk menerapkan *histogram equalization*:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

using Emgu.CV;
using Emgu.CV.CvEnum;
using Emgu.CV.Structure;
using Emgu.CV.Util;
using Emgu.Util;
```

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan <i>Histogram</i>	No. Job : 9
	Revisi : 01	<i>Equalization</i> Menggunakan EmguCV	4 × 50 menit

```

namespace ROI
{
    public partial class Form1 : Form
    {
        Capture cap = null;
        Mat source = new Mat();
        Mat histogram = new Mat();

        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            cap = new Capture();
            cap.ImageGrabbed += cap_ImageGrabbed;
            cap.Start();
        }

        void cap_ImageGrabbed(object sender, EventArgs e)
        {
            cap.Retrieve(source);

            CvInvoke.CvtColor(source, source, ColorConversion.Bgr2Gray);
            CvInvoke.EqualizeHist(source, histogram);

            imageBox1.Image = source;
            imageBox2.Image = histogram;
        }
    }
}

```

H. Pengujian

1. Pengujian dilakukan dengan menjalankan *project* yang telah dibuat.
2. Amati hasil pemuatan setelah *project* dijalankan.
3. Isikan hasil pengamatan dengan kondisi pencahayaan yang berbeda.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan <i>Histogram</i>	No. Job : 9
	Revisi : 01	<i>Equalization</i> Menggunakan EmguCV	4 × 50 menit

Tabel penerapan *histogram* dengan kondisi pencahayaan yang berbeda.

No	Citra Asli	Kondisi Cahaya		
		Redup	Terang	Sangat Terang
1				
2				
3				
4				
5				

I. Tugas

Buatlah sebuah *resume* mengenai praktikum *histogram equalization* yang membahas bagaimana hasil dari proses *histogram equalization* pada berbagai kondisi cahaya, mulai dari pengaruh yang dihasilkan sampai analisisnya!

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan <i>Optical Flow</i> <i>Farneback</i> Menggunakan EmguCV	No. Job : 10
	Revisi : 01		4 × 50 menit

A. Kompetensi

Memahami cara kerja/prinsip *optical flow farneback* menggunakan EmguCV.

B. Indikator

1. Mahasiswa dapat membuat GUI untuk *optical flow farneback*.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk *optical flow farneback*.
3. Mahasiswa dapat menerapkan *optical flow farneback* dengan parameter yang berbeda menggunakan EmguCV.

C. Tujuan

1. Mahasiswa dapat membuat GUI untuk *optical flow farneback* dengan baik dalam waktu yang singkat.
2. Mahasiswa dapat menggunakan *library* EmguCV untuk *optical flow farneback* dengan baik dan benar dalam waktu yang singkat.
3. Mahasiswa dapat menerapkan *optical flow farneback* dengan parameter yang berbeda menggunakan EmguCV dengan baik dan benar dalam waktu yang singkat.

D. Dasar Teori

1. *Optical Flow Farneback*

Optical Flow merupakan salah satu teknik yang sering digunakan dalam *Computer Vision*. Dengan memilih beberapa titik kontur yang ada, sudah bisa digunakan untuk meng-*track* titik kontur tersebut secara terus menerus. Ketika titik kontur terdeteksi, sistem akan menghitung nilai vektor perpindahan dan menunjukkan gerakan pada titik kontur yang bersangkutan. Metode yang paling terkenal untuk melakukan perhitungan nilai vektor perpindahan tersebut adalah algoritma Lucas-Kanade dan algoritma Farneback.

Algoritma Lucas-Kanade digunakan untuk meng-*track* titik kontur tertentu saja. Sedangkan algoritma Farneback digunakan untuk meng-*track* semua titik kontur yang ada pada citra digital. Dalam algoritma Farneback, semua piksel yang ada akan diproses untuk dicari nilai vektor perpindahannya. Dikarenakan prosesnya yang membutuhkan

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan <i>Optical Flow</i> Farneback Menggunakan EmguCV	No. Job : 10
	Revisi : 01		4 × 50 menit

semua piksel yang ada, maka tidak heran jika algoritma Farneback memiliki tingkat keakuratan yang cukup tinggi. Namun disamping itu, algoritma ini akan membuat sistem menjadi memiliki beban yang cukup tinggi, jika diatur untuk penggunaan dengan keakuratan yang tinggi.

2. *Optical Flow* Farneback menggunakan EmguCV

Fungsi *Optical Flow* Farneback yang terdapat pada EmguCV harus memiliki dua masukan citra yang berupa *single-channel*, dimana satu citra berupa citra pada *frame* sebelumnya dan satu citra pada *frame* sekarang. Hasil perhitungan akan dimasukkan ke dalam dua variabel yang memiliki fungsi menyimpan nilai vektor perpindahan dalam koordinat x dan nilai vektor perpindahan dalam koordinat y. Berikut penjelasan lengkap dari fungsi `CvInvoke.CalcOpticalFlowFarneback` pada EmguCV :

