BAB II

LANDASAN TEORI

2.1 Pengolahan Citra Digital

Pengolahan citra digital (*digital image processing*) merupakan suatu proses dengan masukan berupa citra dan hasil dari proses tersebut dapat berupa informasi. Pada awalnya pengolahan citra digital ini dilakukan untuk memperbaiki kualitas citra, namun dengan semakin berkembangnya teknologi pengolahan citra digital tidak hanya memperbaiki kualitas citra, melainkan juga dapat mengekstraksi informasi-informasi yang terdapat dari citra tersebut.

Berdasarkan jenisnya suatu citra dapat diklasifikasikan menjadi 2 jenis yaitu citra kontinyu dan citra digital. Adapun penjelasan mengenai kedua jenis citra adalah sebagai berikut.

2.1.1 Citra Kontinyu

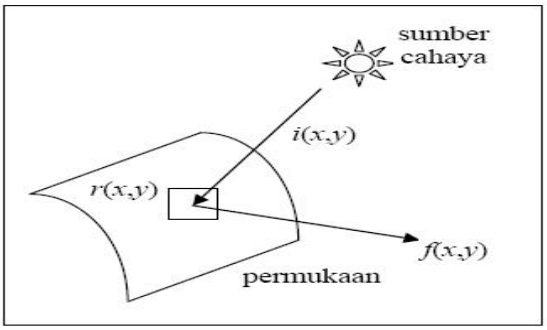
Citra kontinyu adalah fungsi intensitas 2 dimensi f(x,y), adapun x dan y adalah koordinat spasial, dan f pada titik (x,y) merupakan tingkat kecerahan (*brightness*) suatu citra pada suatu titik. Suatu citra diperoleh dari penangkapan kekuatan sinar yang dipantulkan oleh objek. Suatu citra kontinyu dihasilkan dari alat-alat optik yang menerima sinyal analog dari citra tersebut, seperti mata pada manusia, kamera, pemindai (*scanner*) dan sebagainya, sehingga bayangan objek yang disebut citra tersebut terekam. Secara matematis fungsi intensitas cahaya pada bidang dua dimensi disimbolkan dengan $f(x,y)$, yang dalam hal ini :

(x,y) : koordinat pada bidang dua dimensi

f(*x*,*y*) : intensitas cahaya (*brightness*) pada titik (*x*,*y*)

Nilai *f*(*x*,*y*) sebenarnya adalah hasil kali dari :

a. *i*(*x*,*y*) = jumlah cahaya yang berasal dari sumbernya (*ilumination*), nilainya antara 0 sampai tak berhingga, dan
b. *r*(*x*,*y*) = derajat kemampuan objek memantulkan cahaya (*reflection*), nilainya antara 0 dan 1.



Gambar 2.1 Proses Pembentukan Citra [1]

2.1.2 Citra Digital

Citra digital adalah kumpulan data diskrit yang berbentuk matriks dua dimensi yang angka-angkanya dari matriks tersebut menunjukkan tingkat kecerahan dari titik-titik tersebut. Titik-titik yang membangun citra digital inilah yang sering disebut dengan piksel. Untuk mendapatkan suatu citra digital ada beberapa cara untuk mengakusisi citra tersebut, antara lain :

1. menangkap citra dengan menggunakan kamera video. Berdasarkan sensor yang digunakan, kamera video dapat dibagi menjadi 2 jenis, yaitu :
    a. kamera video yang menggunakan sensor CMOS (*Complementary Metal Oxide Semiconduktor*).

b. kamera video yang menggunakan sensor CCD (*Charge Couple Device*).

2. Melakukan konversi citra analog menjadi citra digital, pada umumnya proses konversi ini menggunakan alat-alat bantu seperti *scanner* dan lain-lain.

Citra digital yang berukuran *N x M* dinyatakan dengan matriks yang berukuran *N* baris dan *M* kolom sebagai berikut [3]:

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,M-1) \\ f(1,0) & f(1,1) & \cdots & f(1,M-1) \\ \vdots & \vdots & \ddots & \vdots \\ f(N-1,0) & f(N-1,1) & \cdots & F(N-1,M-1) \end{bmatrix} \quad \ldots\ldots\ldots\ldots\ldots (2.1)$$

Indeks baris *(x)* dan indeks kolom *(y)* menyatakan suatu koordinat titik pada citra, sedangkan *f(x,y)* merupakan intensitas (derajatkeabuan) pada titik *(x,y)*. Masing-masing elemen padacitra digital (berarti elemen matriks) disebut elemen citra (*image element*), elemen gambar (*picture element*) atau piksel (piksel).

2.2 Piksel

Piksel adalah unsur gambar atau representasi sebuah titik terkecil dalam sebuah gambar grafis yang dihitung per inci. Seperti yang telah dijelaskan sebelumnya bahwa citra digital adalah matriks dua dimensi yang besarnya tergantung pada besar kecilnya sampling dan nilai tingkat kecerahan dari setiap titiknya dan *range* nilai dari tingkat kecerahan piksel-piksel yang ada pada cira digital tersebut bergantung pada kuantitas ketika citra tersebut diakuisisi, contohnya : pada citra digital *monocrome* dengan tingkat kuantitas 24 bit, maka citra digital tersebut memiliki *range* 2^{24} tingkat kecerahan atau 16,777,216 tingkat kecerahan dari setiap piksel. Pada umumnya citra digital yang banyak digunakan

adalah citra digital dengan tingkat kuantitas 8 bit, hal ini didasarkan atas pertimbangan besar ukuran berkas dari citra dan kemampuan mata manusia dalam membedakan warna.

2.3 Warna pada Citra Digital

Mata sebagai indra manusia yang berfungsi untuk menerima rangsangan berupa cahaya dan selanjutnya diproses oleh otak dan diubah menjadi suatu persepsi. Sebagai alat indra yang sensitif terhadap cahaya, mata manusia juga memiliki suatu keterbatasan dalam menerima rangsangan cahaya, cahaya yang dapat diterima oleh mata manusia terbatas dari panjang gelombang 400 nm (violet) sampai 770 nm (merah). Oleh karena warna adalah bentuk lain panjang gelombang yang diterima oleh manusia, maka suatu warna dapat di representasikan sebagai fungsi berikut [2].

$$C(x, y, t, \lambda) \quad \text{..... (2.2)}$$

Dengan x dan y adalah posisi dari titik pada citra, t adalah waktu pada citra di tangkap,dan λ panjang gelombang pada titik citra. Berdasarkan warnanya citra digital dapat di bagi menjadi beberapa jenis mode warna, antara lain sebagai berikut.

