BAB II

LANDASAN TEORI

Pada bab ini menjelaskan mengenai tempat dilakukannya penelitian dan menjelaskan mengenai teori yang akan digunakan dalam membuat sistem yang akan dibuat.

2.1. Grafika Komputer 2 Dimensi

Grafika Komputer 2 Dimensi merupakan gambar digital yang dihasilkan oleh komputer, khususnya yang berbentuk dua dimensi karena hanya mempunyai sumbu koordinat X dan Y saja. Gambaran grafika komputer 2D kira-kira adalah seperti sebuah kanvas kosong (pada komputer dianggap sebagai *frame buffer* yang berupa kumpulan *pixels*) berwarna putih (sebagai latar belakang warna, pada komputer latar belakang yang biasanya dipakai adalah berwarna hitam). Kanvas tersebut selanjutnya akan diberi gambar maupun warna sesuai dengan keinginan. Pada komputer, proses penggambaran dan pewarnaan sama dengan mengisi dengan bilangan tertentu sebagai petunjuk warna yang dipakai.

Grafika komputer 2 Dimensi dapat berupa grafik vektor, grafik raster maupun kombinasi antara keduanya. Grafik raster adalah penyimpanan gambar 2D pada komputer yang dilakukan secara apa adanya dalam bentuk *array of pixels* dan sering kali disebut sebagai file bitmap. Sedangkan garfik vektor merupakan proses penyimpanan gambar 2 Dimensi pada komputer yang dilakukan secara vektor, dimana data yang disimpan berupa posisi titik-titik yang kemudian dalam

proses penggambarannya dihubungkan dengan garis, ataupun bangun sehingga membentuk sebuah gambar.

2.1.1. *Image Processing*

Image Processing adalah sebuah metode yang dipakai untuk memproses atau memanipulasi image dalam bentuk dua dimensi. Segala operasi untuk mempebaiki, menganalisa, atau mengubah suatu gambar disebut *image processing* (A. Usman, 2005, Pengolahan Citra Digital dan Teknik pemogramannya). Konsep dasar pemprosesan suatu objek yang menggunakan *image processing* diambil dari kemampuan indera penglihatan manusia yang selanjutnya dihubungkan dengan kemampuan otak manusia. Dalam sejarahnya, *image processing* telah diaplikasikan dalam berbagai bentuk, dengan tingkat kesuksesan yang cukup besar. Seperti berbagai cabang ilmu lainnya, *image processing* pula gabungan berbagai cabang-cabang ilmu, diantaranya adalah optik, elektronik, matematika, fotografi, dan teknologi komputer. Beberapa faktor yang menyebabkan sistem *image processing* menjadi berkembang pesat pada saat ini, salah satu yang utama adalah dibutuhkannya suatu teknologi yang dapat bekerja secara mandiri, dalam artian teknologi tersebut dapat mengambil keputusan sendiri dari hasil pengolahan data sebelumnya. Selain itu juga tersedianya peralatan yang lebih *modern* untuk proses tampilan gambar yang menjadikan *image processing* berkembang pesat.

Pada umumnya, objektifitas dari *image processing* adalah mentransformasikan atau menganalisis suatu gambar sehingga informasi baru tentang gambar dapat dibuat dengan lebih jelas.

2.1.2 Sistem Warna RGB

Semua warna yang ada merupakan perpaduan dari 3(tiga) macam warna dasar yaitu: Merah(Red), Hijau(Green), Biru(Blue). Perpaduan ketiga warna dasar ini dipakai pada sistem warna RGB, apabila ketiga warna dasar dicampur, maka akan dihasilkan suatu warna tertentu, tergantung dari komposisi ketiga warna dasar tersebut.

Gambar pada sistem digital dapat diwakili denag format RGB untuk setiap titiknya, dimana setiap komponen warna diwakili dengan satu *byte.* Jadi untuk masing-masing komponen R, G, dan B mempunyai variasi dari 0 sampai 255. Total variasi yang dapat dihasilkan untuk sistem warna *digital* ini adalah 256 x 256 x 256 atau 16.777.216 jenis warna. Karena setiap komponen warna dieakili dengan satu *byte* atau delapan bit, maka total bit yang digunakan untuk merepresentasikan warna RGB adalah 8+8+8 atau 24 bit.

2.2 Teknik Morphing

Teknik morphing ini dimulai dengan cara-cara yang masih tradisional dan tidak memerlukan bantuan teknologi untuk membuatnya, tetapi hal itu dirasa belum cukup efektif dan sangat membutuhkan waktu yang relatif sangat banyak untuk menerapkannya. Selain membutuhkan waktu yang lama dalam proses pembuatannya, hasilnya juga masih tidak memuaskan.

Dengan bantuan komputer, efek morphing yang dihasilkan menjadi lebih bagus, dan waktu yang dibutuhkan relatif lebih cepat. Sejak tahun 1980an mulai banyak film-film yang menggunakan special effect dengan bantuan komputer. Contoh video yang menggunakan teknik image morphing adalah video klip

Michael Jackson yang berjudul Black or White dan Film yang berjudul Willow dan Indiana Jones and Last Srusade. Teknik morphing menggunakan komputer, pada dasarnya tidak berbeda dengan teknik tradisional yang telah ada sebelumnya dengan cara menggambar. Hanya saja, gambar yang dihasilkan oleh komputer lebih realistik sehingga dapat memberikan kesan bahwa gambar tersebut nyata.

