

## BAB II

### LANDASAN TEORI

#### 2.1. Struktur Tulang Belakang

Tulang belakang atau vertebra adalah tulang tak beraturan yang membentuk punggung yang mudah digerakkan. Seperti yang ditunjukkan pada Gambar 2.1. manusia memiliki 33 ruas tulang belakang, yang terdiri dari empat bagian, yaitu tulang leher (*cervical*), tulang punggung (*thoracic*), tulang pinggang (*lumbar*), dan ekor (*sacral*). Tiga bagian teratas tersusun dari dan 7 tulang leher, 12 tulang dada, dan 5 tulang pinggang. Sedangkan bagian ekor dibentuk dari tulang ekor (*coccyx*) yang disusun oleh 4 tulang terbawah dan 5 tulang di atasnya akan bergabung membentuk bagian *sacrum*.



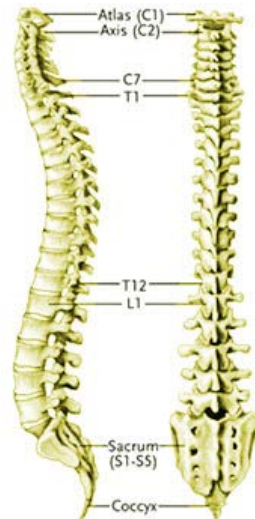
Gambar 2.1 Struktur tulang belakang manusia. (PT. Theramindo Jaya,2008)

### 2.1.1 Struktur Umum Tulang Belakang

Sebuah tulang belakang terdiri atas dua bagian yakni bagian anterior yang terdiri dari badan tulang atau corpus vertebrae, dan bagian posterior yang terdiri dari arcus vertebrae. Arcus vertebrae dibentuk oleh dua “kaki” atau pediculus dan dua lamina, serta didukung oleh penonjolan atau prosesus yakni prosesus articularis, prosesus transversus, dan prosesus spinosus. Prosesus tersebut membentuk lubang yang disebut foramen vertebrale. Ketika tulang belakang disusun, foramen ini akan membentuk saluran sebagai tempat sumsum tulang belakang atau medulla spinalis. Di antara dua ruas tulang belakang dapat ditemui celah yang disebut foramen intervertebrale.

Tulang leher, secara umum memiliki bentuk tulang yang kecil dengan spina atau prosesus spinosus (bagian seperti sayap pada belakang tulang) yang pendek, kecuali tulang ke-2 dan 7 yang prosesus spinosusnya pendek. Gambar 2.2 menunjukkan bahwa setiap ruas tulang diberi nomor sesuai dengan urutannya dari C1-C7 (C dari cervical), namun beberapa memiliki sebutan khusus seperti C1 atau atlas, C2 atau aksis.

Procesus spinosus pada tulang punggung akan berhubungan dengan tulang rusuk. Tulang punggung dapat membuat sedikit gerakan memutar. Bagian ini dikenal juga sebagai ‘tulang punggung dorsal’ dalam konteks manusia. Bagian ini diberi nomor T1 hingga T12.



Gambar 2.2 Struktur ruas-ruas pada tulang belakang. (dari kiri: tampak samping kiri; tampak belakang)

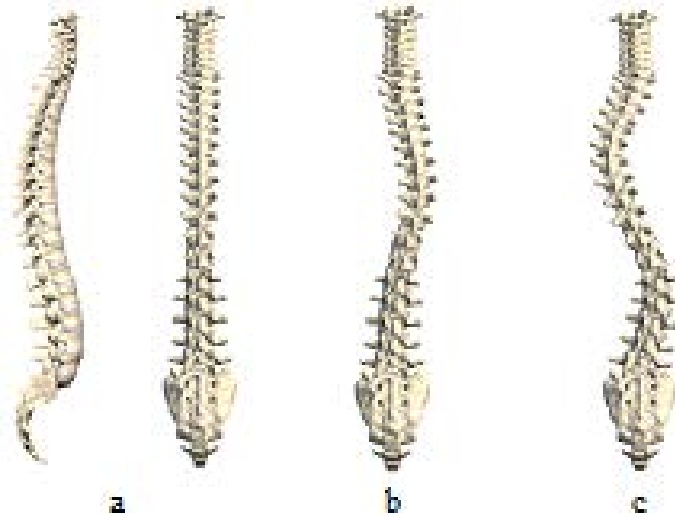
Tulang pinggang merupakan bagian paling tegap konstruksinya dan menanggung beban terberat dari yang lainnya. Bagian ini memungkinkan gerakan fleksi dan ekstensi tubuh, dan beberapa gerakan rotasi dengan derajat yang kecil. Bagian ini diberi nomor L1 hingga L5.