2.3.1 RGB

Jenis ini merupakan tipe mode warna yang paling umum digunakan dalam dunia citra digital, hampir semua peralatan elektronik yang dapat menampilkan citra digital menggunakan tipe jenis ini seperti monitor berwarna dan kamera. Pada umumnya citra dengan mode warna RGB memiliki masing-masing 8 bit

layer R (*red*), 8 bit *layer* G (*green*), dan 8 bit *layer* B (*blue*). Dengan mengkomposisikan ketiga *layer* warna tersebut akan didapat 256^3 atau 16,777,216 warna, yang tentunya sudah mewakili semua warna yang dapat dilihat oleh mata manusia.

2.3.2 *Grayscale*

Mode warna *grayscale* adalah mode warna yang dihasilkan dari perbandingan nilai-nilai *red* (R), *green* (G), dan *blue* (B) dengan beberapa konstanta tertentu sehingga menghasilkan warna abu-abu. Mode warna ini digunakan untuk mempermudah dalam melakukan proses pengolahan selanjutnya, karena dengan mengubah mode warna RGB menjadi mode warna *grayscale* maka ketiga layer matriks yang sebelumnya memiliki nilai yang berbeda-beda untuk dikonversikan menjadi satu layer matriks, dengan nilai derajat keabuan tertentu hal ini hanya dilakukan pada proses pengolahan citra digital yang tidak membutuhkan perbedaaan warna sebagai informasi yang akan diolah.

Beberapa teknik untuk melakukan konversi dari mode warna RGB ke mode warna *grayscale* dapat dituliskan dengan persamaan sebagai berikut [3]:

$$Gr = (0{,}299 * R) + (0{,}587 * G) + (0.144 * B) \quad \text{..... (2.3)}$$

atau,

$$Gr = (0{,}2125 * R) + (0{,}7154 * G) + (0.0721 * B) \quad \text{..... (2.4)}$$

atau,

$$Gr = \frac{R+G+B}{3} \quad \text{..... (2.5)}$$

Untuk persamaan (2.3) dan (2.4) menggunakan perbandingan untuk mendapatkan nilai *grayscale* dan persamaan (2.5) menggunakan nilai rata-rata

dari ketiga layer matriks. Untuk persamaan (2.3) adalah standar persamaan *grayscale* pada televisi, baik sistem PAL (*Phase Alternation Line*) maupun NTSC (*National Television System Committee*).

2.3.3 Biner (Hitam Putih)

Citra dengan mode warna biner merupakan citra dengan komposisi warna yang hanya terdiri dari 2 jenis warna saja, yakni hitam dan putih. Mode warna ini banyak dipakai dalam proses-proses pengenalan citra yang tidak membutuhkan informasi warna, seperti proses pengenalan morfologi dari suatu objek, penipisan objek, manipulasi bentuk objek, dan lain-lain. Ada beberapa keuntungan dari citra biner yaitu [4]:

- kebutuhan memori menjadi lebih kecil karena nilai derajat keabuan hanya membutuhkan representasi satu bit.
- waktu pemprosesan lebih cepat dibandingkan dengan citra *grayscale* karena banyak operasi pada citra biner yang dilakukan sebagai operasi logika.

Terdapat beberapa alasan perlunya dilakukan perubahan dari citra *grayscale* menjadi citra biner, yaitu [4]:

- untuk mengidentifikasi keberadaan objek, yang direpresentasikan sebagai daerah di dalam citra. Misalnya pemisahan suatu objek dari latar belakangnya.
- untuk lebih memfokuskan pada analisis morfologi, dalam hal ini intensitas piksel tidak terlalu penting dibandingkan bentuknya.

- untuk mengkonversikan citra yang telah ditingkatkan kualitas tepinya ke penggambaran garis-garis tepi.

Tidak seperti pengkonversian mode warna RGB ke *grayscale* yang hanya melibatkan operasi perkalian dengan konstanta-konstanta tertentu maupun dengan mencari rata-ratadari ketiga layer matriks tersebut, untuk mendapatkan nilai biner dari suatu warna piksel maka sebelumnya harus dilakukan teknik *thresholding*, maka nilai biner dari suatu piksel akan diperoleh dengan melakukan klasifikasi nilai piksel berdasarkan nilai *thresholding* tersebut, seperti yang terlihat pada persamaan di bawah ini [5]:

$$fb(x,y) = \begin{cases} 1, jika\ fb(x,y) \leq th \\ 0, jika\ fb(x,y) \geq th \end{cases} \quad \text{..... (2.6)}$$

Dengan *th* adalah nilai *threshold* dari citra.

2.4 *Thresholding*

Thresholding adalah proses untuk mengubah kuantisasi pada citra digital, dengan menggunakan teknik *thresholding* maka derajat keabuan suatu citra dapat diubah sesuai dengan keinginan, selain itu teknik *thresholding* juga sering digunakan untuk mengkelompokan beberapa warna menjadi kelas-kelas tertentu. Adapun persamaan yang digunakan untuk mendeteksi pengkelompokan tersebut adalah sebagai berikut [6]:

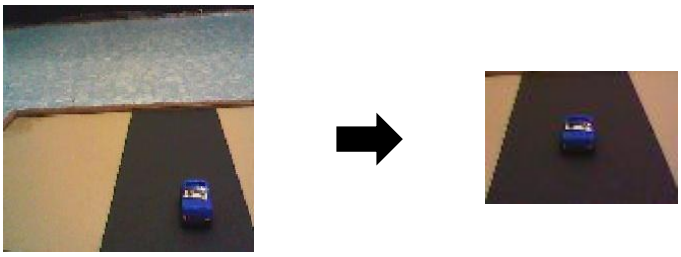
$$x = \frac{256}{th} \quad \text{..... (2.7)}$$

Dengan *x* adalah nilai pengelompokan dari *threshold* dan *th* nilai dari *threshold* itu sendiri.

Contoh penggunaan *threshold* adalah konversi mode warna *grayscale* ke mode warna biner, seperti yang telah dibahas sebelumnya. Untuk melakukan konversi mode warna *grayscale* ke mode warna biner maka nilai tingkat kecerahan dari suatu piksel dikelompokkan menjadi dua kelas, hal ini dilakukan karena citra biner sendiri merupakan citra yang hanya mempunyai dua kemungkinan nilai pada setiap pikselyakni nilai satu dan nilai nol.

