

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Pada perkembangan sistem pengolahan citra dengan kamera, pembuatan *ROFATION* tidak lepas dari penelitian – penelitian yang telah dilakukan oleh orang lain yang terkait dengan pengolahan citra seperti pada penelitian yang dilakukan Asep Nana H, dkk (2014), Wisnu Rizky Kurniawan (2015), Anhar Widodo, dkk (2017).

Pertama, penelitian yang dilakukan oleh Asep Nana H, dkk (2014) memaparkan tentang menanamkan kecerdasan pada sebuah kamera untuk melakukan segmentasi citra yang dapat mendeteksi objek warna pada aplikasi pengambilan bentuk citra *rectangle*. Pada penelitian tersebut berfokus pada warna yang dijadikan *input* dalam penggambaran daerah yang diinginkan (*Region Of Interst*) melalui proses deteksi warna dan tracking warna, sehingga dapat dilakukan pengambilan gambar dalam bentuk tertentu.

Kedua, penelitian dilakukan oleh Wisnu Rizky Kurniawan (2015) memaparkan bagaimana penggunaan kamera sebagai penghitung jumlah kendaraan bermotor. Penelitian tersebut berfokus pada objek kendaraan bermotor yaitu dengan menggunakan metode *background subtractor*, deteksi tepian kontur, *size thresholding*, serta *line intersect counting*. Hasil dari penelitian tersebut adalah dapat menghitung dan mengklasifikasi jenis kendaraan.

Ketiga, penelitian dilakukan oleh Anhar Ari Widodo, dkk (2017) memaparkan tentang kamera yang digunakan sebagai sistem pemantau. Penelitian tersebut menjelaskan bagaimana implementasi pengolahan citra untuk mengidentifikasi objek bergerak pada sistem pemantauan. Objek yang teridentifikasi akan dikirim ke *cloud storage* dan hanya dapat diakses melalui aplikasi pada *mobile device* sehingga hasil dari alat ini dapat mencocokkan kebenaran identifikasi serta dapat memantau objek yang tertangkap oleh kamera.

Dengan penelitian – penelitian yang telah dilakukan, *ROFATION* berfokus pada penggunaan kamera pada pengolahan citra untuk melakukan pendeteksian dan pelacakan wajah yang diterapkan pada sebuah mikrofon.

2.2 Dasar Teori

2.2.1 Suara

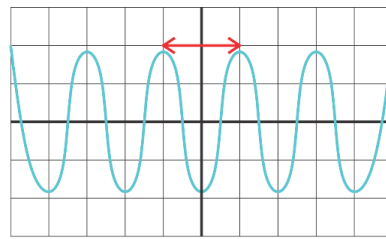
Suara merupakan pemampatan mekanis atau gelombang *longitudinal* yang merambat melalui medium. Medium atau zat perantara ini dapat berupa zat cair, padat, gas sehingga suaranya atau gelombang bunyi dapat merambat melalui air, batu bara, atau udara. Secara ilmiah suara dapat diartikan dengan kecepatan osilasi atau frekuensi dimana satuannya dalam Hertz (Hz) dan kenyaringan atau amplitudo bunyi yang memiliki satuan nilai dalam Desibel (dB)

Manusia dapat mendengar bunyi saat ada gelombang bunyi, dimana ketika ada getaran di udara atau medium lainnya dapat sampai ke gendang telinga manusia. Batas frekuensi bunyi yang dapat didengar oleh telinga manusia sekitar 20 Hz sampai 20 kHz pada amplitudo umum dengan berbagai variasi dalam kurva responsnya. Suara dengan frekuensi diatas 20 kHz disebut ultrasonik sedangkan frekuensi di bawah 20 Hz disebut infrasonik.

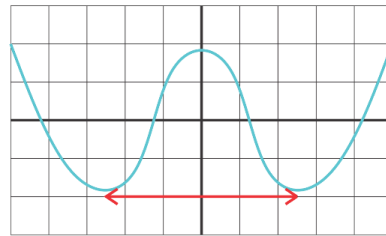
Suara dibedakan menjadi dua yaitu suara periodik dan non periodik. Suara periodik merupakan suara yang terjadi secara berkelanjutan dan getaran yang dihasilkan sama contohnya seperti bernyanyi, suara burung, dan alat musik. Sedangkan suara non periodik merupakan suara yang terjadi tidak berkelanjutan dengan bentuk gelombangnya kotak – kotak seperti suara piring pecah, air jatuh, dan hentakan kaki. Adapun tiga hal yang berkaitan dengan suara adalah:

A. Frekuensi

Frekuensi merupakan banyaknya getaran dalam satu detik dengan satuan Hertz (Hz).



Frekuensi tinggi, panjang gelombang pendek

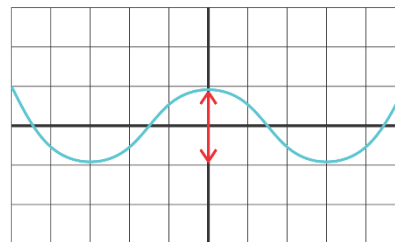


Frekuensi rendah, panjang gelombang panjang

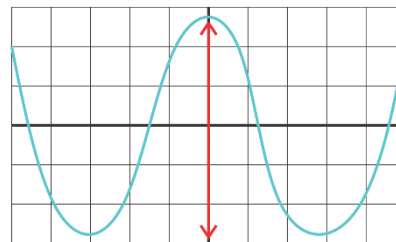
Gambar 2. 1 Frekuensi pada gelombang

B. Amplitudo

Amplitudo merupakan keras lemahnya bunyi atau tinggi rendahnya gelombang dengan satuan Desibel (dB).



Amplitudo kecil



Amplitudo besar

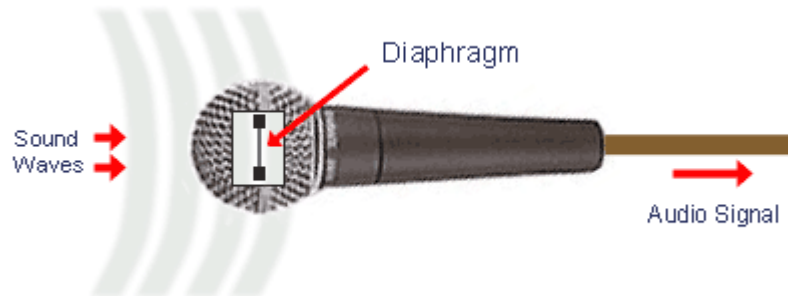
Gambar 2. 2 Amplitudo pada gelombang

C. Velocity

Velocity merupakan kecepatan perambatan gelombang bunyi sampai ke telinga pendengar dengan satuan m/s .

2.2.1.1 Mikrofon

Mikrofon merupakan salah satu perangkat yang dapat mengubah energi dari satu bentuk ke bentuk lainnya (*transduser*), dimana mikrofon dapat mengubah gelombang suara (energi akustik) menjadi sinyal audio (energi listrik). Terdapat berbagai tipe mikrofon dimana masing-masing tipe menggunakan metode yang berbeda dalam mengkonversi energi, namun semua tipe mikrofon tersebut memiliki satu kesamaan yaitu diafragma. Diafragma merupakan sebuah material tipis (berupa kertas, plastik atau alumunium) yang bergetar ketika terkena gelombang suara. Pada mikrofon genggam diafragma terletak didalam kepala seperti pada gambar berikut.