Pada dasarnya pembuatan efek morphing dilakukan dengan membuat gambar-gambar peralihan diantara gambar asal dengan gambar tujuan berdasarkan kedua gambar tersebut. Dengan adanya gambar-gambar peralihan yang baru ini, maka perubahan yang terjadi dari gambar asal ke gambar tujuan akan terlihat lebih halus. Gambar-gambar peralihan inilah yang dibuat dengan menggunakan bantuan komputer. Dalam sebuah gambar teridiri dari banyak titik yang mempunyai nilai berbeda yang biasa disebut dengan pixel. Dalam proses pembuatan gambar-gambar peralihan oleh komputer titik-titik inilah yang akan mengalami perubahan. Perubahan yang terjadi dapat meliputi perubaha warna dan posisinya. Dengan adanya perubahan posisi dari titik-titik pada suatu gambar, maka akan mengakibatkan hilangnya beberapa titik pada gambar, sehingga gambar yang terbentuk nantinya akan terkesan berlubang, untuk mengatasi hal itu digunakan teknik interpolasi. Interpolasi ini digunakan untuk mengisi titik-titik yang hilang pada gambar dengan berpatokan pada titik-titik yang terisi yang ada di sekitarnya.

Ada beberapa algoritma yang dapat digunakan untuk membuat teknik morphing, di antaranya adalah cross dissolve, field morphing, mesh morphing, liquid morphing, dan deformation technique. Dimana masing-masing algoritma tersebut memiliki kelebihan dan kekurangan dalam hal waktu proses maupun

morphing yang dihasilkan. Pada pembuatan efek morphing, terdapat suatu proses yang disebut dengan deformasi. deformasi merupakan suatu teknik yang digunakan untuk mengubah bentuk obyek pada gambar asal menjadi obyek peralihan yang menyerupai obyek pada gambar tujuan. Teknik deformasi inilah yang membedakan antara algoritma yang satu dengan algoritma yang lainnya.

2.3. Proses Cross Dissolve

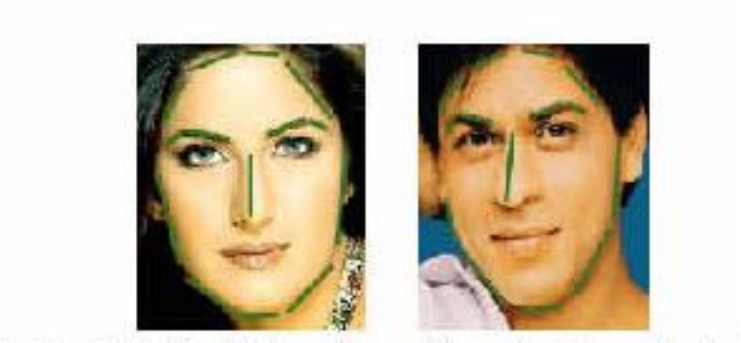
Proses cross dissolve dibutuhkan untuk memperhalus perubahan gambar yang terjadi. Cross dissolve ini merupakan tenik image morphing yang paling sederhana dan mudah. Teknik cross dissolve ini merupakan dasar dari teknik-teknik morphing yang lain. Pada teknik cross dissolve yang dilakukan hanya menampilkan kedua gambar (gambar asal dan gambar tujuan) dengan mode transparan. Tingkat transparansi dari setiap gambar diatur dengan maksimum 100%, dan minimum 0%. Pada tingkat transparansi 100%, maka yang terlihat hanyalah gambar asal, sedangkan pada tingkat transparansi 0% yang terlihat hanyalah gambar tujuan. Selama jumlah frame, tingkat transparansi akan semakin berkurang dengan syarat jumlah tingkat transparansi gambar asal dijumlahkan dengan gambar tujuan adalah 100%. Dengan demikian akan terlihat gambar asal perlahan-lahan menghilang sambil gambar tujuan terlihat perlahan-lahan muncul.

2.4. Field Morphing

Field morphing merupakan salah satu teknik dalam image morphing yang menggunakan bantuan garis feature untuk setiap perubahan yang terjadi, sehingga animasi yang terjadi dengan teknik field morphing ini harus menerima masukan

berupa dua buah gambar. Gambar pertama disebut sebagai image source dan gambar terakhir disebut sebagai image destination. Kegunaan dari garis Feature adalah untuk menunjukkan sebuah feature (bagian-bagian gambar yang membentuk obyek) yang sama antara daerah di gambar asal dengan daerah di gambar tujuan. Dengan adanya garis feature ini membuat komputer mengetahui hubungan antara dua obyek yang sama pada image source danimage destination. Sebuah garis feature di image source memiliki pasangannya di image destination.

Dengan adanya garis feature, maka komputer dapat mengetahui hubungan antara dua object yang dianggap sama di gambar asal dan tujuan. Dalam teknik Field Morphing terdapat dua buah langkah. Langkah pertama adalah deformasi image source dan image destination. Langkah kedua adalah cross dissolve antara kedua gambar yang telah dideformasikan.

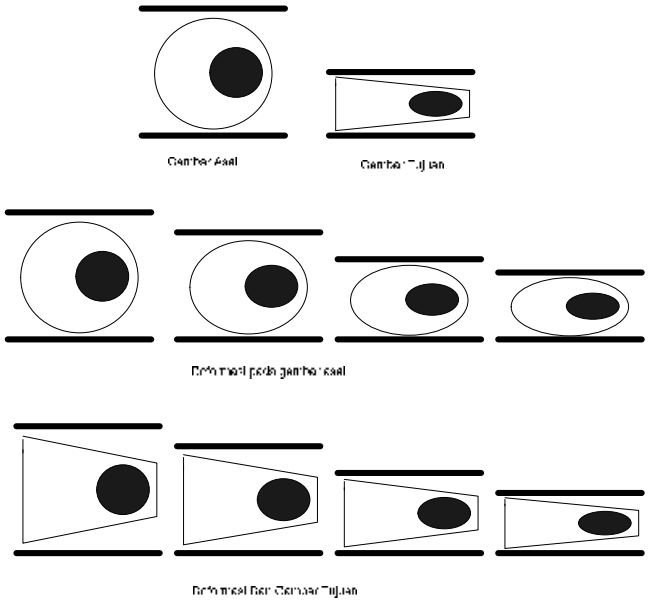


Gambar 2.1. Garis Feature Image Source dan Image Destination

2.4.1. Deformasi Image

Pada gambar 2.1 ditunjukkan garis feature dan hubungannya pada image source dan image destination. Dengan bantuan garis feature tersebut diatas, alis mata dari image source akan diubah menjadi alis mata dari image destination. Perubahan yang ada ditentukan oleh jumlah frame yang ada. Diantara image