Pada tulang ekor bagian *sacrum*, terdapat 5 tulang (S1-S5). Tulang-tulang bergabung dan tidak memiliki celah atau diskus intervertebralis satu sama lainnya. Sedangkan bagian ekor (*coccygeal*), memiliki 3 hingga 5 tulang (Co1-Co5) yang juga saling bergabung dan tanpa celah.

## 2.2. Skoliosis

Skoliosis adalah sebuah fenomena kelainan tulang belakang di mana tulang belakang akan melengkung membentuk sudut yang tidak seharusnya. Misalnya melengkung sehingga membentuk seperti huruf “C” atau “S” seperti pada Gambar 2.3. Fenomena skoliosis ini pun sebenarnya tidak hanya dipandang dari satu sisi saja, sehingga seolah kelainan ini bersifat 2-dimensi, namun fenomena skoliosis

dapat terjadi dalam ruang lingkup 3-dimensi. Jadi tulang belakang, selain dapat melengkung dalam sumbu Y, juga dapat melengkung (terputar) dalam sumbu X, dan Z seperti pada Gambar 2.4.



Gambar 2.3 (a) Tulang belakang normal (tampak kiri; belakang);  
 (b) melengkung membentuk "C" (tampak belakang);  
 (c) melengkung membentuk "S" (tampak belakang);



Gambar 2.4 Tulang belakang yang melengkung sekaligus terputar. (tampak belakang)

Tingkat kelengkungan tulang belakang atau derajat skoliosis ditentukan oleh sudut kelengkungan skoliosis. Untuk mengetahui derajat skoliosis dapat dilakukan dengan menggunakan skoliometer atau yang lebih akurat dapat dilakukan dengan melakukan observasi terhadap gambar sinar-x tulang belakang. Sudut kelengkungan skoliosis dapat diukur dengan dua metode, yaitu metode *risser-ferguson angle* dan *cobb angle*. Seseorang akan dinyatakan memiliki skoliosis bila *cobb angle*-nya lebih dari  $10^{\circ}$ .

Pada dasarnya, skoliosis dapat dikategorikan menjadi skoliosis fungsional (non-struktural) dan skoliosis struktural. Skoliosis fungsional adalah fenomena skoliosis yang terjadi karena postur tubuh ketika duduk atau berdiri tidak tegak lurus, sehingga tulang bahu terlihat tidak sejajar. Namun sesungguhnya *cobb angle* pada skoliosis fungsional ini tidak signifikan, bahkan dapat dibilang struktur tulang belakangnya normal, dan bila memang ada skoliosis, sifatnya cenderung tidak progresif dan tidak berbahaya. Sedangkan skoliosis struktural adalah fenomena skoliosis yang memang disebabkan karena perkembangan kedua sisi tubuh yang tidak seimbang sehingga tulang belakang jadi melengkung secara permanen dan skoliosis ini bersifat progresif sehingga perlu penanganan. Cara paling mudah membedakan skoliosis fungsional dan skoliosis struktural adalah dengan membungkukkan badan ke depan dan melihat apakah skoliosis masih ada. Bila tidak ada, maka fenomena itu termasuk ke dalam skoliosis fungsional, namun bila ada, fenomena tersebut disebut dengan skoliosis struktural, atau secara singkat orang yang bersangkutan memang menderita skoliosis.

Skoliosis adalah kelainan yang meskipun bukanlah sebuah penyakit, namun bisa saja disebabkan karena penyakit. Penyebab skoliosis bermacam-macam,

namun yang paling sering terjadi adalah karena faktor kebiasaan sikap tubuh. Pada umumnya skoliosis mulai terjadi pada masa kanak-kanak, yang disebabkan sikap tubuhnya ketika duduk atau berdiri tidak tegak. Penyebab lain dari kasus skoliosis merupakan efek samping yang diakibatkan karena kecelakaan atau menderita kelainan tertentu, seperti distrofi otot, sindrom Marfan, atau pun sindrom Down. Berbagai kelainan tersebut menyebabkan otot atau saraf di sekitar tulang belakang tidak berfungsi sempurna dan menyebabkan bentuk tulang belakang menjadi melengkung.

Bila ditinjau dari waktu skoliosis mulai muncul, ada tiga tipe skoliosis, yaitu *infantile*, *juvenile*, dan *adolescent*. *Infantile* adalah kondisi ketika skoliosis muncul di antara waktu kelahiran sampai dengan usia 3 tahun. *Juvenile* adalah kondisi ketika skoliosis mulai muncul di antara usia 4 hingga 9 tahun. Bila skoliosis mulai muncul mulai umur 10 tahun hingga masa pertumbuhan selesai dimasukkan dalam kategori *adolescent*.