2.5 *Cropping*

Proses *cropping* adalah proses menghilangkan beberapa bagian dari citra. Pada proses ini bagian citra yang tidak terseleksi akan dihilangkan.



Gambar 2.2 Proses *cropping* pada citra

2.6 Histogram

Histogram adalah suatu grafik yang menunjukkan distribusi piksel-piksel pada suatu citra digital, jadi dengan histogram dapat diketahui frekuensi kemunculan suatu nilai piksel.Misalnya citra digital memiliki L tingkat kecerahan, yaitu dari nilai 0 sampai L – 1 (misalnya pada citra dengan kuantisasi derajat

keabuan 8 bit, nilai derajat keabuan dari 0 – 255). Secara matematis histogram citra dihitung dengan rumus sebagai berikut [6]:

$$hi = \frac{ni}{n} \quad \text{..... (2.8)}$$

Dimana,

hi = frekuensi kemunculan suatu nilai piksel yang memiliki tingkat kecerahan i.

ni= jumlah piksel yang memiliki tingkat kecerahan i.

n = jumlah seluruh piksel yang ada pada citra.

2.7 *Edge Detection*

Proses deteksi tepi (*edge detection*) akan melakukan konversi menjadi dua macam nilai yaitu intensitas warna rendah atau tinggi, contoh bernilai nol atau satu. Dan proses e*dge detection* berguna untuk menyederhanakan analisis citra dengan mengurangi sejumlah data secara drastis untuk diproses, dengan tetap mempertahankan informasi yang penting mengenai garis-garis batas objek. Gradien adalah hasil pengukuran perubahan dalam sebuah fungsi intensitas, dan sebuah citra dapat dipandang sebagai kumpulan beberapa fungsi intensitas kontinyu sebuah citra.

Perubahan mendadak pada nilai intensitas dalam suatu citra dapat dilacak menggunakan perkiraan diskrit pada gradien. Deteksi tepi akan menghasilkan nilai tinggi apabila ditemukan tepi dan nilai rendah jika sebaliknya. Beberapa metode yang digunakan untuk deteksi tepi, antara lain metode *Robert*, metode *Prewitt*, metode *Sobel*, dan metode *Canny*. Metode deteksi tepi yang digunakan pada penelitian ini yaitu metode ***Canny Edge Detection***. Karena metode *Canny*

edge detection mendeteksi semua tepi-tepi suatu objek. Hal ini mempermudah untuk mendeteksi hasil pendeteksian berupa objek atau tidak.

Metode *Cannyedge detection* merupakan pengembangan dari metode dasar *edge detection*. Perancangan sebuah prosedur dengan menerapkan langkah-langkah metode *Canny edge detection* akan menghasilkan sebuah tampilan gambar yang berbeda dengan menampilkan efek relief didalamnya. *Efek relief* adalah seperti sebuah tampilan batu kasar yang diukir, yaitu garis-garis kasar yang membentuk sebuah penggambaran objek didalamnya. *Efek relief* terbentuk dari bayangan terang dan gelap. Kedua bayangan ini terjadi akibat adanya sorotan sinar mengenai gambar dari arah tertentu. Kelebihan dari metode *Canny edge detection* ini kemampuan untuk mengurangi *noise* sebelum melakukan perhitungan deteksi tepi sehingga tepi-tepi yang dihasilkan lebih banyak. Algoritma *Canny edge detection* secara umum beroperasi sebagai berikut [12] :

1. Menghilangkan *noise* yang ada pada citra dengan mengimplementasikan Filter Gaussian Hasilnya citra akan tampak sedikit buram. Hal ini dimaksudkan untuk mendapatkan tepian citra yang sebenarnya. Bila tidak dilakukan maka garis-garis halus juga akan dideteksi sebagai tepian. Berikut ini salah satu contoh filter Gaussian dengan σ = 1.4 :

$$\frac{1}{115}\begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} \quad \text{..... (2.9)}$$

2. Melakukan deteksi tepi dengan salah satu operator deteksi tepi seperti Roberts, Perwit atau Sobel dengan melakukan pencarian secara horizontal

(G_x) dan secara vertikal (G_y). Berikut ini salah satu contoh operator deteksi tepi (operator sobel) :

$$\begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \quad \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \text{..... (2.10)}$$

Gx *Gy*

Hasil dari kedua operator digabungkan untuk mendapatkan hasil gabungan tepi vertikal dan horizontal dengan rumus:

$$|G| = |Gx| + |Gy| \text{ .....(2.11)}$$

3. Mendeteksi Arah tepian yang ditemukan dengan menggunakan rumus :

$$\theta = \arctan\left(\frac{Gx}{Gy}\right) \text{ ..... (2.12)}$$

4. *Non-maximal suppression* adalah citra yang berisi kandidat *edge* serta intensitas dari kekuatan edge di posisi piksel tersebut. Langkah terakhir adalah *thresholding* atau klasifikasi tiap piksel apakah termasuk dalam kategori piksel *edge* atau tidak. Pada tahap ini bisa saja menggunakan *threshold* yang berdasarkan pada satu nilai tertentu. Namun pemilihan *threshold* yang hanya menggunakan satu nilai ini memiliki keterbatasan yaitu adanya kemungkinan piksel yang hilang padahal sebetulnya merupakan piksel *edge* (*false-negative*) atau dimasukkannya piksel yang sebetulnya merupakan *noise* sebagai piksel *edge* (*false-positive*). Oleh sebab itu dalam melakukan klasifikasi tidak hanya diperlukan intensitas dari kekuatan edge sebagai pertimbangan namun juga topologi (keterhubungan antar-piksel) lokal dari piksel tersebut. Sederhananya *hysteresis thresholding* adalah klasifikasi dengan dua buah nilai *High-Threshold* dan *Low-Threshold*. Suatu piksel disahkan sebagai

piksel *edge* jika nilainya lebih besar atau sama dengan *High-Threshold* (*thresholding* umum) atau jika piksel tersebut memiliki intensitas kekuatan edge yang lebih besar dari *Low-Threshold* dan terhubung dengan piksel yang nilainya lebih besar dari *High-Threshold*.

2.8 Elemen-elemen Citra Digital

Citra digital mengandung sejumlah elemen-elemen dasar, diantaranya adalah sebagai berikut.