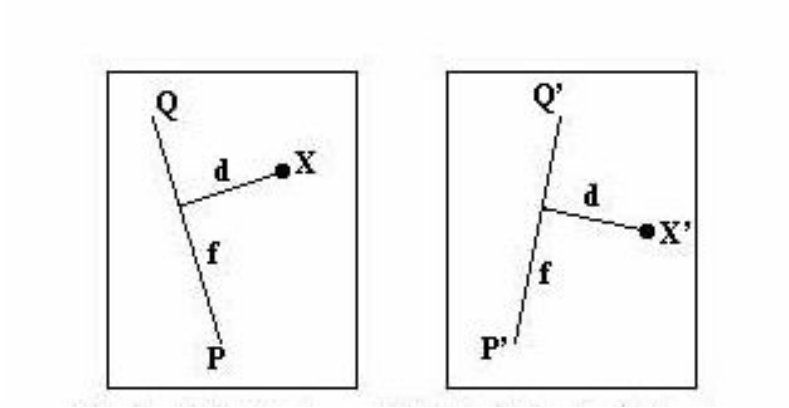
source dan image destination akan dilakukan interpolasi untuk mengisi animasi frame yang dibutuhkan. Parameter yang digunakan untuk interpolasi ini adalah frame saat ini dibagi dengan total frame.



Gambar 2.2. Deformasi Image Source dan Image Destination

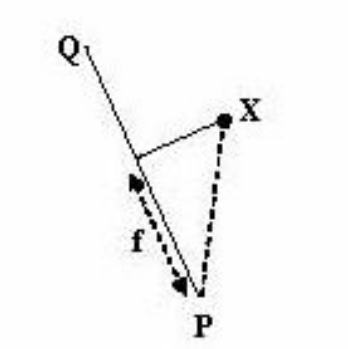
Sumber: Agustinus Nalwan. (2001). Seri Aplikasi Pemrograman Movie & Special Effect2.

Pada gambar 2.2. ditunjukkan image source dan image destination yang diinginkan. Proses deformasi tidak hanya terjadi pada image source, melainkan juga pada image destination. Pada contoh gambar 2.2. image source diubah menjadi image destination dalam 4 frame. Perubahan itu diubah berdasarkan garis feature yang diberikan. Dari hasil keduanya, baru dilakukan proses cross dissolve untuk memperhalus perubahan yang terjadi.



Gambar 2.3. Deformasi Titik Terhadap Garis *Feature*

Dengan menggunakan teknik *field morphing*, perubahan pada setiap titik harus dilakukan pencarian jarak terdekat antara titik dengan garis *feature*. Hal ini dapat dilakukan dengan mencari proyeksi vektor antara vektor PQ dan vektor PX.



Gambar 2.4. Pencarian proyeksi vektor PX terhadap PQ

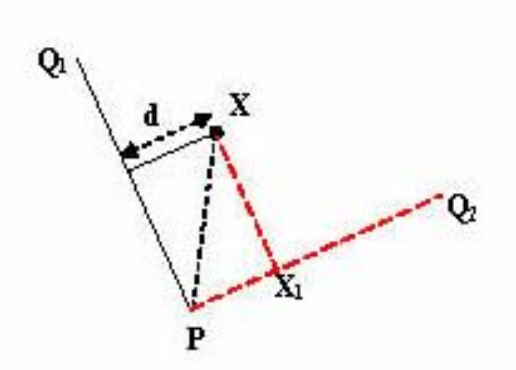
Pencarian proyeksi vektor PX terhadap titik PQ agar diddapat nilai dari f. Untuk mencari nilai dari f digunakan rumus :

$$f = \frac{\overrightarrow{PX} * . \overrightarrow{PQ}}{\left|\overrightarrow{PQ}\right|} \qquad (2.1)$$

Dalam perhitungan nilai f yang kita butuhkan kisaran antara 0 dan1, yaitu dengan cara dilakukan pembagian dengan panjang vektor PQ.

$$f = \frac{\overrightarrow{PX} * \overrightarrow{PQ}}{\left|\overrightarrow{PQ}\right|^2} \tag{2.2}$$

Setelah didapat nilai f langkah selanjutnya mencari nilai d yang merupakan jarak tegak lurus antara titik dengan garis *feature*.