```
public static void CalcOpticalFlowFarneback(
    Image<Gray, byte> prev0,
    Image<Gray, byte> next0,
    Image<Gray, float> flowX,
    Image<Gray, float> flowY,
    double pyrScale,
    int levels,
    int winSize,
    int iterations,
    int polyN,
    double polySigma,
    OpticalflowFarnebackFlag flags
)
```

Dimana :

`prev0` : masukan citra digital *single-channel* pada *frame* sebelumnya
`next0` : masukan citra digital *single-channel* pada *frame* sekarang
`flowX` : hasil perhitungan vektor perpindahan pada koordinat x
`flowY` : hasil perhitungan vektor perpindahan pada koordinat y
`pyrScale` : nilai skala citra pada lapisan berikutnya untuk membentuk piramida pada citra, harus lebih kecil dari 1. Jika `pyrScale = 0.5`, berarti lapisan berikutnya dua kali lebih kecil dari lapisan sebelumnya
`levels` : Jumlah lapisan piramida, termasuk citra awal. `levels = 1` berarti tidak ada lapisan tambahan yang dibuat dan hanya citra asli yang digunakan
`winSize` : semakin besar nilai mampu meningkatkan kehandalan algoritma terhadap gangguan pada citra digital dan mampu membuat kecepatan pendeteksian

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan <i>Optical Flow</i> Farneback Menggunakan EmguCV	No. Job : 10
	Revisi : 01		4 × 50 menit

meningkat, namun juga akan menghasilkan bidang gerakan menjadi lebih kabur

iterations : jumlah iterasi yang dilakukan algoritma pada setiap tingkat piramida

polyN : nilai yang lebih besar berarti citra akan dianggap sebagai permukaan yang lebih halus, menghasilkan algoritma yang lebih kuat dan bidang gerakan yang lebih kabur. Biasanya, *polyN* = 5 atau 7

polySigma : Untuk *polyN* = 5 dapat menggunakan *polySigma* = 1,1, untuk *polyN* = 7 nilai yang baik adalah *polySigma* = 1,5

flags : metode yang digunakan

E. Alat dan Bahan

1. Laptop / Desktop PC.
2. Library EmguCV versi 3.1.0.2282.
3. Software Microsoft Visual Studio 2012.
4. Kamera Webcam.

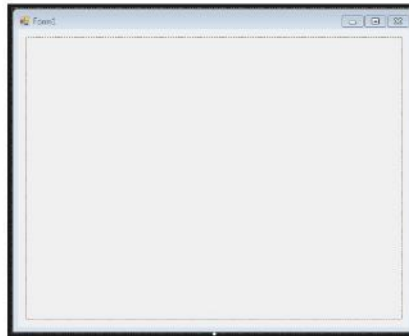
F. Keselamatan Kerja

1. Jauhkan peralatan yang tidak dibutuhkan dari meja kerja.
2. Ikuti prosedur yang ada dengan teliti sesuai dengan langkah kerja yang tersedia.
3. Penggunaan laptop / desktop PC dengan langkah-langkah yang benar dan baik pada saat menyalakan ataupun mematikannya.

G. Langkah Kerja

Buat sebuah *project* baru dengan nama *OpticalFlow*, lalu desain GUI seperti yang terlihat pada Gambar 1 dan konfigurasi EmguCV.

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan <i>Optical Flow</i> <i>Farneback</i> Menggunakan EmguCV	No. Job : 10
	Revisi : 01		4 × 50 menit



Gambar 1. GUI untuk menerapkan *Optical Flow Farneback*

Atur *Size Mode* dari *imageBox* menjadi *StretchImage* agar ukuran citra digital yang akan ditampilkan menyesuaikan ukuran dari *imageBox*. *Double Click* pada *Form* untuk menambahkan *eventHandler Form_Load* serta masuk ke *code panel*.

Masukkan *code* berikut untuk menerapkan *Optical Flow Farneback*:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

using Emgu.CV;
using Emgu.CV.CvEnum;
using Emgu.CV.Structure;
using Emgu.CV.Util;
using Emgu.Util;
```