2.8.1 Kecerahan (*brightness*)

Kecerahan adalah kata lain untuk intensitas cahaya. Kecerahan pada sebuah titik (piksel) di dalam citra bukanlah intensitas yang sebenarnya, tetapi sebenarnya adalah intensitas rata-rata dari suatu area yang melingkupinya. Sistem visual manusia mampu menyesuaikan dirinya dengan tingkat kecerahan *(brightness level)* mulai dari yang paling rendah sampai yang paling tinggi.

2.8.2 Kontras (*contrast*)

Kontras menyatakan sebaran terang *(lightness)* dan gelap *(darkness)* didalam sebuah gambar. Citra dengan kontras rendah dicirikan oleh sebagian besar komposisi citranya adalah terang atau sebagian besar gelap. Pada citra dengan kontras yang baik, komposisi gelap dan terang tersebar secara merata.

2.8.3 Kontur (*contour*)

Kontur adalah keadaan yang ditimbulkan oleh perubahan intensitas padapiksel-pikselyang bertetangga. Karena adanya perubahan intensitas inilah mata kita mampu mendeteksi tepi-tepi *(edge)* objek di dalam citra.

2.8.4 Warna (*color*)

Warna adalah persepsi yang dirasakan oleh sistem visual manusia terhadappanjang gelombang cahaya yang dipantulkan oleh objek. Setiap warna mempunyai panjang gelombang (λ) yang berbeda. Warna merah mempunyai panjang gelombang paling tinggi, sedangkan warna ungu (*violet*) mempunyai panjang gelombang paling rendah.Warna-warna yang diterima oleh mata (sistem visual manusia) merupakan hasil kombinasi cahaya dengan panjang gelombang berbeda.

Penelitian memperlihatkan bahwa kombinasi warna yang memberikan rentang warna yang paling lebar adalah *red (R), green* (G), dan *blue (B).* Persepsi sistem visual manusia terhadap warna sangat relatif sebab dipengaruhi oleh banyak kriteria, salah satunya disebabkan oleh adaptasi yang menimbulkan distorsi. Misalnya bercak abu-abu di sekitar warna hijau akan tampak keungu-unguan (distorsi terhadap ruang), atau jika mata melihat warna hijau lalu langsung dengan cepat melihat warna abu-abu, maka mata menangkap kesan warna abu-abu tersebut sebagai warna ungu (distorsi terhadap waktu).

2.8.5 Bentuk (*shape*)

Shape adalah properti intrinsik dari objek tiga dimensi dengan pengertianbahwa *shape* merupakan properti intrinsik utama untuk sistem visual manusia. Manusia lebih sering mengasosiasikan objek dengan bentuknya ketimbang elemen lainnya (warna misalnya). Pada umumnya, citra yang dibentuk oleh mata merupakan citra dwimatra (2 dimensi), sedangkan objek yang dilihat umumnya berbentuk trimatra (3 dimensi). Informasi bentuk objek dapat diekstraksi dari citra pada permulaaan prapengolahan dan segmentasi citra. Salah satu tantangan utama pada *computer vision* adalah merepresentasikan bentuk, atau aspek-aspek penting dari bentuk.

2.8.6 Tekstur (*texture*)

Tekstur dicirikan sebagai distribusi spasial dari derajat keabuan didalam sekumpulan piksel-pikselyang bertetangga. Jadi, tekstur tidak dapat didefinisikan untuk sebuah piksel. Sistem visual manusia pada hakikatnya tidak menerima informasi citra secara *independen* pada setiap piksel, melainkan suatu citra dianggap sebagai suatu kesatuan. Resolusi citra yang diamati *ditetapkan* oleh skala pada mana tekstur tersebut dipersepsi.

Sebagai contoh, jika kita mengamati citra lantai berubin dari jarak jauh, maka kita mengamati bahwa tekstur terbentuk oleh penempatan ubin-ubin secara keseluruhan, bukan dari persepsi pola di dalam ubin itu sendiri. Tetapi, jika kita mengamati citra yang sama dari jarak yang dekat, maka hanya beberapa ubin yang tampak dalam bidang pengamatan, sehingga kita mempersepsi bahwa tekstur terbentuk oleh penempatan pola-pola rinci yang menyusun tiap ubin.

2.9 Video Digital

Video digital pada dasarnya tersusun atas serangkaian *frame*. Rangkaian*frame* tersebut ditampilkan pada layar dengan kecepatan tertentu, tergantung pada *frame rate* yang diberikan (dalam *frame/second*). Jika *frame rate* cukup tinggi, mata manusia tidak dapat menangkap gambar atau *frame*, melainkan menangkapnya sebagai rangkaian yang kontinyu (*video*). Masing-masing *frame* merupakan citra digital. Suatu citra digital direpresentasikan dengan sebuah matriks yang masing-masing elemennya merepresentasikan nilai intensitas.

Jika I adalah matriks dua dimensi, I (x,y) adalah nilai intensitas yang sesuai pada posisi baris x dan kolom y pada matriks tersebut. Titik-titik ditempatkan *image* di *sampling* disebut *picture elements*, atau sering dikenal sebagai piksel. Karakteristrik suatu video digital *ditetapkan* oleh resolusi *(resolution)* atau *frame dimention*, piksel *depth*, dan *frame rate*.

2.9.1 Resolusi

Resolusi *(resolution)* atau *frame dimention* adalah ukuran sebuah *frame*.Resolusi dinyatakan dalam piksel *x* piksel. Semakin tinggi resolusi, semakin baik kualitas video yang dihasilkan, dalam artian bahwa ukuran fisiknya sama, video dengan resolusi tinggi akan lebih detail. Namun resolusi yang tinggi akanmengakibatkan jumlah *bit* yang diperlukan untuk menyimpan atau mentransmisikannya meningkat.

2.9.2 *Bit Depth*

Bit Depth mendeteksi jumlah *bit* yang digunakan untuk merepresentasikan tiap piksel pada sebuah *frame*. Kedalam *bit* dinyatakan dalam *bit/*piksel. Semakin banyak jumlah *bit* yang digunakan untuk merepresentasikan sebuah piksel, yang berarti semakin tinggi kedalaman piksel-nya, maka semakin baik pula kualitasnya, dengan konsekuensi jumlah *bit* yang diperlukan menjadi lebih tinggi. Dengan 1 *byte* (8*bit*) untuk tiap piksel, diperoleh 256 *level* intensitas.