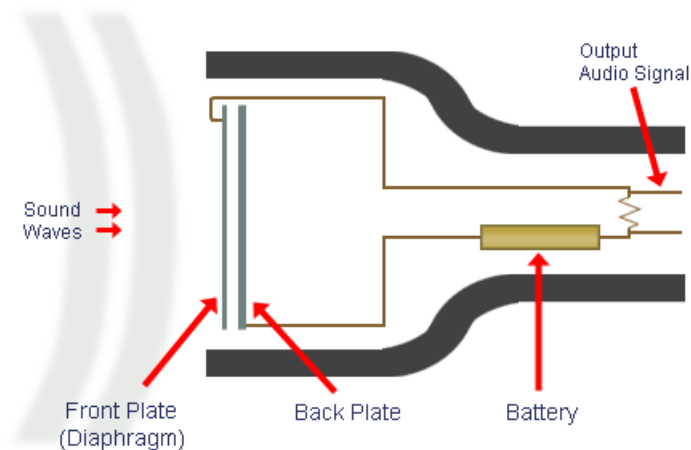
Gambar 2. 3 Mikrofon genggam

Sumber: PT. Goshen Swara Indonesia (<http://www.goshen.co.id>)

Pada saat mikrofon digunakan, diafragma yang ada didalam akan bergetar dan komponen lain dalam mikrofon ikut bergetar. Getaran tersebut akan dikonversi menjadi arus listrik yang kemudian menjadi sinyal audio.

2.2.1.2 Kondensor

Kondensor atau biasa disebut kapasitor, merupakan sebuah komponen elektronik yang menyimpan energi dalam bentuk medan elektrostatis. Mikrofon jenis kondensor ini menggunakan kapasitor untuk mengubah energi akustik menjadi arus listrik dan membutuhkan daya dari baterai ataupun sumber eksternal lain. Sinyal audio yang dihasilkan lebih kuat dan lebih sensitif serta responsif dibandingkan mikrofon dinamis. Mikrofon kondensor dapat menangkap detail-detail kecil pada suara, tapi tidak ideal bekerja pada volume tinggi karena tingkat sensitifitasnya rentan terhadap distorsi.



Gambar 2. 4 Mikrofon kondensor

Sumber: PT. Goshen Swara Indonesia (<http://www.goshen.co.id>)

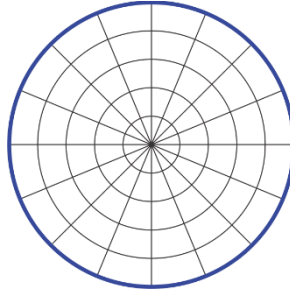
Prinsip kerja dari mikrofon kondensor ini dimana sebuah kapasitor terdiri dari dua buah plat dengan tegangan listrik diantara keduanya. Pada mikrofon kondensor, salah satu plat terbuat dari material yang sangat ringan dan berfungsi sebagai diafragma. Plat diafragma yang terkena gelombang suara akan mengakibatkan getaran diantara kedua plat sehingga menyebabkan terjadinya perubahan jarak antar kedua plat, dimana hal tersebut terjadi suatu perubahan kapasitansi. Dapat diartikan, ketika kedua plat saling merapat, kapasitansi akan meningkat dan terjadi penambahan arus dan ketika kedua plat saling menjauh, kapasitansi akan berkurang dan terjadi pelepasan arus.

2.2.1.3 Directional Properties

Directional properties merupakan karakteristik arah penangkapan suara dari sebuah mikrofon. Properti ini menggambarkan sensitivitas mikrofon terhadap suara dari arah yang berbeda-beda. Terdapat mikrofon yang mampu menangkap suara dari semua arah dengan kualitas yang sama dan mikrofon yang hanya mampu menangkap suara dari satu arah atau kombinasi arah tertentu. *Directional properties* pada mikrofon terbagi 3 dari kemampuan penangkapan suara sebagai berikut.

A. *Omnidirectional*

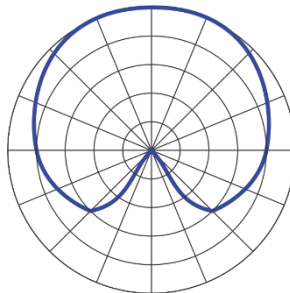
Omnidirectional merupakan kemampuan mikrofon untuk menangkap suara dengan kualitas yang sama dari semua arah. *Omnidirectional* berguna untuk merekam kebisingan sebuah *ambien*, untuk situasi dimana suara datang dari berbagai arah dimana mikrofon diam di satu posisi dengan sumber suar bergerak.



Gambar 2. 5 Pola radiasi penangkapan suara *Omnidirectional*

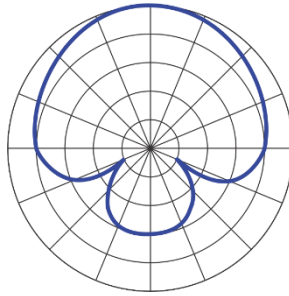
B. *Unidirectional*

Unidirectional merupakan kemampuan mikrofon untuk menangkap suara lebih dominan dari salah satu arah. Kemampuan penangkapan suara secara *Unidirectional* ini dikategori 2 pola radiasi penangkapan suara diantaranya *Cardioid* dan *Hypercardioid*.



Gambar 2. 6 Pola radiasi penangkapan suara *Unidirectional Cardioid*

Pola penangkapan suara *cardioid* yang berarti “berbentuk hati” ini merupakan salah satu pola menangkap suara pada mikrofon. Suara yang ditangkap lebih banyak dari arah depan dan sedikit area dibagian samping serta mengutamakan suara dari arah kemana mikrofon diarahkan namun masih tersedia area untuk pergerakannya dan kebisingan *ambien*

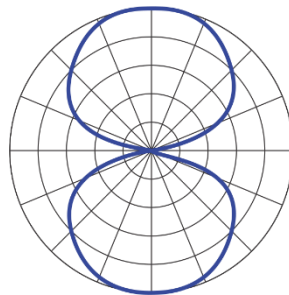


Gambar 2. 7 Pola radiasi penangkapan suara unidirectional hypercardioid

Pola penangkapan suara *Hypercardioid* merupakan tingkatan dari pola *Cardioid*, dimana pola ini terarah dan menghilangkan sebagian besar suara dari samping dan belakang. Mikrofon yang memiliki pola penangkapan suara *Hypercardioid* berguna untuk mengisolasi suara hanya dari satu objek atau satu arah ketika ada banyak kebisingan disekitar.

C. *Bidirectional*

Bidirectional merupakan kemampuan mikrofon untuk menangkap suara dari dua arah yang berlawanan. Pola ini menggunakan pola angka delapan dan mampu menangkap suara secara merata dari dua arah yang berlawanan terlihat seperti gambar 2.8 dibawah.



Gambar 2. 8 Pola radiasi penangkapan suara Bidirectional

Pola penangkapan suara *Bidirectional* ini jarang ditemukan pada situasi yang membutuhkan pola penangkapan suara seperti ini. Salah satu pemanfaatan pola penangkapan suara *Bidirectional* ini adalah pada wawancara dua orang yang saling berhadapan dimana mikrofon berada diantara keduanya.

2.2.2 Robot & Kontrol PID

Menurut kamus bahasa, robot berasal dari kata *robot* yang artinya pekerja, sehingga robot dapat didefinisikan sebagai sebuah alat mekanik yang bekerja terus menerus untuk membantu pekerjaan manusia. Saat ini dunia robotika terus berkembang yaitu bukan hanya robot *humanoid*, robot berkembang dalam berbagai kondisi dan kebutuhan manusia seperti *arm robots, automated guided vehicle, dish washer and cloth washer, wheel LED robot, unmanned vehicle, stand-alone CNC machine millers, humanoid robots, legged robots, caterpillar-tracked robots*. (Ari Sugiharto & Sri Windiyanti, 2017)

Robot merupakan suatu mekanik yang dapat diperintah secara otomatis untuk memproses suatu benda agar dapat bekerja sesuai dengan permintaan tertentu sehingga dapat meringankan pekerjaan manusia. Kontroler *PID* merupakan kontrol cerdas untuk menentukan presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem tersebut. Kontroler *PID* (*Proporsional, Integral, Differential*) memiliki fungsi untuk mengatur kecepatan motor sehingga pergerakan robot menjadi lebih halus. Hal ini disebabkan karena hasil pembacaan citra memiliki nilai yang berubah di setiap robot bergerak. (Basuki Winarno, S.T., M.T. & Meliyana Wahyu Putri Pratama, 2016)