Dampak buruk skoliosis sangatlah besar bagi tubuh. Yang paling jelas terlihat adalah sikap tubuh yang tidak sejajar yang berdampak pada penampilan. Hal ini dapat menyebabkan masalah, seperti timbulnya rasa kurang percaya diri atau ketika melamar pekerjaan. Dampak lain yang sebenarnya lebih buruk adalah skoliosis dapat memicu penyakit yang lain karena tulang belakang yang melengkung dapat menyebabkan saraf tulang belakang terjepit, sehingga penderita akan menjadi lumpuh, bahkan meninggal. Selain itu, organ-organ lain, seperti paru-paru, hati, dan ginjal juga dapat saling terhimpit karena posisi tulang yang tidak benar.

Penanganan skoliosis ada beberapa macam, yang pertama adalah dengan melakukan observasi tingkat skoliosis tanpa melakukan apa-apa dahulu. Observasi skoliosis dilakukan bila *cobb angle* tidak lebih dari  $20^\circ$  karena, terutama pada anak-anak, banyak kasus *infantile* skoliosis yang sembuh dengan sendirinya seiring anak itu tumbuh bila sikap tubuhnya ketika duduk, berdiri, atau tidur baik dan benar. Namun bila *cobb angle* sudah lebih dari  $20^\circ$ , tulang belakang yang melengkung secara tidak normal perlu diluruskan kembali sedikit demi sedikit melalui beberapa macam cara rehabilitasi, yaitu dengan olahraga, pemasangan penyangga eksternal, dan dengan pemasangan penyangga internal pada tulang belakang melalui operasi.

Rehabilitasi skoliosis yang optimal tergantung pada derajat kemiringan skoliosis. Bila sudut kemiringan skoliosis (*cobb angle*) kurang dari  $40^\circ$ , rehabilitasi yang perlu dilakukan cukup dengan olahraga ataupun pemasangan penyangga eksternal (*bracing*). Namun bila *cobb angle* lebih dari  $40^\circ$  atau meningkat dengan cepat, perlu dilakukan pemasangan penyangga internal pada tulang belakang.

### 2.3. Tophat Filter

Dalam matematika morfologi dan pengolahan citra digital, transformasi top-hat adalah sebuah operasi yang mengekstrak elemen kecil dan detail dari gambar yang diberikan. Ada dua jenis tophat filter :

1. White Tophat, didefinisikan sebagai perbedaan antara gambar input dan pembukaannya oleh beberapa elemen penataan.

2. Black Tophat, didefinisikan sebagai perbedaan antara *closing* dan gambar input.

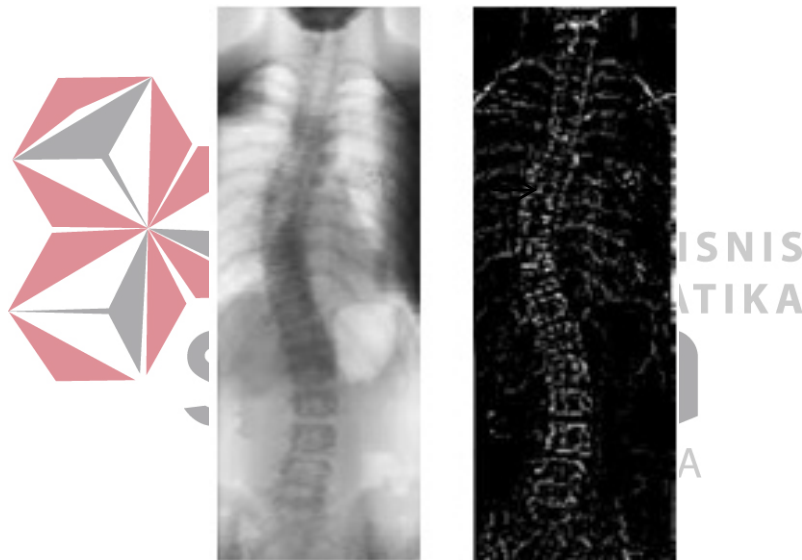
Tophat filter banyak digunakan untuk berbagai tugas pengolahan gambar, ekstraksi fitur, pemerataan latar belakang, peningkatan citra, dll.

Rumus White Tophat Filter :

$$T_w(f) = f - f \circ b \dots\dots\dots(1)$$

Rumus Black Tophat Filter :

$$T_b(f) = f \bullet b - f \dots\dots\dots(2)$$



Gambar 2.5 Contoh hasil *filtering* menggunakan black tophat filter.

#### 2.4. Model Snake Parametrik ( Tradisional Snake )

Snakes atau *active contour* adalah garis-garis lengkung yang didefinisikan pada citra yang dapat bergerak di bawah pengaruh gaya internal yang berasal dari garis lengkung itu sendiri dan gaya eksternal yang dikomputasikan dari data citra. Gaya internal dan eksternal didefinisikan sehingga snake dapat menyesuaikan diri dengan tepi objek atau fitur lain dari citra yang diinginkan. Snakes digunakan



secara luas dalam berbagai aplikasi, termasuk deteksi tepi, pemodelan bentuk, segmentasi, dan pendeteksi gerakan.