Dengan *level* intensitas sebanyak itu, umumnya mata manusia sudah dapat dipuaskan. Kedalaman piksel paling rendah terdapat pada *binary value image* yang hanya menggunakan 1 *bit/*piksel, sehingga hanya ada dua kemungkinan bagi tiap piksel, yaitu 0 untuk hitam dan 1 untuk putih. Pada kenyataannya semakin sedikit jumlah *bit* yang digunakan untuk tiap piksel, maka kualitas gambar akan semakin turun.

2.9.3 *Frame Rate*

Frame Rate menunjukkan jumlah *frame* yang digambarkan tiap detik dandinyatakan dengan *frame/second*. Sehubungan dengan *frame rate* ini, ada dua hal yang diperlukan diperhatikan yaitu kehalusan gerakan *(smooth mation)* dan kilatan *(flash)*. Kehalusan gerakan *ditetapkan* oleh jumlah *frame* yang berbeda per detik. Untuk mendapatkan gerakan yang halus, video setidaknya harus menampilkan sedikitnya 25 *frame/second*. Kilatan *ditetapkan* oleh jumlah berapa kali layar digambar perdetik (*frame rate*), dengan 29 *frame/second* kilatan sudah dapat dilenyapkan. Video yang berkualitas baik akan memiliki *frame rate* yang

tinggi, setidaknya sesuai dengan mata manusia, yang berarti membutuhkan jumlah *bit* yang lebih tinggi.

2.10 *Cross Correlation* (Korelasi Silang)

Korelasi silang adalah metode standar untuk memperkirakan sejauh mana dua seri saling berkorelasi. Misalnya dengan mempertimbangkan dua seri x (i) dan y (i), dimana i = 0,1,2 ... N-1. Korelasi silang didefinisikan sebagai berikut [13].

$$r = \frac{\sum_i [(x(i)-mx)*(y(i-d)-my)]}{\sqrt{\sum_i (x(i)-mx)^2}\sqrt{\sum_i (y(i-d)-my)^2}} \quad \text{..... (2.13)}$$

Dimana, mx dan my merupakan hubungan seri yang sesuai dan d merupakan delay (lama penundaaan).

2.11 Logika *Fuzzy*

Logika *Fuzzy* pertama kali dicetuskan oleh Lotfi Zadeh dari Universitas Berkeley, California pada tahun 1965. Logika Fuzzy adalah "Suatu sistem yang digunakan untuk menangani konsep kebenaran parsial yaitu kebenaran yang berada diantara sepenuhnya benar dan sepenuhnya salah" (Yan, Jun. 1994;14). Logika Fuzzy berbeda dengan logika digital biasa, dimana logika digital biasanya hanya mengenal dua keadaan yaitu "ya"-"tidak" atau "on"-"off" atau "*high"*-"*low*" atau "1"-"0". Logika *Fuzzy* menggantikan logika *Boolean* dalam tingkat kebenaran..Sistem logika yang merupakan perluasan dari logika *multivalued*.Logika *Fuzzy* banyak digunakan dalam 10 pengenalan pola (*pattern recognition*), pengolahan citra, pengendalian (*control)*, hingga pengambilan keputusan (*decision making*).

2.11.1 Himpunan *Fuzzy*

Himpunan *Fuzzy* adalah suatu kumpulan atau koleksi objek-objek yang mempunyai kesamaan sifat tertentu. Himpunan *Fuzzy* merupakan suatu pengembangan lebih lanjut tentang konsep himpunan dalam matematika. Arti lain himpunan *Fuzzy* adalah rentang nilai-nilai, masing-masing nilai mempunyai derajat keanggotaan antara 0 sampai dengan 1. Himpunan *Fuzzy* memiliki 2 atribut, yaitu:

a. Linguistik, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti sepi, sedang, dan padat.
b. Numerik, suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel seperti 5, 10, dan 15.

Beberapa hal yang perlu diketahui dalam memahami *Fuzzy*, yaitu:

a. Variable *Fuzzy*, merupakan variabel yang hendak dibahas dalam suatu sistem *Fuzzy*. Contoh: umur, *temperature*, permintaan, dan lain sebagainya.
b. Semesta pembicaraan, merupakan keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variabel *Fuzzy*. Semesta pembicaraan merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai semesta pembicaraan dapat berupa bilangan positif maupun negatif.
c. Domain himpunan *Fuzzy*, merupakan keseluruhan nilai yang diijinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *Fuzzy*. Seperti halnya semesta pembicaraan, domain merupakan himpunan

bilangan *real* yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai domain dapat berupa bilangan positif dan bilangan negatif.

2.11.2 Fungsi Keanggotaan *Fuzzy* (*Membership Function*)

Fungsi keanggotaan adalah suatu kurva yang menunjukkan pemetaan titik-titik input data kedalam nilai keanggotaannya (sering juga disebut dengan derajat keanggotaan) yang memiliki interval antara 0 sampai 1. Fungsi keanggotaan *Fuzzy* berguna untuk mendapatkan nilai keanggotaan dan digunakan sebagai perhitungan input output sistem. Ada 2 (dua) fungsi keanggotaan *Fuzzy* yang digunakan dalam tugas akhir ini, yaitu :

a. Representasi Linier Kurva Segitiga

Kurva segitiga pada dasarnya merupakan gabungan antara 2 garis (linier) seperti terlihat pada **Gambar 2.3.**

Fungsi keanggotaan dari representasi kurva segitiga adalah :

$$\mu[x] = \begin{cases} 0 & ; a \leq x \; atau \; x \geq c \\ (x-a)/(b-a); a \leq x \leq b \\ (c-x)/(c-b); b \leq x \leq c \end{cases} \quad \text{..... (2.14)}$$

Atau

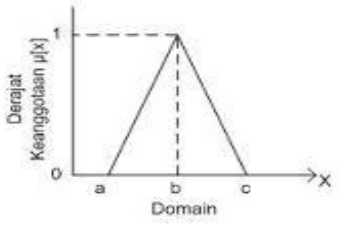
$$\mu[x] = \max(min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), o) \quad \text{..... (2.15)}$$

a = nilai domain terkecil yang mempunyai derajat keanggotaan nol

b = nilai domain yang mempunyai derajat keanggotaan satu

c = nilai domain terbesar yang mempunyai derajat keanggotaan nol

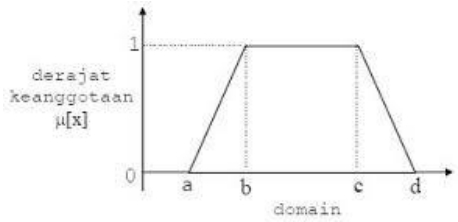
x = nilai input yang akan diubah ke dalam bilangan *Fuzzy*