2.2.3 Pengolahan Citra

Pengolahan citra digital merupakan sebuah ilmu yang mempelajari hal - hal yang berkaitan dengan perbaikan kualitas citra. Dengan kata lain pengolahan citra adalah kegiatan memperbaiki kualitas citra agar mudah diterjemahkan oleh manusia atau mesin (komputer). (Ihsan Nugraha Putra Mukhti dkk, 2015) Jenis citra pada umumnya dibagi dua yaitu citra analog dan citra digital. Citra analog merupakan citra yang bersifat berkelanjutan yang diperoleh dari sistem optik yang menerima sinyal analog. Sedangkan citra digital merupakan citra yang dihasilkan melalui digitalisasi terhadap citra analog. Citra digital dibagi menjadi tiga jenis yaitu citra warna, citra *grayscale* dan citra biner. (Abdul Khair Tarigan dkk, 2016)

2.2.3.1 Citra Warna

Citra berwarna merupakan gabungan dari beberapa lapis citra kanal warna yang bertumpuk. Masing-masing lapisan merepresentasikan nilai intensitas warna tertentu terhadap warna gelap sehingga pada citra berwarna setiap *pixel* mempunyai informasi warna tertentu yang merupakan gabungan warna-warna dari citra kanal warna. Misalnya *RGB (Red Green Blue)* mempunyai warna dasar merah, hijau dan biru digabungkan dalam membentuk suatu susunan warna yang luas.

2.2.3.2 Citra Grayscale

Citra *grayscale* adalah warna-warna *pixel* yang berada dalam rentang gradasi warna hitam dan putih. Citra *grayscale* digunakan untuk menyetarakan intensitas warna. Pada intensitas warna merah, hijau dan biru, citra *grayscale* mempunyai nilai *pixel* yang sama.

2.2.3.3 Citra Biner

Dalam citra biner setiap *pixel* hanya mempunyai dua kemungkinan nilai yaitu 0 dan 1. Umumnya, angka nol mewakili warna hitam dan angka satu mewakili warna putih. Citra biner diperoleh melalui proses pemisahan *pixel – pixel* berdasarkan skala keabuan yang dimilikinya. *Pixel* yang memiliki skala keabuan lebih kecil dari nilai batas yang ditentukan diberikan nilai 0, sementara *pixel* yang memiliki skala keabuan yang lebih besar dari batas diubah menjadi bernilai 1. Jadi untuk gambar yang biasanya berwarna hitam putih, nilainya 0 dan 1.

2.2.4 Face Detection

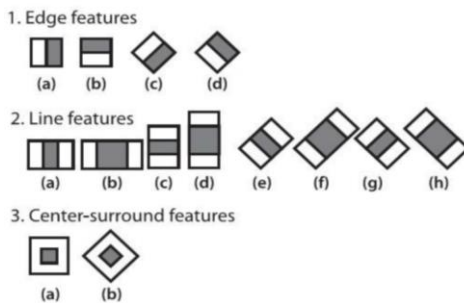
Face Detection adalah salah satu bidang kajian dalam ilmu *computer vision* yang bertujuan untuk mendeteksi ada atau tidaknya wajah didalam sebuah cakupan *frame* citra. *Face Detection* merupakan langkah awal dari pengenalan sebuah wajah yaitu *face recognition*. Pengolahan citra wajah belakangan telah banyak dijadikan studi, dengan maksud menawarkan *service*.

Face detection pada *OpenCV* menggunakan sebuah tipe detektor bernama *Haar Cascade Classifier* yang akan menunjukkan sebuah gambar dengan sebuah tanda apabila terdeteksi wajah. Jika ada sebuah citra dinamis diakses maka *face detector* akan

menguji ke setiap lokasi citra dan mengklasifikasikannya sebagai “wajah” atau “bukan wajah”. Klasifikasi ini dimisalkan dengan sebuah skala tetap untuk wajah, misal 50x50 *pixel*. Jika wajah pada citra lebih besar atau lebih kecil dari *pixel* tersebut, maka *classifier* akan terus menerus jalan beberapa kali untuk mencari wajah. (Evander Alfa Bahana dkk, 2017)

2.2.4.1 Fitur Haar

Fitur ini merujuk kepada sebuah fungsi matematika yang bernama *haar wavelet*



Gambar 2. 9 Bentuk fitur Haar

Sumber: Budi Permana dkk, 2015

Pada tahap awal pengolahan gambar menggunakan *haar cascade*, gambar yang tertangkap oleh kamera akan dilakukan proses *scaling*, *grayscale*, dan *thresholding*. Proses *scaling* berguna untuk mengubah ukuran digital gambar sehingga dapat membuat semua citra digital memiliki ukuran yang sama. Proses merubah citra warna dilakukan pada tahap *grayscale* dan masing – masing *channel pixel* dengan komponen *R*, *G*, dan *B* akan diproses dengan menggunakan rumus sebagai berikut:

$$W = 0.2989 R + 0.5870 G + 0.1140 B$$

Dimana:

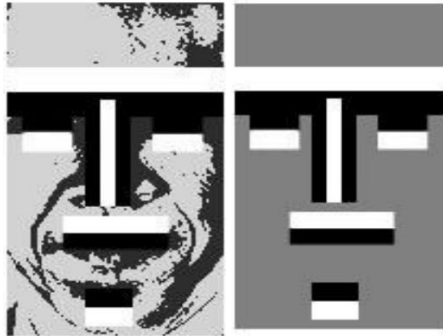
W = *Widht* (lebar *channel pixel*)

R = *Red*

G = *Green*

B = *Blue*

Pendeteksian dengan *Haar Cascade Classifier* menggunakan *haar like feature* dimana untuk mendapatkan nama *casceade classifier* sebagai penentu apakah ada objek atau tidak di dalam setiap *frame* yang diproses perlu dilakukan latihan terlebih dahulu untuk mendapatkan suatu pohon keputusan. Terdapat 3 jenis fitur berdasarkan jumlah persegi Panjang yang terdapat didalamnya seperti gambar 2.4.



Gambar 2. 10 Pencarian wajah dengan fitur Haar

Sumber: Budi Permana dkk, 2015

Memanggil *cascade classifier* merupakan bagian yang digunakan untuk memanggil file dari pendeteksian wajah yang telah dideklarasikan pada global *variable*. Berikut adalah program untuk memanggil *cascade classifier* terlihat pada gambar 2.11.

```
7% video.py - D:\TUGAS AKHIR\File C\opencv\video.py
File Edit Format Run Options Windows Help
import cv2
videoCam = cv2.VideoCapture(0)
face = cv2.CascadeClassifier('D:\File C\download\Software TA\opencv\sources\data\haarcascades\haarcascade_frontalface_default.xml')
eye = cv2.CascadeClassifier('D:\File C\download\Software TA\opencv\sources\data\haarcascades\haarcascade_eye.xml')

while True:
    cond, frame = videoCam.read()

    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    muka = face.detectMultiScale(gray, 1.3, 5)
    for (x,y,w,h) in muka:
        cv2.rectangle(frame, (x,y), (x+w, y+h), (0, 255, 0), 2)
        roi_warna = frame[y:y+h, x:x+w]
        roi_gray = gray[y:y+h, x:x+w]
        mata = eye.detectMultiScale(roi_gray)
        for (mx,my,mw,mh) in mata:
            cv2.rectangle(roi_warna, (mx,my), (mx+mw, my+mh), (255,255,0), 2)

    cv2.imshow('Face', frame)

    k = cv2.waitKey(1) & 0xff
    if k == ord('q'):
        break

videoCam.release()
cv2.destroyAllWindows()
```

Gambar 2. 11 Program untuk memanggil cascade classifier

2.2.4.2 Ruang Warna

Ruang warna *RGB* merupakan tiga warna pokok yang memiliki panjang spektrum paling dominan, yaitu merah (*red*), hijau (*green*), dan biru (*blue*). Masing – masing warna pokok ini memiliki intensitas dengan jumlah 8-bit (0-255). Sehingga ruang warna *RGB* sendiri memiliki intensitas dengan jumlah 24-bit. Berbeda dengan citra *RGB*, *grayscale* merupakan jenis citra yang setiap *pixel* dalam *array*-nya hanya terdiri dari satu komponen warna. Oleh karena itu citra *grayscale* memiliki intensitas sebesar 8-bit (0-255), interval tersebut menunjukkan perbedaan warna keabuan pada setiap interval, semakin mendekati 0 maka citra akan cenderung berwarna hitam, ketika mendekati nilai 255 maka citra akan cenderung berwarna putih. Citra *RGB* dikonversikan menjadi citra *grayscale* dengan cara merata-ratakan ketiga nilai *R*, *G* dan *B* yang dimiliki.