Gambar 2.5. Pencarian jarak d

Dengan vektor tegak lurus PQ maka nilai d didapat dengan cara mencari vektor proyeksi PX terhadap PQ.

$$PX_1 = \frac{\overrightarrow{PX} * \overrightarrow{PQ_2}}{\left|PQ\right|} \tag{2.3}$$

PQ_2 harus mempunyai panjang yang sama dengan PQ_1 agar nilai PX_1 sama dengan panjang d. Persamaan f dan d akan mentranslasikan koordinat x dan y ke bentuk koordinat baru, yaitu koordinat d dan f. Koordinat f dalam sistem koordinat baru dapat diidentifikasikan sebagai berikut :

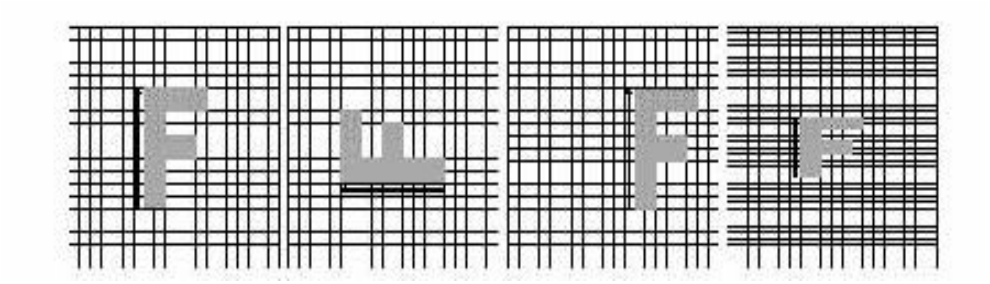
$$f \times \overrightarrow{P'Q'} \tag{2.4}$$

Sedangkan vector d dapat diidentifikasikan sebagai berikut :

$$\frac{\overrightarrow{\perp P^1Q^1}}{\overrightarrow{P^1Q^1}} \tag{2.5}$$

Jadi, titik asal X ditransformasikan ke titik tujuan X’ dengan rumus :

$$X' = P' + f \times P'Q' + d \times \frac{\overrightarrow{\perp P'Q'}}{\left|\overrightarrow{P'Q'}\right|} \quad (2.6)$$



Gambar 2.6. Pengaruh Satu Garis Feature

Sumber: http://www.hammerhead.com/thad/morph.html

Apabila garis feature hanya satu, maka X’ dapat dicari dengan menggunakan rumus tersebut. Tetapi apabila garis feature yang mempengaruhi sebuah image lebih dari satu maka untuk mendapatkan nilai X’ diperlukan pencarian bobot untuk setiap titik terhadap semua garis feature, semakin jauh jarak titik terhadap garis feature maka semakin kecil perubahan terhadap perubahan titik dan sebaliknya. Rumus pencarian bobot untuk setiap titik adalah sebagai berikut :

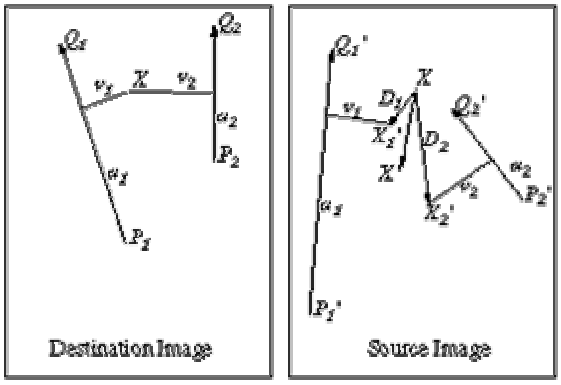
$$W = \left(\frac{1}{1 + dist} \right) \quad (2.7)$$

Dimana dist merupakan jarak terdekat antara titik dengan garis feature. Jarak terdekat yang berarti jika nilai f lebih dari 1 atau kurang dari 0, maka panjang dist tidak akan sama dengan nilai d, dikarenakan nilai d didapat dari perpanjangan dari garis feature. Sehingga apabila nilai f < 0 maka jarak terdekat antara titik dengan garis feature adalah jarak dari titik ke titik awal dari garis feature. Dan apabila nilai f > 1, maka jarak antara titik dengan garis feature adalah jarak dari titik ke titik akhir dari garis feature.

$$D_1 = X_1' - X$$
$$D_2 = X_2' - X \tag{2.8}$$

Akibat dari perhitungan bobot dan perubahan jarak dari setiap garis feature maka didapat rumus untuk perhitungan koordinat akhir yang dipengaruhi oleh lebih dari satu garis feature sebagai berikut :

$$X' = X + \frac{D_1 W_1 + D_2 W_2}{W_1 + W_2} \tag{2.9}$$



Gambar 2.7. Titik pada Image Source dan Image Destination yang Dipengaruhi oleh Lebih dari Satu Garis Feature

Rumus umum untuk mencari titik baru yang dipengaruhi oleh lebih dari satu garis feature adalah sebagai berikut :

$$X' = X + \frac{\sum_{i=1}^{n} D_1 W_1}{\sum_{i=1}^{n} W_1} \tag{2.10}$$

2.4.2. Algoritma Field Morphing

Algoritma keseluruhan dari field morphing adalah :

1. Menentukan jumlah frame gambar yang ingin dihasilkan
2. Mengerjakan proses warping sebanyak n kali. Setiap kali, dua proses

3. warping dikerjakan, yaitu warping image source dan warping image destination.

Hasil warping image source dan image destination kemudian mengalami proses cross dissolve untuk setiap frame gambar.

2.5. Deformation Technique

Metode image morphing yang lain adalah deformation techniques. Metode image morphing deformation techniques menggunakan turunan dari C1 – continuous and one to one warp dari setiap titik yang ada pada dua image. Metode deformation techniques lebih sering disebut juga dengan metode energy minimization. Metode deformation technique menggunakan garis feature untuk menunjukkan hubungan antara image source dan image destination.

2.5.1. Fungsi *warphing* pada *deformation technique*

Rumus yang digunakan pada *deformation technique* befokus pada energi yang diharapkan dapat menggunakan energi yang paling minimal.

$$Ep(w) = \beta \sum_{t} \| w(p_i) - q_i \|^2 \tag{2.11}$$

Dimana q_i merupakan posisi baru yang diinginkan dari titik q_i pada. Parameter ß mengatur kepekaan suatu titik dengan titik yang lainnya agar pixel pada image tidak mengalami banyak perubahan.