Gambar 2.6 Tepi batas yang memiliki celah (Xu dan Prince, 1998)



Gambar 2.7 Hasil konvergensi menggunakan snake tradisional (Li dan Acton, 2007)

Ada dua tipe umum dari model *active contour*, yaitu *parametric active contours* dan *geometric active contours*. GVF termasuk ke dalam *parametric active contours*. *Parametric active contour* mensintesis garis-garis lengkung parametric dalam sebuah citra dan membuat mereka bergerak pada fitur yang diinginkan. Fitur yang paling umum digunakan adalah tepi. Secara khusus, garis-garis lengkung itu akan diarahkan pada tepi-tepi oleh gaya potensial yang didefinisikan sebagai gradien negatif dari sebuah fungsi potensial. Gaya tambahan lain seperti gaya tekan dan gaya potensial termasuk pada gaya eksternal. Selain itu, ada juga gaya internal yang dirancang untuk menahan garis-garis lengkung itu agar tidak

lepas (gaya elastis) dan untuk menjaganya agar tidak terlalu melengkung (gaya lengkung).

Ada dua permasalahan pada algoritma parametric *active contour*. Masalah pertama adalah garis kontur inisial harus dekat dengan batas yang diinginkan atau hasilnya akan menjadi keliru. Masalah kedua adalah *active contour* masih mengalami kesulitan untuk menyesuaikan diri pada bentuk-bentuk yang memiliki batas-batas bercelah (cekungan) seperti pada Gambar 2.6. Hasil konvergensi yang didapatkan pada Gambar 2.6 dengan menggunakan snake tradisional dapat dilihat pada Gambar 2.7. (Xu dan Prince, 1998)

Snake tradisional adalah sebuah garis lengkung yang didefinisikan sebagai

$$x(s) = [X(s), Y(s)], \text{ di mana } s \in [0,1] \dots \dots \dots (3)$$

Yang bergerak melalui wilayah spasial dari sebuah citra untuk meminimalisasi fungsional energi yang didefinisikan sebagai:

$$E = \int_0^1 \frac{1}{2} [\alpha |x'(s)|^2 + \beta |x''(s)|^2] + E_{ext}(x(s)) ds \dots \dots \dots (4)$$

Di mana  $\alpha$  dan  $\beta$  adalah parameter bobot yang mengendalikan tingkat ketegangan dan kekakuan snake. Sedangkan  $x'(s)$  dan  $x''(s)$  mendefinisikan turunan pertama dan kedua dari  $x(s)$ . Fungsi energi eksternal  $E_{ext}$  diturunkan dari citra yang diambil dari nilai yang lebih kecil daripada fitur yang diinginkan atau tepi. Dengan mempertimbangkan citra *grayscale*  $I(x,y)$  yang dianggap sebagai fungsi dari sebuah variabel posisi yang kontinyu  $(x,y)$ , energi eksternal yang serupa didesain supaya mengarahkan *active contour* untuk mendekati tepi citra. Energi eksternal tersebut dinyatakan sebagai:

$$E_{ext}^{(1)}(x, y) = -|\nabla I(x, y)|^2 \dots \dots \dots (5)$$

$$E_{ext}^{(2)}(x, y) = -|\nabla [G_\sigma(x, y) * I(x, y)]|^2 \dots \dots \dots (6)$$

Di mana  $G_\sigma(x,y)$  adalah fungsi Gaussian dua dimensi dengan standar deviasi  $\sigma$  dan  $\nabla$  adalah operator gradien. Bila gambar yang ada di citra adalah gambar garis hitam di latar putih, maka energi eksternal yang sesuai harus mengikut-sertakan:

$$E_{ext}^{(3)}(x, y) = I(x, y) \dots\dots\dots (7)$$

$$E_{ext}^{(4)}(x, y) = G_\sigma(x, y) * I(x, y) \dots\dots\dots (8)$$

Dapat dilihat dengan jelas dari definisi tersebut bahwa semakin besar nilai  $\sigma$  akan menyebabkan batas tepi menjadi kabur. Namun kadang nilai  $\sigma$  yang besar diperlukan untuk meningkatkan jarak tangkap dari *active contour*.