2.2.5 Pergerakan Mikrofon

Pergerakan mikrofon merupakan teknik penangkapan suara untuk mendapatkan posisi yang sesuai dengan mulut wajah/ objek yang tertangkap kamera. Pada pergerakan mikrofon terdapat pergerakan yang dapat dilakukan dalam penangkapan suara dan teknik penangkapan suara menggunakan pergerakan sebagai berikut.

Pan atau *panning* merupakan teknik menggerakkan mikrofon secara mendatar (horizontal) dari kanan ke kiri atau sebaliknya.

- *Pan right* : Pergerakan mikrofon dengan memutar ke arah kanan.
- *Pan left* : Pergerakan mikrofon dengan memutar ke arah kiri.

Tilt atau *tilting* merupakan pergerakan mikrofon secara tegak lurus (vertikal) dari atas ke bawah atau sebaliknya.

- *Tilt up* : Pergerakan mikrofon dengan mengarah ke atas.
- *Tilt down* : Pergerakan mikrofon dengan mengarah ke bawah.

2.2.6 Webcam

Webcam (singkatan dari *web camera*) adalah sebutan bagi kamera *real-time* yang gambarnya bisa diakses atau dilihat melalui *World Wide Web*, *program instant messaging*, atau aplikasi *video call*. Pada umumnya *webcam* dihubungkan ke komputer menggunakan *port USB*. *Frame rate* pada *webcam* mengindikasikan jumlah gambar yang dapat diambil dalam satu detik. *Webcam* adalah sebuah periferal berupa kamera sebagai pengambil citra/ gambar dan mikrofon (optional) sebagai pengambil suara/ audio yang dikendalikan oleh sebuah komputer atau oleh jaringan komputer. Gambar yang diambil oleh *webcam* ditampilkan ke layar monitor dan memiliki *interface* atau *port* yang digunakan untuk menghubungkan *webcam* dengan komputer atau jaringan.



Gambar 2. 12 Webcam Logitech C270

Webcam memiliki jenis-jenis sesuai spesifikasinya yaitu:

A. *Serial and Parallel Port Webcam*

Webcam jenis ini menghasilkan kualitas gambar yang rendah dan *frame rate* yang rendah.

B. *Usb Webcam*

Webcam jenis ini merupakan *webcam* dengan fasilitas *PnP (Plug and Play)* dan dapat dihubungkan ke *port USB* tanpa harus mematikan komputer dengan sistem operasi komputer yang mendukung fasilitas *port USB*.

C. *Firewire and Card Based Webcam*

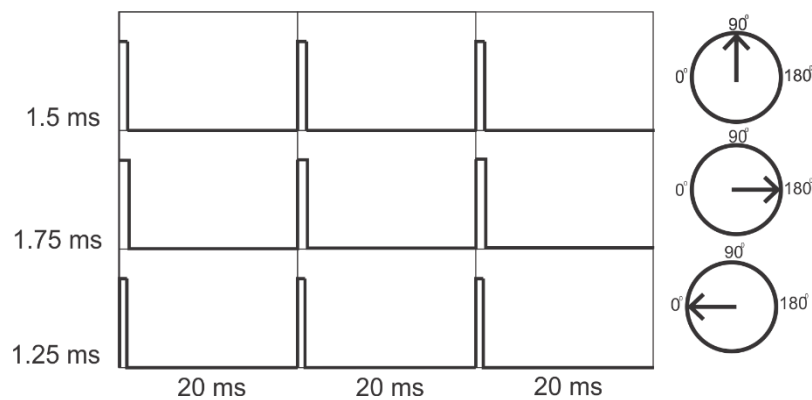
Firewire adalah salah satu teknologi *video capture device* yang diperlukan bagi kamera yang mendukungnya. *Webcam* jenis ini menghasilkan *frame rate* tinggi, yaitu 24 sampai 30 *frame per second (fps)*.

D. *Network and Wireless Camera*

Network Camera adalah perangkat kamera yang tidak memerlukan sama sekali fasilitas komputer, karena dapat langsung terhubung ke jaringan melalui modem. Jenis *webcam* ini bekerja dengan mengirim gambar dan suara langsung menuju jaringan *LAN* atau *line* telepon via modem.

2.2.7 Motor Servo

Motor *servo* adalah motor *DC* yang dilengkapi dengan sistem kontrol yang dapat memberikan umpan balik posisi perputaran motor dari 0 sampai 180 derajat. Selain dapat memberikan umpan balik posisi perputaran motor juga memiliki torsi relatif cukup kuat. Sistem pengkabelan motor servo terdiri atas 3 bagian, yaitu *Vcc*, *GND*, dan kontrol (*PWM= Pulse Width Modulation*). Pemberian *PWM* pada motor *servo* akan membuat *servo* bergerak pada posisi tertentu dan kemudian berhenti (kontrol posisi). (Ragil Febrio Giant dkk, 2015)



Gambar 2. 13 Hubungan lebar pulsa PWM dengan arah putaran motor servo

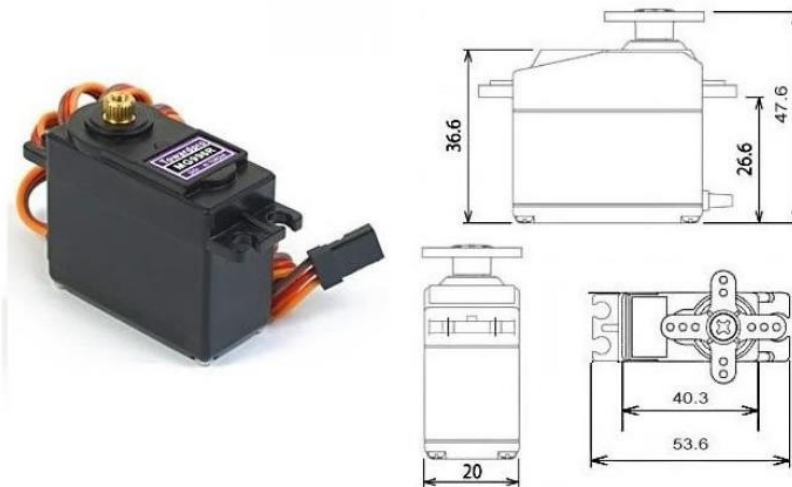
Prinsip utama dari pengendalian motor *servo* adalah pemberian nilai *PWM* pada kontrolnya. Frekuensi *PWM* yang digunakan pada pengontrol motor *servo* selalu 50 Hz

sehingga pulsa dihasilkan setiap 20 ms. Pemberian lebar pulsa 1,5 ms akan membuat motor *servo* berputar ke posisi netral (90 derajat), lebar pulsa 1,75 ms akan membuat motor *servo* berputar 1 derajat mendekati posisi 180 derajat, dan dengan lebar pulsa 1,25 ms motor *servo* akan bergerak ke posisi 0 derajat dimana lebar pulsa akan menentukan posisi *servo* yang dikehendaki.