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan <i>Optical Flow</i> Farneback Menggunakan EmguCV	No. Job : 10
	Revisi : 01		4 × 50 menit

```

namespace opticalFlow
{
    public partial class Form1 : Form
    {
        Capture cap = null;
        Mat source = new Mat();
        Mat gray = new Mat();
        Mat output = new Mat();
        Image<Gray, Byte> current;
        Image<Gray, Byte> previous;

        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            cap = new Capture();
            cap.ImageGrabbed += cap_ImageGrabbed;
            cap.Start();
        }

        void cap_ImageGrabbed(object sender, EventArgs e)
        {
            cap.Retrieve(source);
            output = source.Clone();
            CvInvoke.CvtColor(source, gray, ColorConversion.Bgr2Gray);
            current = gray.ToImage<Gray, Byte>();
            Image<Gray, float> flowX = new Image<Gray, float>(current.Width, current.Height);
            Image<Gray, float> flowY = new Image<Gray, float>(current.Width, current.Height);
            if (previous != null)
            {
                CvInvoke.CalcOpticalFlowFarneback(current, previous, flowX, flowY, 0.5, 5, 15, 1, 7, 1.5, 0);
                Point from = new Point();
                Point to = new Point();
                for (int i = 0; i < flowY.Rows; i += 10)
                {

```

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan <i>Optical Flow</i> Farneback Menggunakan EmguCV	No. Job : 10
	Revisi : 01		4 × 50 menit

```

for (int j = 0; j < flowX.Cols; j += 10)
{
    to.X = (int)flowX.Data[i, j, 0];
    to.Y = (int)flowY.Data[i, j, 0];
    from.X = j;
    from.Y = i;
    to.X = from.X + to.X;
    to.Y = from.Y + to.Y;
    CvInvoke.Line(output, from, to, new
    Bgr(Color.Red).MCvScalar, 1);
}
}
previous = current.Clone();
imageBox1.Image = output;
}
}
}

```

H. Pengujian

1. Pengujian dilakukan dengan menjalan *project* yang telah dibuat.
2. Amati hasil pemuatan setelah *project* dijalankan.
3. Isikan hasil pengamatan dengan berbagai gerakan tubuh yang berbeda.

Tabel penerapan *Optical Flow Farneback* dengan kondisi pencahayaan yang berbeda.

No	Keluaran	Deskripsi Gerakan (Kecepatan, Arah, Jarak)
1		
2		
3		
4		
5		
Kesimpulan :		

	PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA		
	LABSHEET PENGINDERAAN VISUAL ROBOT		
	Semester V	Penerapan <i>Optical Flow</i> <i>Farneback</i> Menggunakan EmguCV	No. Job : 10
	Revisi : 01		4 × 50 menit

I. Tugas

Terapkan *Optical Flow Farneback* dengan menggunakan nilai parameter `levels` dan `winSize` yang berbeda, dengan gerakan yang berbeda pula, dengan ketentuan pada tabel sebagai berikut :

No	Deskripsi Gerakan	Levels	winSize	Keluaran
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
Kesimpulan :				

Lampiran 10 Dokumentasi