Rumus energi dari fungsi w adalah.

$$E_\beta(w) = \iint \Omega \left[\left\| \frac{\partial^2 w}{\partial u^2} \right\|^2 + 2 \left\| \frac{\partial^2 w}{\partial u \partial v} \right\|^2 + \left\| \frac{\partial^2 w}{\partial v^2} \right\|^2 \right] du dv \tag{2.12}$$

Hasil dari fungsi w adalah *continous partial derivatives*, ?w/?u dan ?w/?v. Dengan menggunakan C^1- *continuity*, satu per satu pixel dan fungsi w akan menggunakan *minimizing the jacobian energy,*

$$E_j(w) = \alpha \iint \Omega (J-1)^2 \, dudv, \tag{2.13}$$

Dimana $$J = \frac{\partial x \partial y}{\partial u \partial v} - \frac{\partial x \partial y}{\partial v \partial u} \tag{2.14}$$

Fungsi w akan merupakan *one to one* pada?, jika nilai J pada rumus jacobian tidak menghasilkan nilai 0 pada ruang dari ? dan jika w merupakan batas dari ?. Dari fungsi-fungsi energy yang ada, maka dengan hukum kekekalan energy, maka didapat rumus,

$$ED(w) = \frac{1}{2}(E_q(w) + E_j(w) + W_p(w)) \tag{2.15}$$

Jika fungsi w merupakan nilai terkecil dari sebuah fungsi energy ED(w), maka ED(w) harus disederhanakan dengan menggunakan turunan.

$$\frac{sED}{sw} = \frac{1}{2}\left(\frac{sE_q}{sw} + \frac{sE_j}{sw} + \frac{sE_p}{sw}\right) = 0, \tag{2.16}$$

Dimana

$$\frac{sE_q}{sw} = 2\left(\frac{\partial^4 w}{\partial u^4} + 2\frac{\partial^4 w}{\partial u^2 \partial v^2} + \frac{\partial^4 w}{\partial v^4}\right) \tag{2.17}$$

$$\frac{sE_j}{sw} = 2\alpha\left(\frac{\partial}{\partial u}\left(J\frac{\partial w^{\perp}}{\partial v}\right) - \frac{\partial}{\partial v}\left(J\frac{\partial w^{\perp}}{\partial u}\right)\right) \tag{2.18}$$

$$\frac{sE_p}{sw} = 2\beta(w(p_i) - q_i) \tag{2.19}$$

Dengan w- merupakan vektor (-y,x) yang tegak lurus pada vektor w = (x,y). Differential partial yang digunakan untuk fungsi ED(w) biasa disebut dengan

Euler-Lagrange equation. Dari perumusan yang ada apabila disederhanakan penulisannya dengan titik-titik yang ada, maka penulisan rumusnya akan menjadi,

$$Aw + aj(w) + ß(I'w - q) = 0. \quad (2.20)$$

Dimana, A merupakan matriks MN × MN yang merupakan koefisien dari setiap titik yang ada dari turunan Des/ dw, sedangkan j(w) merupakan vektor dengan dimensi MN yang merupakan turunan dari Jacobian force Dej/dw pada sebuah variabel. I' merupakan hasil korespodensi dari setiap titik. Untuk sebuah perubahan digunakan kondisi ?w/?u = ?r / ?u, ?w/?u = ?r/?v, dan ? 2 w/?u 2 =?2 w/?v 2 = 0 dengan menggunakan penurunan matriks A dan vektor j(w).

Maka didapatkan rumus,

$$\frac{\partial w}{\partial t} = Aw + \Im(w) + \beta(\int w - q) \quad (2.21)$$

Untuk menurunkan fungsi w yang ada, maka t diinisialisasikan sebagai bilangan yang tak terhingga (8). Ketika dijalankan menjadi sebuah waktu t maka hasil dari t akan dipengaruhi oleh waktu t-1 atau waktu sebelumnya. Untuk menghitung titik x dan y dari j(w), x dan y dapat diasumsikan dalam satuan waktu. Sehingga penulisannya,

$$-y(x_1 - x_{t-1}) = Ax1 + aB(x_{t-1})y1 + ß(I^1x1 - x) \quad (2.22)$$

$$-y(y1 - yt\text{-}1) = Ay1 + aB(x_{t-1})y1 + ß(I^1y1 - y2) \quad (2.23)$$

Pada rumus diatas dapat diketahui bahwa xt dan yt dapat dihasilkan dari xt-1 dan yt1. Untuk menurunkan fungsi W0 yang ada, harus diketahui titik-titik yang berkesinambungan pada image source dengan image destination. Sehingga dari titik-titik yang berkesinambungan tersebut didapatkan komponen dari W0 adalah X dan Y.

$$W0(x,y) = (X(x,y),Y(x,y)) \quad (2.24)$$

Pertama-tama yang harus dilakukan adalah menghitung komponen yang pertama yaitu X baru dilakukan perhitungan untuk komponen yang kedua yaitu Y.

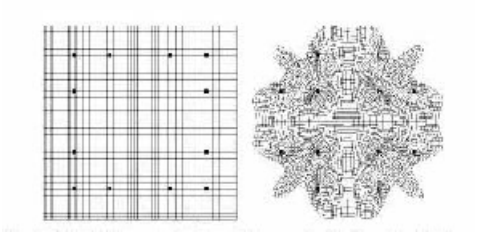
$$X(x,y) = x + ?X(x,y) \quad (2.25)$$

Dimana

$$?X(x0,y0) = ?x = x1 - x0 \quad (2.26)$$

$$Pi(xi,yi),\ I= 0,1,\ dan\ (P0,P1)\ € S \quad (2.27)$$

Karena W0 merupakan fungsi yang memindahkan titik P0 menjadi titik P1 untuk setiap pixel pada image source.