2.2.7.1 Motor Servo Tower Pro MG 996R

Motor *servo* MG996 ini adalah versi lebih baru dari *servo* motor seri MG946 dan MG995, *servo* motor berkinerja tinggi dengan *gear* logam (*metal gear*), *ball bearing* ganda, 180° rotasi, kabel koneksi sepanjang 30 cm, dan dilengkapi dengan aksesoris untuk digunakan sesuai kebutuhan. (Muhamad Arie Kurniawan, 2016)

Servo motor ini cocok untuk aplikasi yang membutuhkan motor dengan torsi yang memadai hingga 13 kg/cm (batas *stall torque* pada 7,2 Volt). Dibanding pendahulunya (MG995), *servo* ini bekerja dengan lebih akurat, lebih cepat dan responsif, dan berdaya lebih kuat. (Muhamad Arie Kurniawan, 2016)



Gambar 2. 14 Motor Servo Tower Pro MG 996R

Sumber: www.xlsemi.com

Motor *servo* Tower Pro MG996R memiliki tegangan minimum yaitu pada tegangan 4,8 Volt untuk dapat mengoperasikan motor ini, kecepatan operasi motor ini

mencapai 0,17 detik untuk rotasi 60° (pada tegangan 4,8 Volt tanpa beban), dengan batas *stall torque* sebesar 9,4 kg/cm. Untuk batas tegangan maksimum sebesar 7,2 Volt, namun dianjurkan untuk membatasi tegangan catu daya pada tingkat 6 Volt. Pada tegangan 6 VDC, motor ini mampu beroperasi dengan kecepatan 0,14 detik per 60° (konsumsi arus tipikal antara 500 mA ~ 900 mA) dengan batas *stall torque* sebesar 11 kg/cm (konsumsi arus maksimum / *stall current* 2,5 A)

Spesifikasi *motor servo MG996* :

- *Weight*: 55g
- *Dimension*: L 40.7 mm x W 19.7 mm x H 42.9 mm
- *Stall torque*: 9.4 kg/ cm (4.8 V) - 11 kg/ cm(6.0 V)
- *Gear*: Metal gear set
- *Operating speed*: 0.19 sec/ 60 degree (4.8 V) - 0.15 sec/ 60 degree (6.0 V)
- *Servo Plug*: JR (Fits JR and Futaba)

Untuk menjalankan atau mengendalikan motor *servo* berbeda dengan motor *DC*, dimana untuk mengendalikan motor *servo* perlu diberikan sumber tegangan dan sinyal kontrol. Sinyal kontrol didapat dari metode *PWM* (Pulse Width Modulation) yang didapat dari proses konversi *mapping ADC* pada Arduino.

2.2.8 Arduino

Arduino adalah kit elektronik atau papan rangkaian elektronik *open source* yang di dalamnya terdapat komponen utama sebuah *chip* mikrokontroler dengan jenis *AVR* (*Automatic Voltage Regulator*) dari perusahaan *Atmel*. Mikrokontroler tersebut berupa *chip* atau *IC* (*integrated circuit*) yang bisa diprogram menggunakan komputer. (Dendy Pratama dkk, 2016)

Pada mikrokontroler akan tanamkan program dengan tujuan agar rangkaian elektronik dapat membaca masukan, memproses masukan tersebut dan kemudian menghasilkan keluaran sesuai program yang dibuat. Jadi mikrokontroler bertugas sebagai “otak” yang mengendalikan masukan, proses dan keluaran sebuah rangkaian elektronik. Komponen utama Arduino adalah mikrokontroler, maka Arduino pun dapat diprogram menggunakan komputer.



Gambar 2. 15 Arduino Uno R3

Arduino Uno R3 adalah papan mikrokontroler berbasis ATmega328 yang memiliki 14 pin digital *input/ input* (di mana 6 pin dapat digunakan sebagai *input PWM*), 6 *input* analog, *clock speed* 16 MHz, koneksi *USB jack* listrik, *header ICSP*, dan tombol *reset*. (Ghoni Musyahar & Miftakhul Huda, 2017) *Board* Arduino Uno R3 dapat beroperasi pada pasokan daya dari 6 – 20 Volt dengan rentang yang dianjurkan adalah 7 - 12 Volt dan jika *Board* Arduino di beri tegangan lebih dari 12 Volt, maka regulator tegangan akan panas sehingga dapat merusak *board*.

Tabel 2. 1 Spesifikasi Arduino Uno R3

Chip Mikrokontroler	ATmega328P
Tegangan operasi	5V
Tegangan <i>input</i> (via <i>jack DC</i>)	7V – 12V
Tegangan <i>input</i> (limit, via <i>jack DC</i>)	6V – 20V
Digital I/O pin	14 buah, 6 diantaranya menyediakan <i>PWM</i>
Analog <i>input</i> pin	6 buah
Arus DC per pin I/O	20 mA
Arus DC pin 3.3V	50 mA

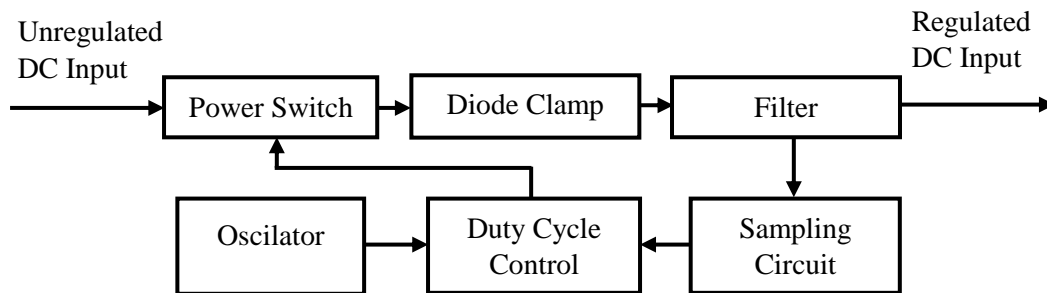
Tabel 2. 2 Spesifikasi Arduino Uno R3 (Lanjutan)

Chip Mikrokontroler	<i>ATmega328P</i>
Memori Flash	32 KB, 0.5 KB telah digunakan untuk bootloader
SRAM	2 KB
EEPROM	1 KB
Clock speed	16 Mhz
Dimensi	68.6 mm x 53.4 mm
Berat	25 g

(<http://ecadio.com/mengenal-dan-belajar-arduino-uno-r3>)

2.2.9 Switching Regulator

Switching regulator merupakan catu daya yang bekerja dengan mengambil potongan kecil energi, sedikit demi sedikit dari sumber tegangan *input* dan akan dipindahkan ke *ouput*. *Switching regulator* bekerja dengan bantuan sebuah saklar listrik dan pengontrol yang dapat mengatur keluarannya. *Switching regulator* mampu menghasilkan tegangan *input* yang lebih tinggi dari tegangan *input* atau polaritas yang berlawanan.



Gambar 2. 16 Blok diagram switching regulator

Pada rangkaian switching regulator memiliki 4 elemen dasar yaitu:

- A. Switching Transistor
- B. Dioda Clamp
- C. LC Filter
- D. Rangkaian Kontrol

Rangkaian *switching regulator* akan menghasilkan tegangan regulasai dengan cara mengalihkan (*switching*) transistor seri *on* atau *off*. Pada dasarnya *switching*

regulator mempunyai frekuensi yang konstan untuk mengalihkan transistor seri, sehingga besarnya frekuensi *switching* harus lebih besar dari 20 kHz agar frekuensi *switching* tersebut tidak dapat didengar oleh manusia. Dimana frekuensi yang terlalu tinggi akan menyebabkan operasi *switching* transistor tidak efisien. *Dioda clamp* yang digunakan pada *switching regulator* memiliki karakteristik *fast recovery rectifier* yang berguna untuk mempertahankan titik kerja dari *switching* transistor dengan cara memotong (*clamp*) tegangan *spike* yang dihasilkan oleh transistor *switching*.