Snake yang meminimalisasi  $E$  harus memenuhi persamaan Euler:

$$\alpha x''(s) - \beta x''''(s) - \nabla E_{ext} = 0 \dots\dots\dots (9)$$

Persamaan 9 dapat dianggap sebagai persamaan keseimbangan gaya:

$$F_{int} + F_{ext}^{(p)} = 0 \dots\dots\dots (10)$$

di mana:

$$F_{int} = \alpha x''(s) - \beta x''''(s) \dots\dots\dots (11)$$

dan

$$F_{ext}^{(p)} = -\nabla E_{ext} \dots\dots\dots (12)$$

Gaya internal  $F_{int}$  berguna untuk mengurangi peregangan dan pelengkungan garis-garis kontur snake, sedangkan gaya potensial eksternal  $F_{ext}^{(p)}$  menarik snake untuk menyesuaikan diri pada tepi-tepi citra yang diinginkan.

Untuk mencari solusi dari aturan Euler pada Persamaan 9, snake dibuat dinamis dengan menganggap  $x$  sebagai fungsi dari  $t$  sebagaimana  $s$  menjadi  $x(s,t)$ . Kemudian turunan parsial dari  $x$  yang berhubungan dengan  $t$  dibuat bernilai sama dengan sisi sebelah kiri Persamaan 9 sebagai berikut:

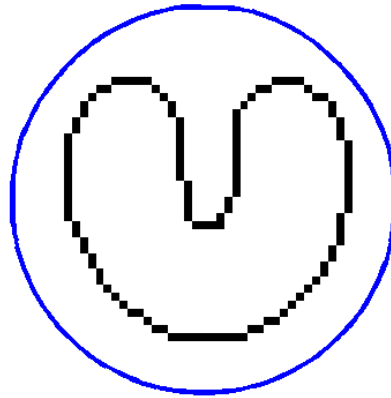
$$x_t(s, t) = \alpha x''(s, t) - \beta x''''(s, t) - \nabla E_{ext} \dots \dots \dots (13)$$

Ketika solusi  $x(s, t)$  sudah stabil,  $x_t(s, t)$  akan hilang dan akan didapatkan solusi dari Persamaan 9. Solusi numerik dari Persamaan 13 dapat diperoleh dengan mendiskritkan persamaan dan menyelesaikan sistem diskrit tersebut secara berulang-ulang. Kebanyakan implementasi snake menggunakan parameter yang mengalikan  $x_t$  untuk mengendalikan *step-size* sementara atau parameter untuk mengalikan  $\nabla E_{ext}$ , yang mengakibatkan kekuatan gaya eksternal dapat mengendalikannya. Untuk membuat agar magnitudo maksimumnya setara dengan satu, gaya eksternal perlu dinormalisasi. (Kaas dkk, 1987)

## 2.5. *Gradient Vector Flow* (GVF)

Untuk mengatasi 2 kekurangan yang dimiliki oleh Snake Tradisional maka dikembangkanlah sebuah metode baru yang menggunakan gaya eksternal yang disebut dengan medan gradient vector flow (GVF) yang merupakan bidang vektor yang rapat yang diturunkan dari citra dengan meminimalisasi fungsional energi tertentu pada kerangka yang bervariasi. Proses minimalisasi didapatkan dengan menyelesaikan pasangan persamaan diferensial parsial linier secara terpisah yang menyebarkan vektor-vektor gradien tepi citra biner atau *grayscale* yang dihitung dari citra. *Active contour* yang menggunakan medan GVF sebagai gaya eksternal tersebut disebut dengan GVF snake. GVF snake berbeda dengan hampir seluruh formulasi snake sebelumnya dalam hal gaya eksternalnya tidak dapat didefinisikan sebagai gradien negatif dari fungsi potensial. Karena itu, GVF snake tidak dapat diformulasikan menggunakan kerangka minimalisasi energi yang

standar. Untuk itu, hal tersebut harus dispesifikkan secara langsung dari kondisi keseimbangan gaya.



Gambar 2.8 Inisialisasi garis kontur GVF Snake (Li dan Acton, 2007)

Kelebihan utama dari GVF snake dibandingkan dengan snake tradisional adalah tidak terlalu bergantung pada inisialisasi dan dapat bergerak pada tepi yang berbentuk cekungan. Inisialisasi GVF snake dapat dilakukan di dalam, di luar, atau bahkan melewati tepi objek seperti pada Gambar 2.8. GVF snake tidak perlu diberi pengaturan apakah harus mengerut atau mengembang untuk menyesuaikan diri pada tepi objek. GVF snake juga memiliki jarak tangkap yang besar yang berarti bahwa selama tidak ada objek lain yang menginterferensi, garis kontur inisialisasi dapat diletakkan sangat jauh dari tepi objek. Peningkatan jarak tangkap ini didapatkan melalui penyebaran proses yang tidak mengaburkan tepi-tepinya sendiri. (Xu dan Prince, 1998)