Gambar 2.3 Representasi Kurva Segitiga [7]

b. Representasi Kurva Trapesium

Pada dasarnya kurva trapesium hampir sama dengan kurva segitiga, akan tetapi ketika nilai domain telah mencapai derajat keanggotaan tertinggi terjadi nilai domain dengan nilai sama untuk beberapa saat kemudian nilai domain turun kembali menuju derajat keanggotaan terendah, seperti terlihat pada **Gambar 2.4.**



Gambar 2.4 Representasi Kurva Trapesiumm [7]

Fungsi keanggotaan dari representasi kurva trapezium adalah :

$$\mu[x,a,b,c,d] = \begin{cases} 0 & ;x \le a \; atau \; x \ge d \\ (x-a)/(b-a); & a \le x \le b \\ 1 & ;b \le x \le c \\ (d-x)/(d-c); & c \le x \le d \\ 0 & ;x \ge d \end{cases} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (2.16)$$

Atau

$$\mu[x] = \max(min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), o) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\

x = nilai input yang akan diubah ke dalam bilangan *Fuzzy*

2.11.3 Operasi Logika *Fuzzy*

Operasi logika adalah operasi yang mengkombinasikan dan memodifikasi 2 atau lebih himpunan *Fuzzy*. Nilai keanggotaan baru hasil operasi dua himpunan disebut *firing strength* atau αpredikat, ada 3 operasi dasar yang diciptakan oleh Zadeh:

- Operator AND, berhubungan dengan operasi *intersection* pada himpunan, α predikat diperoleh dengan mengambil nilai minimum antar kedua himpunan.

$$\mu A \cap \mu B = \min(\mu A[x], \mu B[y]) \quad \text{..... (2.18)}$$

- Operator OR, berhubungan dengan operasi union pada himpunan, α predikat diperoleh dengan mengambil nilai maximum antar kedua himpunan.

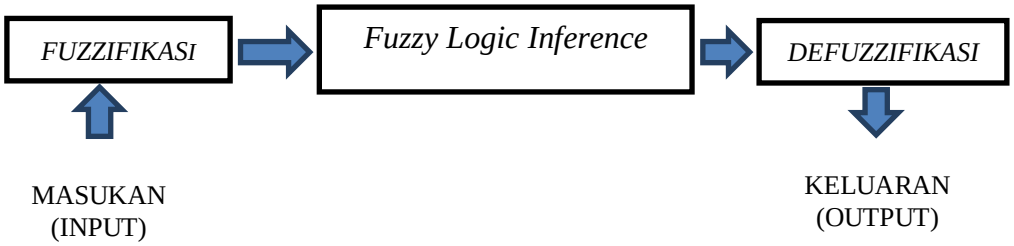
$$\mu A \cup \mu B = \max(\mu A[x], \mu B[y]) \quad \text{..... (2.19)}$$

- Operasi NOT, berhubungan dengan operasi komplemen pada himpunan, αpredikat diperoleh dengan mengurangkan nilai keanggotaan elemen pada himpunan dari 1.

$$\mu A' = 1 - \mu A[x] \quad \text{..... (2.20)}$$

2.11.4 Tahapan Membangun Sistem *Fuzzy*

Tahapan membangun sistem *Fuzzy* tergantung metoda yang digunakan, karena banyak metoda untuk membangun sistem *Fuzzy*. Namun secara garis besar dapat disimpulkan sebagai berikut :



Gambar 2.5 Proses Sistem *Fuzzy* [11]

a. *Fuzzification*

Proses *Fuzzification* berfungsi untuk mengubah nilai suatu masukan menjadi suatu fungsi keanggotaan *Fuzzy*. Pada proses ini *Membership Function* sudah *ditetapkan*.

b. *Fuzzy Logic Inference*

Proses ini berfungsi untuk mengaplikasikan aturan *(Fuzzy Rule)* pada masukan *Fuzzy* yang dihasilkan dalam proses *Fuzzification* dan mengevaluasi tiap aturan dengan masukan yang dihasilkan dari proses *Fuzzyfikasi* dengan mengevaluasi hubungan atau derajat keanggotaannya.

c. *Defuzzification*

Proses *Defuzzification* merupakan pengubahan kembali data-data *Fuzzy* kembali kebentuk *numeric* yang dapat dikirimkan ke peralatan pengendalian. Proses *Defuzzification* dapat dilakukan dengan beberapa cara, diantaranya [11:

- *Centroid Of Area,*

$$z_{COA} = \frac{\int z\, \mu A(z) z\, dz}{\int z\, \mu A(z)\, dz} \quad \text{..... (2.21)}$$

- *Bisector Of Area,*

$$\int_{\alpha}^{z\ BOA} \mu A(z) dz = \int_{z\ BOA}^{\beta} \mu A(z) dz \quad \text{..... (2.22)}$$

- *Mean Of Maximum,*

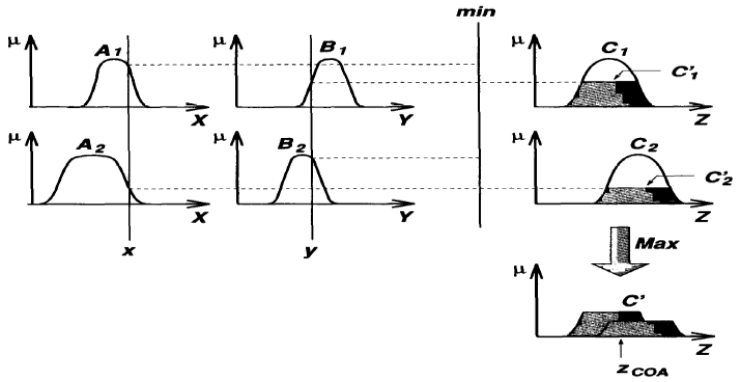
$$z_{MOM} = \frac{\int z' \, z \, dz}{\int z' \, dz} \quad \text{..... (2.23)}$$

2.11.5 *Fuzzy Inference System*

Sistem Inferensi *Fuzzy (Fuzzy Inference System/FIS)* disebut juga *Fuzzy Inference Engine* adalah sistem yang dapat melakukan penalaran dengan prinsip serupa seperti manusia melakukan penalaran dengan nalurinya. Ada banyak jenis FIS, tetapi yang paling terkenal adalah Mamdani dan Sugeno, karena paling mudah dimengerti daripada FIS yang lainya.

a. FIS Mamdani

FIS Mamdani adalah FIS yang paling mudah dimengerti, karena paling sesuai dengan naluri manusia. Metode ini diperkenalkan oleh Ebrahim Mamdani pada tahun 1975. FIS mamdani bekerja berdasarkan kaidah-kaidah linguistik dan memiliki algoritma *Fuzzy* yang menyediakan sebuah aproksimasi untuk dimasuki analisa matematik.