Transistor seri pada *switching regulator* merupakan transistor yang dirangkai seri dengan sumber tegangan dan tegangan *input* yang teregulasi. Untuk penyearah pada *switching regulator* dimana menggunakan penyearah *bridge* dan sebuah filter kapasitor. Rangkaian filter *input* terdapat komponen induktor dan kapasitor dimana nilainya dapat ditentukan dengan persamaan sebagai berikut:

$$L = \frac{(V_{in} - V_0)v_0}{[(0.05 \times V_{in})(1_0 \times f)}$$

$$C = \frac{(V_{in} - V_0)V_0}{[(2L \times f^2)(V_{in} \times v_0)}$$

Dimana:

v_0 = Tegangan *ripple*

V_0 = Tegangan *input* teregulasi

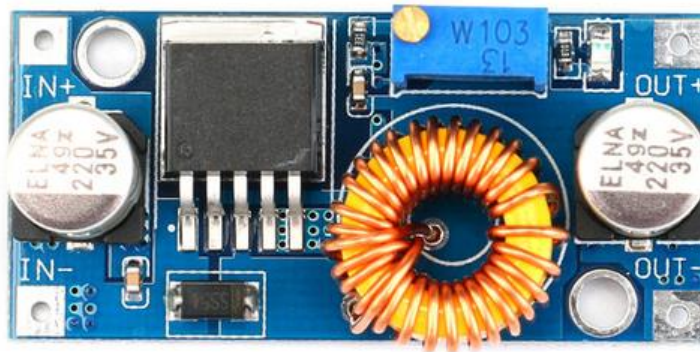
V_{in} = Tegangan *DC* tidak teregulasi

f = Frekuensi *switching*

Prinsip kerja dari *switching regulator* adalah tingginya efisiensi dari *switching regulator* dapat dipengaruhi oleh efisiensi kerja dari *switching* transistor seri. Ketika transistor *switching* dalam kondisi aktif, maka semua tegangan *input* akan dilewatkan filter *LC*. Sedangkan ketika transistor *switching* mati, maka tegangan *input* tidak akan melewati transistor *switching* sehingga tegangan yang masuk ke filter *LC* bernilai nol.

2.2.9.1 Module Regulator XL4005

Module Regulator XL4005 adalah rangkaian modul konverter *DC / DC* dengan frekuensi tetap 300 KHz *fixed-voltage (PWM step-down)* menggunakan *IC Regulator XL4005*, yang mampu menggerakkan beban 5A dengan efisiensi tinggi, derek rendah dan regulasi garis dan beban yang sangat baik. Membutuhkan jumlah minimum komponen eksternal, *regulator* mudah digunakan dan termasuk kompensasi frekuensi internal dan osilator frekuensi tetap.



Gambar 2. 17 Module Regulator XL4005

Modul regulator *XL4005* dapat bekerja dengan suplai tegangan 5-32V dan suhu operasinya -40 - +85 degrees. Pada *module regulator XL4005* menggunakan *ic SMD (Surface Mount Device)* dan terdapat sebuah potensio untuk mengatur tegangan masukannya dari 0.8 – 24V *DC* pada frekuensi kerja 300 KHz sehingga dapat disesuaikan dengan kebutuhan tegangan.

2.2.10 Sketchup

Sketchup adalah perangkat lunak pemodelan 3D yang dirancang untuk arsitek, insinyur sipil, pembuat film, pengembang permainan, aplikasi, dan profesi lain yang terkait bidang 3D.(Irham Fa'idh Faiztyan dkk, 2015) Keunggulan *Sketchup* dibanding perangkat lunak lain diantaranya adalah :

- A. Antarmuka yang mudah dimengerti dan menarik, sehingga mudah digunakan bagi pemula sekalipun.
- B. Dukungan berbagai *open source plugin* yang akan memudahkan kinerja *Sketchup*.




C. *Warehouse* atau gudang model 3D yang sangat lengkap dan terorganisir sehingga memudahkan pengguna dalam mencari model 3D.

Adapun *tools* yang terdapat di *Sketchup* beserta fungsinya sebagai berikut:



Tabel 2. 3 *Tools Sketchup*

No	Gambar/ Simbol	Tools	Penjelasan
1		<i>Select</i>	<i>Select</i> memiliki fungsi untuk menyeleksi objek-objek yang ada di area
2		<i>Line</i>	<i>Line</i> berfungsi untuk membuat objek garis lurus
3		<i>Rectangle</i>	<i>Rectangle</i> memiliki fungsi untuk membuat objek berbentuk persegi empat
4		<i>Circle</i>	<i>Circle</i> sama seperti <i>rectangle</i> namun <i>circle</i> dapat membuat objek berbentuk lingkaran
5		<i>Arc</i>	<i>Arc</i> berfungsi sama dengan <i>line</i> , namun <i>arc</i> digunakan untuk membuat objek garis lengkung
6		<i>Make Component</i>	<i>Make component</i> berfungsi membuat objek-objek yang terseleksi menjadi satu bagian/komponen
7		<i>Eraser</i>	Berbedanya dengan eraser yang ada pada <i>software</i> 2D, <i>eraser</i> berfungsi hanya untuk menghapus <i>line</i> .

Tabel 2. 4 Tools Sketchup (Lanjutan)

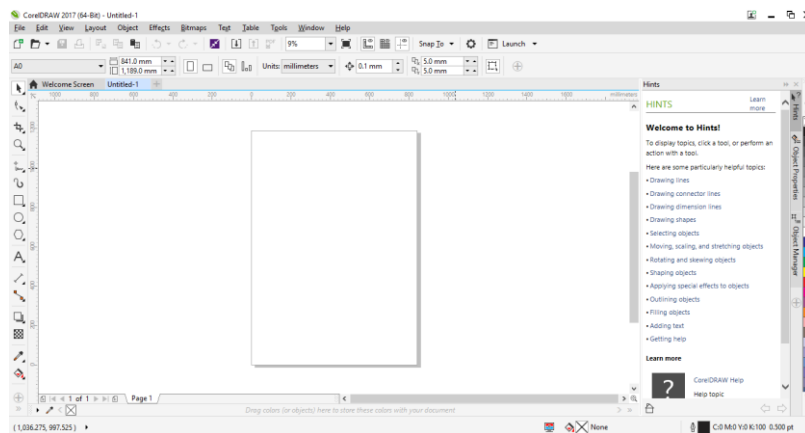
No	Gambar/ Simbol	Tools	Penjelasan
9		<i>Paint Bucket</i>	<i>Paint bucket</i> berfungsi untuk menambahkan material / warna / <i>texture</i> pada sisi objek
10		<i>Push/ Pull</i>	<i>Push/pull</i> memiliki fungsi untuk membuat objek 2D menjadi 3D seperti <i>rectangle</i> jika di- <i>pull</i> akan menjadi sebuah kubus.
11		<i>Move</i>	<i>Move</i> berfungsi untuk memindahkan objek yang dipilih
12		<i>Rotate</i>	<i>Rotate</i> memiliki fungsi untuk merubah rotasi objek yang dibuat.
13		<i>Offset</i>	<i>Offset</i> berfungsi untuk membuat duplikat dari <i>grid</i> sisi objek yang terseleksi
14		<i>Orbit</i>	<i>Orbit</i> berfungsi untuk merotasi <i>view</i> terhadap objek sehingga objek dapat dilihat dari berbagai arah
15		<i>Pan</i>	<i>Pan</i> memiliki fungsi untuk memindahkan posisi <i>view</i> terhadap objek

Tabel 2. 5 Tools Sketchup (Lanjutan)

No	Gambar/ Simbol	Tools	Penjelasan
16		<i>Zoom</i>	<i>Zoom</i> berfungsi untuk memperbesar atau memperkecil <i>view</i> terhadap objek
17		<i>Zoom Extens</i>	<i>Zoom extens</i> berfungsi ketika objek di <i>zoom in</i> maka <i>zoom extens</i> dapat mengembalikan <i>view</i> seluruh objek yang ada pada area/ <i>zoom out</i> ke semua objek

2.2.11 CorelDraw

CorelDraw adalah sebuah program komputer yang melakukan *editing* pada garis *vector*. Program ini dibuat oleh *Corel*, sebuah perusahaan *software* yang berkantor pusat di Ottawa, Kanada. *CorelDraw* memiliki kegunaan untuk mengolah gambar, oleh karena itu banyak digunakan pada pekerjaan dalam bidang publikasi atau percetakan ataupun pekerjaan di bidang lain yang membutuhkan proses visualisasi. (Erwin Prawijaya & Roberta Zulfhi Surya, 2017)