Sebuah gaya eksternal tipe baru untuk snake yang tidak berubah terhadap waktu ataupun bergantung pada posisi snake itu sendiri memiliki dasar matematis dari teorema Helmholtz yang menyatakan bahwa medan vektor statis yang paling

umum dapat didekomposisi menjadi dua komponen, yaitu komponen yang *irrotational* (bebas lingkaran) dan komponen solenoidal (bebas perbedaan). Gaya potensial eksternal yang dihasilkan dari variasi formulasi snake tradisional harus dimasukkan dulu ke dalam persamaan keseimbangan gaya sebagai sebuah medan *irrotational* yang statis karena ini adalah fungsi potensial gradien. Oleh karena itu, medan statis yang lebih umum  $F_{ext}^{(g)}$  dapat diperoleh dengan mempertimbangkan kemungkinan yang meliputi komponen *irrotational* dan komponen solenoidal. Medan gaya eksternal didesain supaya memiliki properti yang diinginkan, baik dari jarak tangkap yang besar maupun adanya gaya yang mengarah pada celah tepi objek. Hasil formulasi tersebut menghasilkan medan gaya eksternal yang dapat diperkirakan memiliki komponen *irrotational* maupun solenoidal. Pendekatan yang dilakukan adalah dengan menggunakan kondisi keseimbangan gaya sebagai poin awal untuk mendesain snake. Medan gaya eksternal statis didefinisikan sebagai

$$F_{ext}^{(g)} = v(x, y) \dots \dots \dots (14)$$

Untuk mendapatkan persamaan snake dinamis yang berkaitan gaya potensial  $-\nabla E_{ext}$  pada persamaan 10 harus digantikan dengan  $v(x, y)$  menurut:

$$x_t(s, t) = \alpha x''(s, t) - \beta x''''(s, t) + v \dots \dots \dots (15)$$

Lengkungan parametrik yang menyelesaikan persamaan dinamis tersebut disebut dengan GVF snake. Persamaan tersebut dapat diselesaikan secara numerik dengan membuatnya menjadi diskrit dan berulang-ulang dengan cara yang sama seperti pada snake tradisional.

Meskipun konfigurasi akhir GVF snake memenuhi persamaan keseimbangan gaya, persamaan ini tidak merepresentasikan persamaan Euler tentang masalah

minimalisasi energi secara umum pada persamaan 3. Hal ini disebabkan karena  $v(x, y)$  tidak akan menjadi medan irrotational. Namun kekurangan ini dikompensasi dengan adanya peningkatan performa yang signifikan dari GVF snake.

Langkah awal implementasi GVF snake dimulai dengan mendefinisikan *edge map*  $f(x, y)$  yang diturunkan dari citra  $I(x, y)$  yang memiliki properti yang semakin mendekati tepi citra nilainya semakin besar. *Edge map* yang digunakan dapat berupa citra biner maupun *grayscale*. Contohnya:

$$f(x, y) = -E_{ext}^{(i)}(x, y), \text{ di mana } i = 1, 2, 3, \text{ atau } 4 \dots\dots\dots (16)$$

Tiga properti umum dari *edge map* sangat penting dalam konteks ini. Pertama, gradien dari *edge map*  $\nabla f$  memiliki vektor-vektor yang menunjuk kepada tepi-tepi citra yang normal pada tepi. Kedua, vektor-vektor ini memiliki magnitudo yang besar hanya di dekat tepi citra. Ketiga, di area yang homogen di mana  $I(x, y)$  mendekati konstan, nilai  $\nabla f$  akan mendekati nol.

Bila gradien dari sebuah *edge map* sebagai sebuah gaya eksternal pada snake tradisional, properti pertamanya akan membuat snake yang diinisialisasi dekat dengan tepi akan menjadi konvergen pada konfigurasi stabil di dekat tepi. Properti kedua akan menyebabkan jarak tangkap menjadi sangat kecil. Sedangkan properti ketiga akan menyebabkan daerah-daerah yang homogen tidak memiliki gaya eksternal. Efek yang dihasilkan properti pertama adalah efek yang menguntungkan, sedangkan efek dari dua properti lainnya tidak menguntungkan. Untuk itu harus dibuat properti yang memiliki gradien tinggi di sekitar tepi namun dapat mengembangkan peta gradien lebih jauh dari tepi hingga ke daerah homogen menggunakan proses penyebaran yang terkomputasi. Hal ini juga akan

menyebabkan keuntungan lain, yaitu proses difusi akan menimbulkan vektor-vektor yang mengarah pada tepi-tepi yang bercelah.