Gambar2.8.Pixel yang terbentuk setelah *deformation technique*

2.5.2 Fungsi transisi pada *deformation technique*

Untuk memberikan transisi pada setiap titik pada image source maka digunakan suatu penurunan rumus energy minimization. Untuk menghasilkan sebuah transisi yang halus dan menghemat energi, maka sebuah fungsi f merupakan turunan dari fungsi minimizing energy.

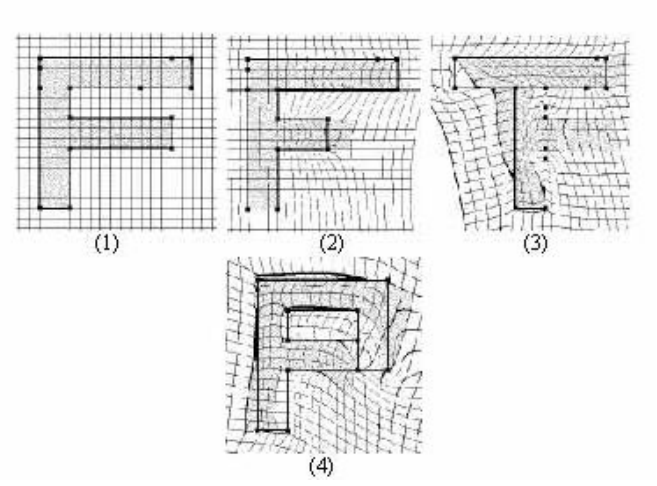
$$ES(f) = \frac{1}{2}\left(\iint \Omega \left[\left(\frac{\partial^2 f}{\partial u^2} \right)^2 + 2\left(\frac{\partial^2 f}{\partial u \partial v} \right)^2 + \left(\frac{\partial^2 f}{\partial v^2} \right)^2 \right] dudv + \beta \sum_i \left(f(p_2) - i_2 \right)^2 \right)$$

(2.28)

t_i merupakan nilai dari setiap point p_i pada ?. Jika fungsi f yang telah digabungkan dengan fungsi energy diturunkan, maka kondisi dari ekspresi tersebut,

$$\frac{sES}{sf} = \frac{\partial^4 f}{\partial u^4} + 2\frac{\partial^4 f}{\partial u^2 \partial v^2} + \frac{\partial^4 f}{\partial v^4} + \beta(f(p_i) - i_i) = 0 \qquad (2.29)$$

Apabila disederhanakan dengan sebuah simbol matriks, maka fungsi f dapat menghasilkan sebuah nilai pada image. Dengan menggunakan aljabar linear, maka f akan diubah menjadi vektor f yang berdimensi MN.Untuk implementasi dalam image source dan image destination, maka transisi ini dapat digambarkan sebagai animasi yang terbentuk pada saat titik P0 pada image source (I0) akan diubah menjadi titik (P0) pada image destination (I1). Dengan adanya *frame-frame* yang digunakan maka didapat sebuah fungsi T0(t) dengan menurunkan fungsi *energy*.Dimana fungsi ?T merupakan selisih dari setiap transisi ?tPada setiap transisi akan terjadi perubahan titik-titik pada image source menjadi image destination. Pada image source titik tersebut akan diubah tetapi perubahan yang terjadi pada deformation techniques tidak merusak semua susunan pixel, tetapi hanya titik-titik dan sekitar titik-titik yang berhubungan.



Gambar2.9. Perubahan pixel dari F ke T dengan *deformation technique*

Setelah teknik tersebut diterapkan untuk menghasilkan image morphing, maka untuk memperhalus pembentukannya, maka digunakan proses cross dissolve.

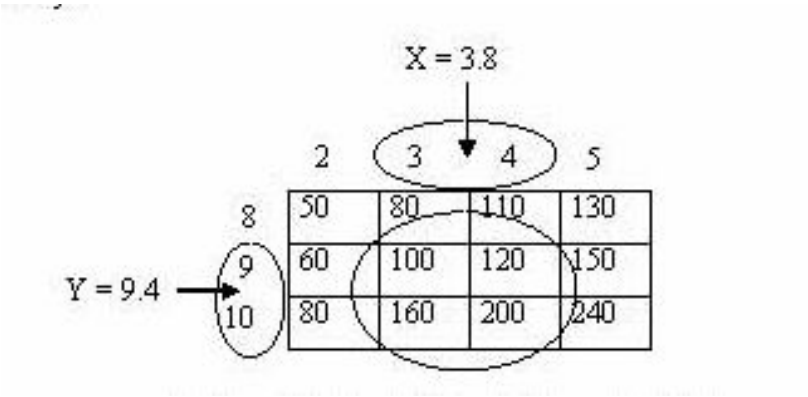
2.6. Tweening

Tweening lebih dikenal dengan teknik interpolasi. Melakukan interpolasi berarti menyisipkan sesuatu di antara dua bagian yang berbeda atau memperkirakan nilai dari suatu fungsi antara dua nilai yang telah diketahui (Sumber : *Morphing Magic* 1 st *edition).*

2.7. Interpolasi Bilinear

Operasi geometrik berhubungan dengan perubahan bentuk geometri citra, yaitu baik ukuran maupun orientasinya. Operasi geometri di antaranya meliputi pencerminan (flipping), rotasi/pemutaran (rotation), penskalaan (scaling/zooming), dan pembengkokan (warping). Operasi geometrik dilakukan dengan cara memetakan koordinat titik-titik citra asal ke koordinat titik-titik di citra hasil. Pemetaan dilakukan dengan menerapkan fungi interpolasi bilinear.