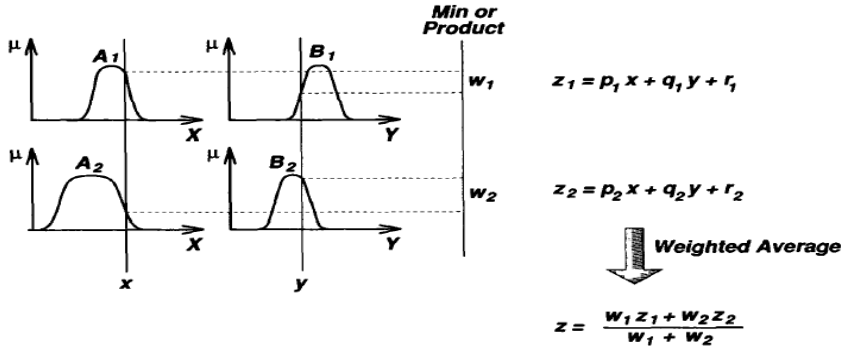


Gambar 2.6 FIS Mamdani [11]

b. FIS Sugeno

FIS Metode Sugeno hampir sama dengan FIS metode Mamdani, hanya saja *output* sistem tidak berupa himpunan *Fuzzy* melainkan berupa

konstanta atau persamaan linier. Metoda ini diperkenalkan oleh Takagi-Sugeno Kang pada tahun 1985.



Gambar 2.7 FIS Sugeno [11]

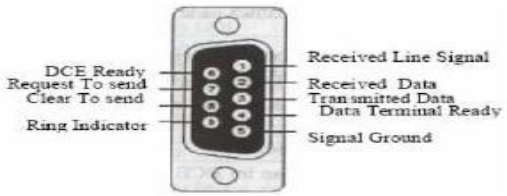
2.12 Komunikasi Serial

Komunikasi serial adalah komunikasi yang mengantarkan data digital secara bit per bit secara bergantian melalui media interface serial. Pengiriman data melalui interface serial dapat dilakukan secara bit per bit (setiap satu step waktu → bit) atau juga dalam satuan *baud* dimana 1 *baud* tidak mesti senilai dengan 1 bit per *second*, tergantung besar data untuk setiap *clock* transfer. Dalam tugas akhir ini penulis menggunakan komunikasi serial RS232.

Komunikasi Serial RS232

Standar RS232 ditetapkan oleh *Electronic Industry Association and Telecomunication Industry Association* pada tahun 1962. Nama lengkapnya adalah *EIA/TIA-232 Interface Between Data Terminal Equipment and Data Circuit-Terminating Equipment Employing Serial Binary Data Interchange*. Meskipun namanya cukup panjang tetapi standar ini hanya menyangkut komunikasi data antara komputer dengan alat-alat pelengkap computer.

Standard RS232 mendeteksi jenis-jenis sinyal yang dipakai mengatur pertukaran informasi antara DTE dan DCE, semuanya terdapat 24 jenis sinyal tapi yang umum dipakai hanyalah 9 jenis sinyal. Konektor yang dipakai pun *ditetapkan* dalam standard RS232, untuk sinyal yang lengkap dipakai konektor DB25, sedangkan konektor DB9 hanya bisa dipakai untuk 9 sinyal yang umum dipakai.



Gambar 2.8 Konektor DB9 [14]

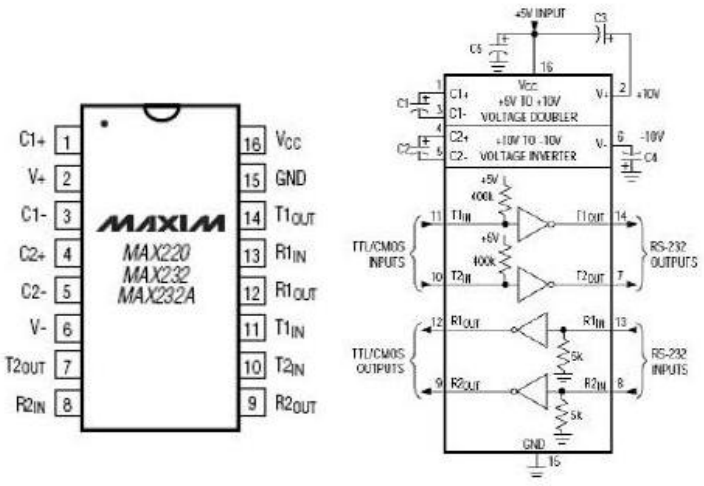
Tabel 2.1 Fungsi Pin Konektor DB9

Pin DB9	**Singkatan**	***Keterangan***	**Fungsi**
Pin 3	TD	*Transmit Data*	Untuk pengiriman data serial (TDX)
Pin 2	RD	*Receive Data*	Untuk penerimaan data serial (RDX)
Pin 7	RTS	*Request To Send*	Sinyal untuk menginformasikan modem bahwa UART siap melakukan pertukaran data
Pin 8	CTS	*Clear To Send*	Digunakan untuk memberitahukan bahwa modem siap untuk melakukan pertukaran data
Pin 6	DSR	*Data Set Ready*	Memberitahukan UART bahwa modem siap untuk melakukan pertukaran data
Pin 5	SG	*Signal Ground*	
Pin 1	CD	*Carrier Detect*	Saat modem mendeteksi suatu *carrier* dari modem lain maka sinyal ini akan diaktifkan
Pin 4	DTR	*Data Terminal Ready*	Kebalikan dari DSR untuk memberitahukan bahwa UART siap melakukan hubungan komunikasi
Pin 9	RI	*Ring Indikator*	Akan aktif jika modem mendeteksi adanya sinyal dering dari saluran telepon

Sinyal-sinyal tersebut ada yang menuju ke DCE ada pula yang berasal dari DCE. Bagi sinyal yang menuju ke DCE artinya DTE berfungsi sebagai output dan DCE berfungsi sebagai input, misalnya sinyal TD, pada sisi DTE kaki TD adalah output, dan kaki ini dihubungkan ke kaki TD pada DCE yang berfungsi sebagai input. Kebalikan sinyal TD adalah RD, sinyal ini berasal dari DCE dan dihubungkan ke kaki RD pada DTE yang berfungsi sebagai output.