Medan gradient vector flow didefinisikan sebagai medan vektor  $v(x, y) = [u(x, y), v(x, y)]$  yang meminimalisasi fungsi energi:

$$\varepsilon = \iint \mu(u_x^2 + u_y^2 + v_x^2 + v_y^2) + |\nabla f|^2 |v - \nabla f|^2 dx dy \dots\dots\dots (17)$$

Variasi formulasi ini mengikuti prinsip standar untuk membuat hasil yang halus ketika tidak ada data yang tersedia. Ketika  $|\nabla f|$  bernilai kecil, energi didominasi oleh penjumlahan kuadrat dari turunan parsial medan vektor mengikuti medan yang bervariasi pelan-pelan. Di sisi lain, ketika  $|\nabla f|$  bernilai besar, operan yang kedua akan mendominasi integral dan diminimalisasi dengan pengaturan ketika  $v = \nabla f$ . Hal ini membuat efek yang diinginkan dengan menjaga nilai  $v$  agar sama dengan gradien *edge map* ketika nilainya menjadi besar. Namun hal itu juga akan memaksa medan untuk bervariasi secara pelan di area yang homogen. Parameter  $\mu$  adalah parameter reguler yang mengatur timbal balik antara operan pertama dan kedua dalam integral. Parameter ini harus diatur sesuai dengan jumlah *noise* yang ada di dalam citra. Semakin banyak *noise*, maka nilai  $\mu$  harus semakin besar.

Dengan menggunakan variasi kalkulus medan GVF dapat ditemukan dengan menyelesaikan persamaan Euler berikut ini:

$$\mu \nabla^2 u - (u - f_x)(f_x^2 + f_y^2) = 0 \dots\dots\dots (18)$$

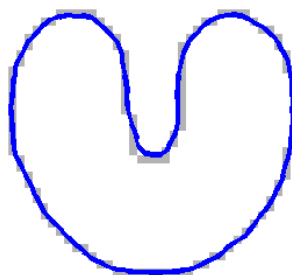
$$\mu \nabla^2 v - (v - f_y)(f_x^2 + f_y^2) = 0 \dots\dots\dots (19)$$

Di mana  $\nabla^2$  adalah operator Laplacian. Persamaan ini memberikan intuisi yang lebih jauh daripada formulasi GVF. Pada area homogen, di mana  $I(x, y)$  bernilai konstan, operan kedua di tiap persamaan bernilai nol karena gradien  $f(x, y)$  juga

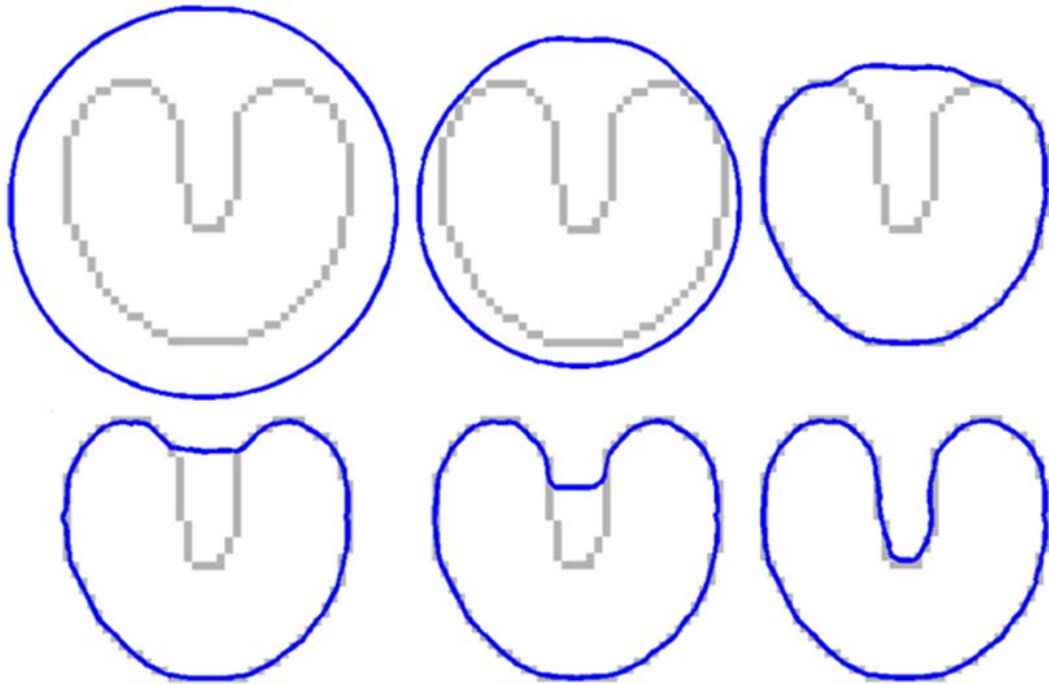


bernilai nol. Oleh karena itu, di dalam area tersebut masing-masing nilai  $u$  dan  $v$  ditentukan oleh persamaan Laplace dan GVF yang dihasilkan diinterpolasikan dari tepi area dengan merefleksikan semacam kompetisi di antara vektor-vektor tepi. Inilah sebabnya GVF menghasilkan vektor-vektor yang dapat menunjuk pada tepi-tepi yang bercelah. Sehingga hasil yang didapatkan oleh GVF snake adalah seperti pada Gambar 2.9. (Li dan Acton, 2007)