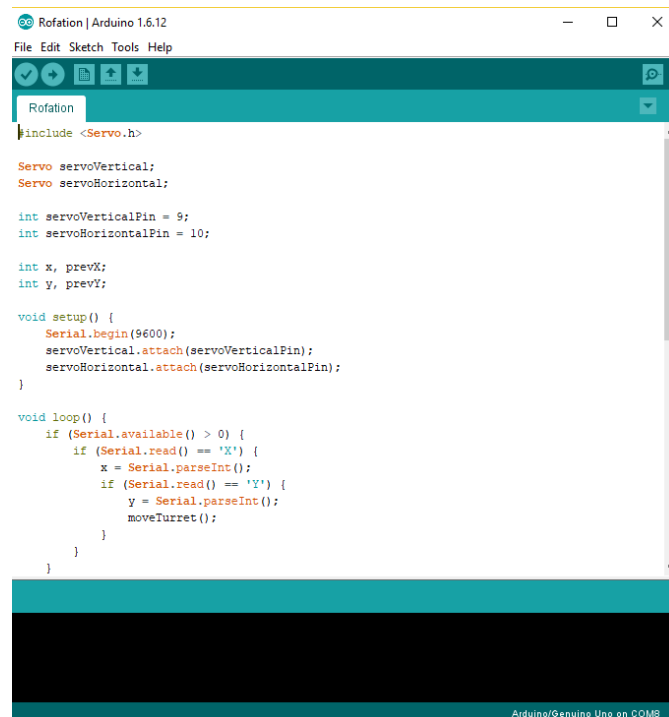
Gambar 2. 18 Tampilan CorelDraw 2017

CorelDraw yang memiliki fungsi untuk mengolah gambar dapat menciptakan dasain seperti logo atau simbol dua dimensi karena kemudahannya dalam mengolah garis dan warna. Selain desain logo dapat juga membuat desain *cover* buku dimana desain sampul dan teknik pewarnaan lebih detail akan terlihat jelas menggunakan *CorelDraw*. Desain tidak akan sempurna tanpa adanya *toolbox* yang terdapat pada *CorelDraw*, dimana *toolbox* dapat diartikan kotak perkakas yang berisi *tool-tool* yang memiliki fungsi membuat dan memodifikasi objek. Beberapa *tool* dasar yang ada pada *toolbox CorelDraw*.

- A. *Pick Tool*
- B. *Shape Tool*
- C. *Crop Tool*
- D. *Zoom*
- E. *Freehand*
- F. *Rectangle*
- G. *Ellipse*
- H. *Polygon*
- I. *Text tool*

2.2.12 Arduino IDE

Arduino *IDE* (*Integrated Development Environment*) adalah *software* yang disediakan di situs arduino.cc yang ditujukan sebagai perangkat pengembangan *sketch* yang digunakan sebagai program di mikrokontroler Arduino. (Rizka Septiandoyo Nugroho, 2017) Arduino *IDE* merupakan perangkat lunak yang dijalankan dengan menggunakan *Java* dan terdiri dari beberapa fitur seperti *editor* program, *compiler*, dan *uploader*. Penulisan dan mengedit program dilakukan pada *editor* program dengan menggunakan bahasa *processing* atau bahasa C. Bahasa C yang dibuat pada *editor* program akan di *compile* yang bertujuan untuk mengubah kode program bahasa C menjadi bahasa mesin dalam bentuk **.hex* (*hexadecimal*). Program yang sudah di-*compile* akan di *upload* yang berfungsi mentransfer kode-kode biner dari komputer ke dalam memori yang ada pada *board Arduino*.



```
Arduino IDE - Rofation | Arduino 1.6.12
File Edit Sketch Tools Help
Rofation
#include <Servo.h>

Servo servoVertical;
Servo servoHorizontal;

int servoVerticalPin = 9;
int servoHorizontalPin = 10;

int x, prevX;
int y, prevY;

void setup() {
  Serial.begin(9600);
  servoVertical.attach(servoVerticalPin);
  servoHorizontal.attach(servoHorizontalPin);
}

void loop() {
  if (Serial.available() > 0) {
    if (Serial.read() == 'X') {
      x = Serial.parseInt();
      if (Serial.read() == 'Y') {
        y = Serial.parseInt();
        moveTurret();
      }
    }
  }
}
```

Arduino/Genuino Uno on COM8

Gambar 2. 19 Tampilan Arduino IDE

Pada gambar 2.16 menampilkan tampilan dari Arduino IDE dan terdapat kode program yang dituliskan pada teks editor. Teks editor memiliki fitur untuk cutting/paste dan searching/ replacing sehingga dapat memudahkan dalam penulisan kode program. Arduino IDE memiliki message box yang berada di bawah teks editor yang berfungsi untuk menampilkan status apakah pada program terdapat error, gagal compile dan keterangan upload program. Ketika USB Arduino sudah terpasang dengan komputer maka pada Arduino IDE bagian bawah menunjukkan board yang terkonfigurasi beserta COM ports yang digunakan. Arduino IDE dapat bekerja dengan adanya tools dan masing-masing memiliki fungsinya.

A. Verify

Verify memiliki fungsi untuk memeriksa program atau checking program yang sudah dibuat untuk mengetahui ada kesalahan atau program sudah sesuai.



Gambar 2. 20 Simbol Verify

B. Upload

Upload berfungsi untuk mentrasfer kode program ke *Arduino* dengan bahasa pemrograman yang sudah berubah supaya dapat dipahami oleh mesin/ *Arduino*.



Gambar 2. 21 Simbol Upload

C. New

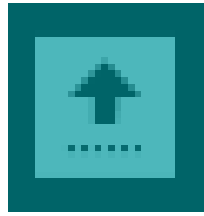
New sama dengan *software* lainnya dimana berfungsi membuat lembaran kerja baru atau pada *Arduino IDE* membuat *sketch* baru.



Gambar 2. 22 Simbol New

D. Open

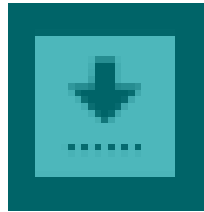
Open pada umumnya sama seperti *software* yang lain dimana berfungsi untuk membuka *file* yang sudah dibuat atau *sketch* yang pernah dibuat untuk dilakukan *editing* dan *upload* ke *Arduino*



Gambar 2. 23 Simbol Open

E. *Save*

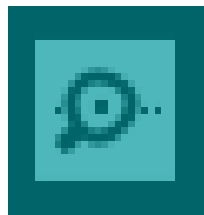
Save berfungsi untuk menyimpan *file sketch* yang sudah dibuat



Gambar 2. 24 Simbol Save

F. *Serial Monitor*

Serial monitor memiliki fungsi untuk menampilkan data yang dikirimkan dan dipertukarkan antara *Arduino* dengan *sketch program* pada *port* serialnya yang menampilkan nilai proses, pembacaan, dan pesan *error* tanpa menggunakan *LCD*.