Pada metode interpolasi bilinear, maka titik-titik yang terdekat akan mempengaruhi hasil dari titik yang akan dicari. Makin dekat titik tetangga, maka makin besar bobotnya, dan sebaliknya makin jauh maka akan makin kecil bobotnya.



Gambar 2.10. Contoh Pixel Tetangga terdekat

Dari contoh Gambar 2.25, maka bobot ke arah horisontal untuk koordinat x = 3 dan x = 4 masing-masing adalah :

w(3) = 0.2 dan w(4) = 0.8 (2.30)

Koordinat x = 3.8 lebih dekat ke x = 4 dibandingkan dengan ke x = 3. Sementara bobot ke arah vertikal untuk koordinat y = 9 dan y = 10, masing-masing adalah :

w(9) = 0.6 dan w(10) = 0.4 (2.31)

Dari bobot vertikal dan bobot horisontal yang didapat, maka dapat dihitung bobot keseluruhan suatu titik.

w(3,9) = w(3) × w(9) = 0.2 × 0.6 = 0.12 (2.32)

w(3,10) = w(3) × w(10) = 0.2 × 0.4 = 0.08 (2.33)

w(4,9) = w(4) × w(9) = 0.8 × 0.6 = 0.48 (2.34)

w(4,10) = w(4) C w(10) = 0.8 × 0.4 = 0.32 (2.35)

Terlihat bahwa titik yang paling dekat dengan koordinat titik asal memiliki bobot yang paling besar (0.48), sedangkan titik terjauh memiliki bobot yang paling kecil (0.08). Total bobot untuk keempat titik tersebut adalah 1.

Setelah semua bobot pada keempat titik bertetangga tersebut diperoleh, maka perhitungan untuk mendapatkan sebuah titik hasil adalah :

Kå ´ = ?w x Ki (2.36)

= w(3,9)xK?(3,9)+w(3,10)xK?(3,10)+w(4,9)xK?(4,9)+w(4,10)xK?(4,10) (2.37)

= 0.12 x100 + 0.08 x 160 + 0.48 x 120 + 0.32 x 200 (2.38)

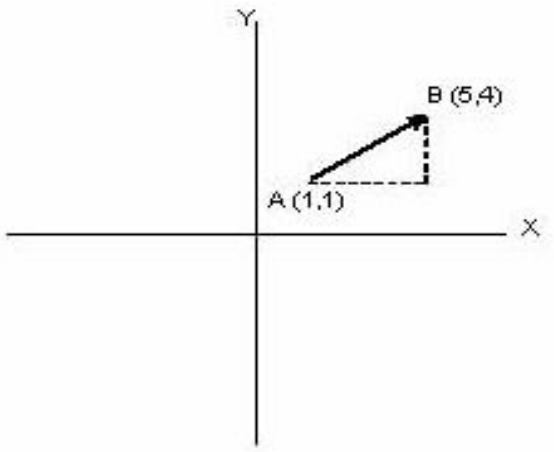
= 146.4 (2.39)

Karena nilai sebuah titik merupakan bilangan bulat, maka apabila hasil penjumlahan berbobot tersebut bilangan pecah, maka harus dilkukan pembulatan ke bilangan bulat terdekat. Kelebihan dari interpolasi bilinear adalah citra hasilnya

terlihat lebih halus (smooth) dibandingkan dengan hasil interpolasi tetangga terdekat, terutama jika dilakukan pembesaran atau zoom-in terhadap suatu citra.

2.8. Vektor

Vektor merupakan sebuah format untuk menyatakan sebuah perpindahan dari satu tempat ke tempat lainnya yang terdiri atas arah dan panjang. Vektor selalu mempunyai dua unsur, yaitu besar dan arah. Dalam dunia dua dimensi, Vektor memiliki dua buah komponen, yaitu x dan y. x merupakan komponen arah horisontal (sumbu X), sedangkan y merupakan komponen arah vertikal (sumbu Y). Dalam dunia tiga dimensi, vektor memiliki tiga komponen, yaitu x, y, dan z. Dalam penerapannya komponen x, y, dan z sering disebut sebagai i, j, k. Tambahan komponen z dalam dunia 3 dimensi merupakan sebuah komponen untuk mendeskripsikan arah kedalaman (sumbu Z).



Gambar 2.11. Vektor AB = 4x + 3y

Garis AB merupakan vektor dengan komponen 4x+ 3y. Vektor ini dapat diartikan sebagai perpindahan dari titik awal (titik A) ke titik akhir (titik B) Pada vektor AB karena dari titik A ke titik B diperlukan perpindahan koordinat sejauh 4 (koordinat X dari B dikurangi koordinat X dari A), maka komponen x berharga 4. Sedangkan pada sumbu Y dibutuhkan perpindahan sejauh 3 (Koordinat Y dari B dikurangi koordinat Y dari A), maka komponen y berharga 3.

2.9. Multimedia

Komputer sebagai alat pengolahan *digital* saat ini hampir dimiliki oleh setiap orang, seperti halnya televisi, komputer juga banyak digunakan sebagai media hiburan. Dengan perkembangan komputer (serta pengetahuan dalam pengolahan sinyal *digital*), maka data-data dalam bentuk *digital* semakin banyak digunakan.

Penggunaan data *digital* baik berupa text, suara, citra maupun video sangat pesat, ditambah dengan perkembangan teknologi jaringan antar komputer didunia (internet), pertukaran data *digital* semakin mudah untuk dapat menggabungkan semua data *digital* yang ada (multimedia).