Permasalahan yang muncul dari GVF snake ini adalah kegagalan pada lengkungan dan *boundary leakage* seperti pada Gambar 2.8 yang disebabkan karena ketiadaan energi citra. Masalah kedua adalah masih terlalu sensitif pada proses inialisasi yang harus simetris. Yang ketiga adalah kegagalan bila ada celah yang terlalu sempit dan dalam karena gaya eksternal pada celah tersebut terlalu lemah. Permasalahan keempat adalah kemungkinan terjadinya *local minima* karena syarat energi global minimal tidak kunjung terpenuhi. Permasalahan terakhir adalah ketidak-mampuannya beradaptasi pada perubahan topologi yang terjadi pada citra untuk kasus-kasus tertentu. (Jalba dkk, 2004)



Gambar 2.9 Hasil konvergensi menggunakan GVF snake (Li dan Acton, 2007)



Gambar 2.10 Hasil konvergensi menggunakan GVF snake (Li dan Acton, 2007)

## 2.6 OpenCV

OpenCV (*Open Computer Vision*) adalah sebuah API (*Application Programming Interface*) library yang sudah sangat familiar pada pengolahan citra *computer vision*. *Computer vision* itu sendiri adalah salah satu cabang dari bidang ilmu pengolahan citra (*Image Processing*) yang memungkinkan komputer dapat melihat seperti manusia. Dengan *computer vision* tersebut komputer dapat mengambil keputusan, melakukan aksi, dan mengenali terhadap suatu objek. Beberapa pengimplementasian dari *computer vision* adalah *face recognition*, *face detection*, *face/object tracking*, *road tracking*, dll.

OpenCV adalah *library open source* untuk *computer vision* untuk C/C++, OpenCV didesain untuk aplikasi *real-time*, memiliki fungsi-fungsi akuisisi yang baik untuk *image/video*.

Fitur yang dimiliki OpenCV antara lain :

1. Manipulasi data citra (*allocation, copying, setting, convert*).
2. Citra dan video I/O (*file dan kamera based input, image/video file output*).
3. Manipulasi Matriks dan Vektor beserta rutin-rutin aljabar linear (*products, solvers, eigenvalues, SVD*).
4. Data struktur dinamis (*lists, queues, sets, trees, graphs*).
5. Pemroses citra fundamental (*filtering, edge detection, corner detection, sampling and interpolation, color conversion, morphological operations, histograms, image pyramids*).
6. Analisis struktur (*connected components, contour processing, distance Transform, various moments, template matching, Hough Transform, polygonal approximation, line fitting, ellipse fitting*).
7. Kalibrasi kamera (*calibration patterns, estimasi fundamental matrix, estimasi homography, stereo correspondence*).
8. Analisis gerakan (*optical flow, segmentation, tracking*).
9. Pengenalan obyek (*eigen-methods, HMM*).
10. Graphical *User Interface* (*display image/video, penanganan keyboard dan mouse handling, scroll-bars*).

OpenCV terdiri dari 3 *library*, yaitu:

1. CV : Untuk algoritma *Image Processing* dan *Vision*
2. Highgui : Untuk GUI, *Image* dan *Video I/O*
3. CXCORE : Untuk struktur data, support XML dan fungsi-fungsi grafis.

## 2.7 AUC (Area Under ROC Curve)

ROC (*Receive Operating Characteristics*) adalah hasil pengukuran klasifikasi dalam bentuk 2 dimensi. Penggunaan kurva ROC semakin populer dalam berbagai aplikasi terutama dalam bidang *image processing*. Berikut ada empat peluang yang dapat diformulasikan dalam table kontingensi  $2 \times 2$  untuk menganalisis ROC :

Kelas prediksi	Kelas sebenarnya	
	Benar	salah
Positif	Benar positif	Salah positif
Salah	Benar negatif	Salah negatif

Tabel 2.1 Tabel Kontingensi ROC

Untuk tingkat akurasi dapat diuraikan dari table diatas.

$$\text{Tingkat Akurasi} = \frac{(TP+TN)}{(TP+TN+FP+FN)} \times 100 \% \dots\dots\dots(20)$$

AUC adalah luas daerah di bawah kurva ROC. Bila nilainya mendekati satu, maka model yang di dapat lebih akurat. Karakteristik dari AUC adalah sebagai berikut:

1. Area maksimum adalah 1.
2. Jika ROC = 0.5 maka model yang dihasilkan belum terlihat optimal.
3. Sedangkan jika ROC > 0.5 maka model yang dihasilkan akan lebih baik.