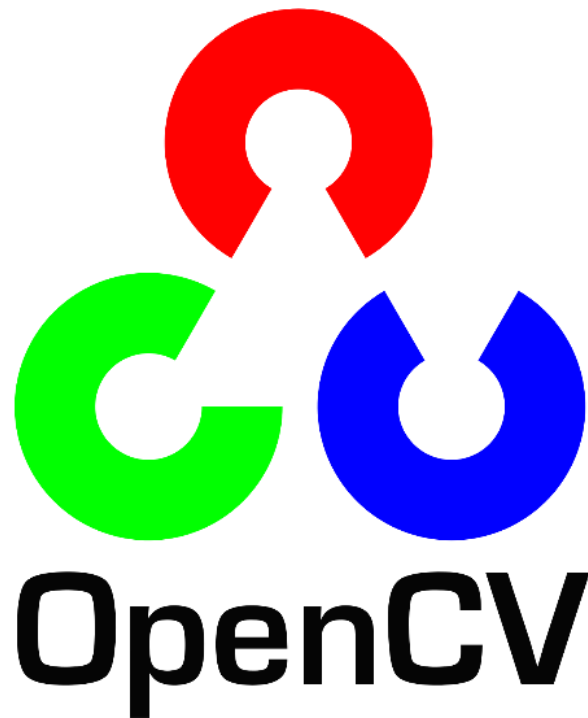


Gambar 2. 25 Simbol Serial Monitor

2.2.13 OpenCV

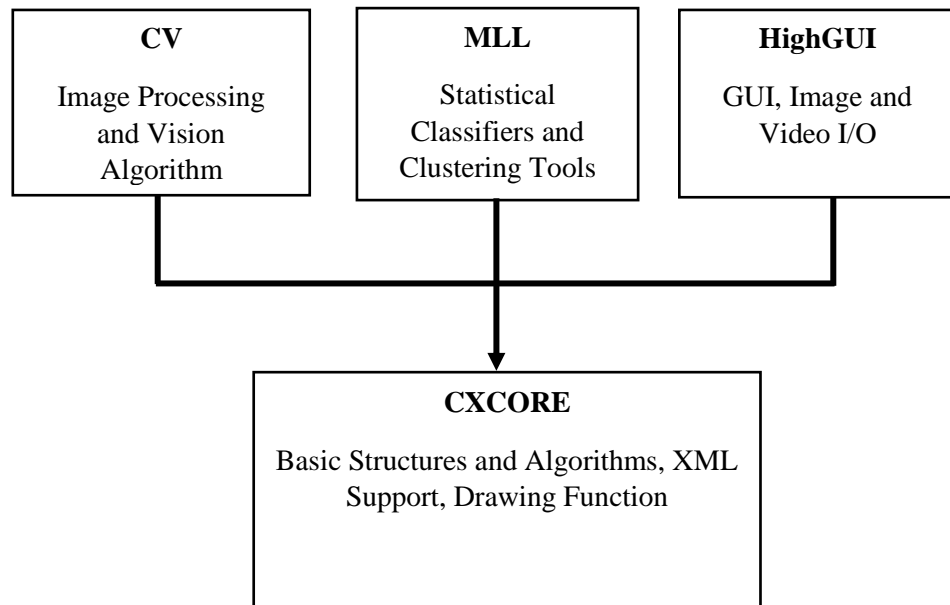
Open Computer Vision (OpenCV) adalah sebuah *Application Programming Interface (API) library* yang digunakan pada pengolahan citra *computer vision*. *Computer Vision* merupakan bidang ilmu pengolahan citra (*Image Processing*) yang memungkinkan komputer dapat melihat seperti manusia, dengan *vision* tersebut komputer dapat mengambil keputusan, melakukan aksi, dan mengenali terhadap suatu

objek. Pengimplementasian dari *Computer Vision* adalah *Face Recognition*, *Face Detection*, *Face/ Object Tracking*, *Road Tracking*. *OpenCV* menggunakan bahasa pemrograman *C++* yang banyak digunakan oleh banyak ilmuwan, pengembang, dan *programmer* untuk mengembangkan sebuah program *pattern recognition*. (Fadli dkk, 2017)



Gambar 2. 26 Simbol OpenCV

OpenCV memiliki 4 *library* dengan fungsi yang berbeda yaitu *CV*, *MLL*, *Highgui*, dan *CXCORE*. *CV* memiliki fungsi untuk algoritma *image processing* dan *vision* dan untuk menampilkan *GUI*, *image* dan *video I/ O* menggunakan *Highgui*. *Library CXCORE* pada *OpenCV* digunakan untuk membantu struktur data, *support XML* dan fungsi-fungsi grafis dan *openCV* juga dilengkapi *Machine Learning Library (MLL)* dimana *MLL* memiliki algoritma seperti *naivebayes classifier*, *k-nearest neighbor algorithm*, *support vector machine*, *decision trees*, *boosting*, *random forest*, *expectation maximization*, dan *neural network*.



Gambar 2. 27 Struktur dan konten OpenCV

2.2.14 Python

Python adalah bahasa pemrograman interpretatif multiguna dengan teori perancangan yang berfokus pada tingkat keterbacaan kode. *Python* diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif. (Mufid Ridlo Effendi dkk, 2017)

Python memiliki salah satu fitur yang tersedia adalah dapat manajemen memori secara otomatis karena menggunakan bahasa pemrograman yang dinamis. *Python* umumnya menggunakan bahasa skrip namun pada praktiknya bahasa ini penggunaannya lebih luas yang mencakup konteks pemanfaatan sehingga bahasa skrip tidak digunakan. Penggunaan *python* banyak diterapkan diberbagai keperluan pengembangan perangkat lunak dan dapat bekerja pada berbagai *platform* sistem operasi. Sistem operasi yang dapat menjalankan *python* diantaranya *Linux*, *Windows*, *Mac OS X*, *Java Virtual Machine*, *OS/2*, *Amiga*, *Palm*, dan *Symbian*.

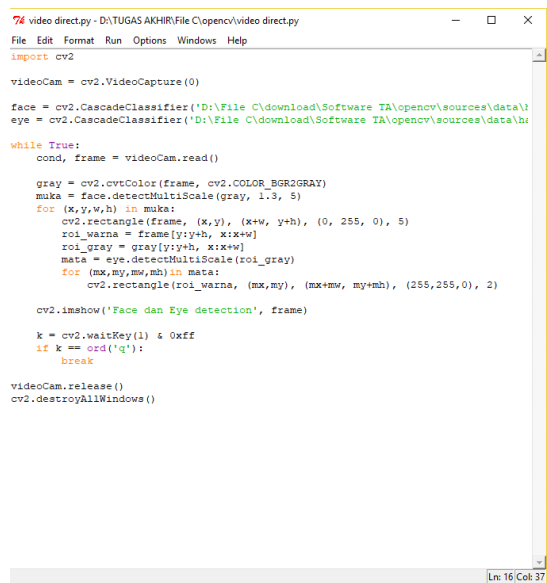
IDLE Python GUI (Integrated Development and Learning Environment Python) merupakan *software editor* untuk membuat *script program python* yang

menggunakan bahasa pemrograman mudah dipahami. *IDLE* memiliki dua tipe jendela utama yaitu jendela *Shell* dan jendela *Editor* dimana kedua jendela ini dapat bekerja secara bersamaan. Jendela *Shell* digunakan untuk mengecek *library* apakah sudah terinstall atau tidak dan sebagai jendela *input*, sedangkan untuk melakukan pembuatan program yaitu menggunakan jendela *Editor*.



```
Python 2.7.5 Shell
File Edit Shell Debug Options Windows Help
Python 2.7.5 (default, May 15 2013, 22:43:36) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> import cv2
>>> import numpy
>>> import serial
>>> |
```

Gambar 2. 28 Tampilan jendela Shell



```
video direct.py - D:\TUGAS AKHIR\File C\opencv\video direct.py
File Edit Format Run Options Windows Help
import cv2
videoCam = cv2.VideoCapture(0)
face = cv2.CascadeClassifier('D:\File C\download\Software TA\opencv\sources\data\haarcascade_frontalface_alt.xml')
eye = cv2.CascadeClassifier('D:\File C\download\Software TA\opencv\sources\data\haarcascade_eye.xml')
while True:
    cond, frame = videoCam.read()
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    muka = face.detectMultiScale(gray, 1.3, 5)
    for (x,y,w,h) in muka:
        cv2.rectangle(frame, (x,y), (x+w, y+h), (0, 255, 0), 5)
        roi_warna = frame[y:y+h, x:x+w]
        roi_gray = gray[y:y+h, x:x+w]
        mata = eye.detectMultiScale(roi_gray)
        for (mx,my,mw,mh) in mata:
            cv2.rectangle(roi_warna, (mx,my), (mx+mw, my+mh), (255,255,0), 2)
    cv2.imshow('Face dan Eye detection', frame)
    k = cv2.waitKey(1) & 0xff
    if k == ord('q'):
        break
videoCam.release()
cv2.destroyAllWindows()
```

Gambar 2. 29 Tampilan jendela Editor