Dalam pembuatan aplikasi-aplikasi multimedia diperlukan beberapa komponen pendukung seperti *text*, suara / audio, dan citra / *image.*

2.9.1. Data *Text*

Text merupakan sekumpulan karakter terdiri dari huruf-huruf, angka-angka (A-Z, a-z, 0-9), dan simbol-simbol lainnya seperti %, &, ^, =, @, £, $, !,*, dan lain-lain, dengan menggunakan kode ASCII setiap karakter dari *text* berjumlah 8 bit atau 1 *byte.*

2.9.2. Data *Image*

Citra / *images* adalah gambar *digital* dua dimensi yang dihasilkan dari gambar analog, dibagi kedalam N baris dan M kolom sehingga menjadi gambar diskrit melalui proses *sampling* (proses pengambilan *sample* warna sebelum melakukan proses digitalisasi citra), dan persilangan antara baris dan kolom disebut dengan *pixel. Images* diklasifikasikan ke dalam dua jenis yaitu :

1. *Images* Tetap

   Images / gambar mati (grafik statis) yang tidak mengalami proses animasi biasanya berupa format BMP, JPG, JPEG, ICO dan lain-lain.

2. *Images* Bergerak /Animasi

Image bergerak adalah *image* tetap yang diberikan efek seolah-olah seperti bergerak, atau sering disebut animasi. Animasi adalah proses kejadian dalam pembuatan film (*movie*), saat ini banyak film animasi tiga dimensi daripada dua dimensi.

2.10. *Software* Pendukung

2.10.1. *Borland Delphi* Versi 7.0

Borland Delphi atau yang biasa disebut dengan delphi saja merupakan sarana pemrograman aplikasi visual. Bahasa pemrograman yang digunakan adalah bahasa pemrograman pascal. Delphi merupakan generasi penerus dari turbo pascal. Turbo pascal yang diluncurkan pada tahun 1983 dirancang untuk dijalankan pada sistem operasi DOS, sedangkan delphi yang diluncurkan pertama kali tahun 1995 dirancang untuk beroperasi dibawah sistem operasi *windows*. Borland delphi merupakan salah satu bahasa pemrograman yang semenjak diluncurkan pertama kali langsung dilirik dan diminati oleh para programer komputer. Hal ini disebabkan karena delphi menyediakan fasilitas untuk pembuatan aplikasi visual secara mudah dan dapat memberikan hasil yang memuaskan. Keunggulan bahasa pemrograman ini terletak pada produktivitas, kualitas, pengembangan Perangkat Lunak, kecepatan kompilasi, pola disain yang menarik serta diperkuat dengan pemrograman yang terstruktur. Keunggulan lain dari delphi adalah dapat digunakan untuk merancang program aplikasi yang memiliki tampilan seperti program aplikasi lain yang berbasis *windows*.

2.10.2 *Komponen TImage*

Untuk menghasilkan suatu aplikasi pengolahan citra, maka delphi menyediakan sarana untuk menampilkan citra, yaitu melalui komponen Timage yang ter dapat pada palet Additional.

Komponen ini memiliki properti picture yang dapat digunakan untuk menyimpan data citra. Citra yang akan ditampilkan diambil dari file gambar yang dapat ditentukan baik pada saat mendesain dengan cara mengisi nilai properti maupun pada saat program dijalankan dengan menggunakan procedure LoadFromFile. Subproperti yang paling penting pada picture antara lain adalah

1. *Height,* berisi nilai tinggi citra
2. *Width*, berisi lebar citra
3. *Bitmap*, berisi data format dan *pixels citra*.

2.10.3 Komponen AVILoade*r*

Komponen AVILoader merupakan komponen tambahan pada delphi yang menyediakan fasilitas untuk membuka sebuah file AVI, memainkan , dan mengambilnya dalam bentuk image per frame. Komponen AVILoader menggunakan file dll yang dimiliki oleh windows yang bernama AVIfill32.dll.

Dalam komponen AVILoader ini, proses pengambilan frame dala sebuah file AVI dengan membaca per pixel dari AVI yang dijalankan, sehingga pada waktu proses pengambilan frame tidak diperlukan adanya penyimpanan file bitmap per frame.

2.10.4 Komponen TGIFImage

GIF merupakan singkatan dari *Graphics Interchange Format*, sebuah file image yang mampu menyimpan sebuah animasi dengan kualitas yang baik tetapi dengan size yang kecil apabila dibandingkan dengan file.AVI. File standart milik GIF adalah GIF89a. Pada delphi apabila tidak menggunakan komponen tambahan, maka file dengan format GIF ini tidak akan bisa dibuka ataupun disimpan dalam bentuk animasi GIF.

Komponen ini merupakan komponen tambahan pada delphi yang berfungsi untuk menyimpan maupun membuka file dengan format GIF. Komponen ini juga dapat memberikan bantuan agar sebuah form dapat membuka suatu animasi yang ber format GIF.

2.10.5 *Komponen TAVIWriter*

Komponen TAVIWriter merupakan komponen tambahan pada delphi yang menyediakan fasilitas untuk menghasilkan file dalam format AVI. Komponen TAVIWriter ini menggunakan file dll yang telah dimiliki oleh Windows, procedure-procedure yang ada pada komponen TAVIWriter ini beragam. Tetapi komponen yang digunakan dalam Tugas Akhir ini hanya komponen penyimpanan dalam bentuk file AVI.

Karena menggunakan file dll yang telah dimiliki oleh windows, maka komponen ini tidak bisa bekerja pada OS lain, selain Windows. Propeti yang dimiliki oleh komponen TAVIWriter antara lain:

1. *Bitmaps :* berisi gambar-gambar bitmap yang akan disatukan dan disimpan dalam file AVI.

2. *Height, Width* : berisi dimensi file AVI.
3. *Frame Time* : berisi durasi dalam satuan miliseconds
4. *File Name* : berisi nama file AVI.