Komunikasi serial RS232 memiliki level tegangan yang berkisar antara -25 volt sampai +25 volt. Oleh karena itu, dalam menghubungkan keduanya harus dilakukan konversi level tegangan dari level TTL menjadi RS232 dan begitu pula sebaliknya. Konverter level tegangan RS232 yang biasa digunakan adalah IC MAX232. IC MAX232 memiliki 16 buah pin dengan pin 16 sebagai VCC 5 volt dan pin 15 GND. Pin 8 dan 13 sebagai input receiver dari RS232, sedang pin 7 dan pin 14 sebagai *output* transmitter RS232

Berikut ini adalah konfigurasinya:



Gambar 2.9 Konfigurasi IC MAX232

2.13 Mikrokontroler

Mikrokontroler adalah sebuah *chip* yang berfungsi sebagai pengontrol rangkaian elektronik dan umunya dapat menyimpan program didalamnya. Mikrokontroler umumnya terdiri dari CPU *(Central Processing Unit),* memori, I/O tertentu dan unit pendukung seperti *Analog-to-Digital Converter* (ADC) yang sudah terintegrasi di dalamnya.



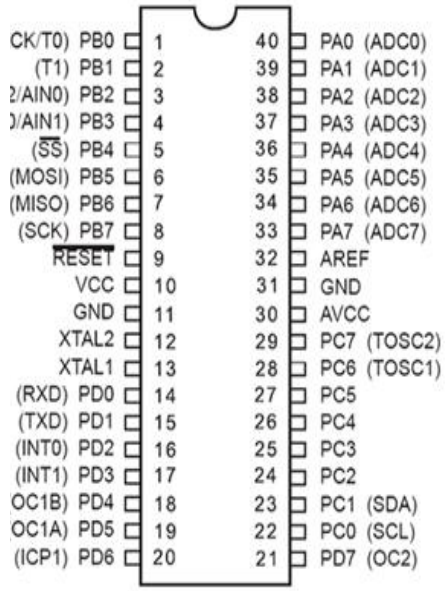
Gambar 2.10 ATmega8535

Atmega8535 merupakan mikrokontroller AVR yang memiliki arsitektur RISC 8 Bit, sehingga semua instruksi dikemas dalam kode 16-bit (16-bits word) dan sebagian besar instruksi dieksekusi dalam satu siklus instruksi clock. Dan ini sangat membedakan sekali dengan instruksi MCS-51 (Berarsitektur CISC) yang membutuhkan siklus 12 clock. RISC adalah *Reduced Instruction Set Computing* sedangkan CISC adalah *Complex Instruction Set Computing*. AVR dikelompokkan kedalam 4 kelas, yaitu ATtiny, keluarga AT90Sxx, keluarga ATMega, dan keluarga AT86RFxx. Dari kesemua kelas yang membedakan satu sama lain adalah ukuran *onboard memori, on-board peripheral* dan fungsinya. Dari segi arsitektur dan instruksi yang digunakan mereka bisa dikatakan hampir sama.

Fitur ATMega8535

- Sistem processor 8 bit berbasis *RISC* dengan kecepatan maksimal 16 MHz.
- Ukuran memory *flash* 8KB, *SRAM* sebesar 512 byte, *EEPROM* sebesar 512 byte.
- ADC internal dengan resolusi 10 bit sebanyak 8 channel
- Port komunikasi serial USART dengan kecepatan maksimal 2.5 Mbps
- Mode Sleep untuk penghematan penggunaan daya listrik

Konfigurasi Pin ATMega8535



Gambar 2.11 Konfigurasi Pin ATmega8535

- VCC merupakan Pin yang berfungsi sebagai pin masukan catudaya
- GND merupakan Pin *Ground*
- Port A (PA0...PA7) merupakan pin I/O dan pin masukan ADC
- Port B (PB0...PB7) merupakan pin I/O dan pin yang mempunyai fungsi khusus yaitu Timer/Counter, komparator Analog dan SPI

- Port C (PC0...PC7) merupakan port I/O dan pin yang mempunyai fungsi khusus, yaitu komparator analog dan Timer Oscillator
- Port D (PD0...PD1) merupakan port I/O dan pin fungsi khusus yaitu komparator analog dan interrupt eksternal serta komunikasi serial
- RESET merupakan pin yang digunakan untuk mereset mikrokontroler
- XTAL1 dan XTAL2 merupakan pin masukan *clock* eksternal
- AVCC merupakan pin masukan untuk tegangan ADC
- AREF merupakan pin masukan tegangan referensi untuk ADC

Arsitektur ATMega8535

- Saluran IO sebanyak 32 buah, yaitu Port A, Port B, Port C dan Port D
- ADC 10 bit sebanyak 8 Channel
- Tiga buah timer / counter
- 32 register
- *Watchdog Timer* dengan oscilator internal
- SRAM sebanyak 512 byte
- Memori Flash sebesar 8 kb
- Sumber Interrupt internal dan eksternal
- Port SPI (*Serial Pheriperal Interface*)
- EEPROM on board sebanyak 512 byte
- Komparator analog
- Port USART (*Universal Shynchronous Ashynchronous Receiver Transmitter*)

2.14 LED

LED adalah singkatan dari *Light Emiting Dioda*, merupakan komponen yang dapat mengeluarkan emisi cahaya. LED merupaan produk temuan lain setelah diode. Strukturnya juga sama dengan diode, tetapi belakangan ditemukan bahwa electron yang menerjang sambungan P-N juga melepaskan energi berupa energi panas dan energy cahaya. LED dibuat agar lebih efesien jika mengeluarkan cahaya. Untuk mendapatkan emisi cahaya pada semikonduktor, doping yang dipakai adalah galium, arsenic, dan phosphorus. Jenis doping yang berbeda menghasilkan warna cahaya yang berbeda pula.



Gambar 2.12 LED