

RANCANG BANGUN ROBOT PANTILT : PENDETEKSIAN POSISI TANGAN BERBASIS ALGORITMA GENETIKA

Ingot Samuel S Siagian^{#1}, Indra Adji Sulistijono^{#2}

Jurusan Teknik Elektronika, Politeknik Elektronika Negeri Surabaya
Kampus PENS-ITS Sukolilo, Surabaya

¹ingot@student.eepis-its.edu

²indra@eepis-its.edu

Abstrak—Paper ini membahas tentang keuntungan dan kekurangan dari penentuan posisi tangan manusia dengan menggunakan Algoritma Genetika. Algoritma Genetika merupakan salah satu metode optimasi yang efektif digunakan dalam melakukan optimasi sistem yang kompleks yang tidak dapat diselesaikan dengan metode optimasi konvensional. Dijelaskan juga bagaimana metode Human Hand Template dapat membedakan tangan manusia dengan objek lainnya. Nantinya semua metode ini diaplikasikan pada Robot Pan-Tilt agar mempunyai kemampuan berinteraksi dengan manusia.

Kata kunci: Pengolahan Gambar, Human Hand Template, Algoritma Genetika, Robot Pan-Tilt.

I. PENDAHULUAN

Pan-Tilt robot merupakan sebuah robot yang dapat berkomunikasi dengan manusia dengan cara mendeteksi bagian tubuh manusia dan menentukan perbedaan dengan bentuk-bentuk benda yang lain disekitarnya. Hal ini merupakan salah satu dasar dari cara manusia berkomunikasi, ketika mata mengamati sesuatu maka mata harus dapat mengikuti gerakannya dan membedakannya dengan benda lain disekitarnya tersebut.

Robot ini adalah salah satu bentuk aplikasi dari *computer vision*. Pant-Tilt robot mendeteksi tangan manusia dengan metode human hand template dan kemudian mengikuti pergerakannya dengan metode algoritma genetika. Disebut Pan-Tilt karena robot ini terdiri dari 2 gerakan yaitu menggeleng dan mengangguk. Sedangkan tangan

yang dimaksud, meliputi ujung jari tangan hingga pergelangan tangan.

Warna dari tangan dapat digunakan sebagai sumber informasi untuk proses pendeteksian. Informasi yang diperoleh dari warna tangan sangat relevan untuk proses pendeteksian antara lain untuk proses-proses pendeteksian manusia, pendeteksian wajah dan penjejakan wajah (face tracking), penjejakan tangan yang meliputi pengenalan posisi tangan dan gerakannya (gesture). Dalam hal ini warna tangan diperjelas dengan menggunakan sarung tangan berwarna.

Keuntungan yang dapat diperoleh jika warna dijadikan sumber awal informasi untuk proses-proses yang disebut di atas antara lain adalah memungkinkannya pemrosesan yang cepat dan langsung, tahan (robust) terhadap variasi geometris dari pola-pola/ tekstur kulit, tahan terhadap perubahan resolusi citra dan mengurangi ketergantungan pada penggunaan peralatan penjejakan yang khusus. Selain itu karena tangan yang memiliki karakteristik warna, hal ini memberikan kemudahan lain karena warna adalah salah satu aspek yang dengan cepat dan mudah dapat dikenali oleh manusia.

Algoritma genetika merupakan metode optimasi yang banyak digunakan untuk menyelesaikan masalah optimasi yang kompleks dan sukar diselesaikan dengan metode konvensional. Algoritma genetika diharapkan dapat memberikan hasil yang maksimal dalam melakukan pendeteksian posisi tangan manusia

II. PEMBUATAN SOFTWARE

A. Pengolahan Gambar

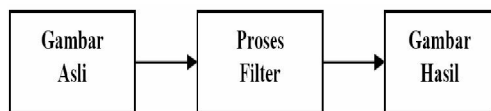
Image yang diperoleh dari lingkungan masih terdiri dari warna yang sangat kompleks sehingga

masih diperlukan proses lebih lanjut agar image tersebut dapat diolah ke dalam sebuah sistem.

Image processing atau sering disebut dengan pengolahan gambar digital merupakan suatu metode yang digunakan untuk mengolah atau memproses dari gambar asli sehingga menghasilkan gambar lain yang sesuai dengan kebutuhan [6]. Pengambilan gambar bisa dilakukan oleh kamera video atau alat-alat yang lain yang dapat digunakan untuk mentransfer gambar. Dalam pengolahan citra, dilakukan operasi terhadap citra asli menjadi citra baru berdasarkan citra asli. Operasi yang dilakukan pada citra dikategorikan sebagai berikut :

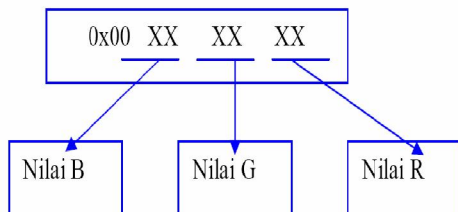
1. Point, yaitu operasi yang menghasilkan output dimana setiap pixel hanya dipengaruhi oleh pixel pada posisi yang sama citra asli.
2. Local, yaitu operasi yang menghasilkan output dimana pixelnya dipengaruhi oleh pixel-pixel tetangga pada citra asli.
3. Global, yaitu operasi yang menghasilkan output dimana pixelnya dipengaruhi oleh semua pixel yang ada dalam citra asli.

Misalnya ada suatu gambar yang terlalu gelap maka dengan *image processing* gambar tersebut bisa diproses sehingga mendapatkan gambar yang jelas. Secara garis besar bisa digambarkan seperti blok diagram pada gambar 2.1 dibawah ini:



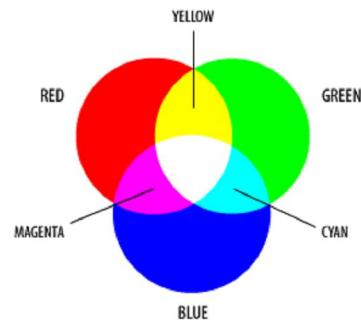
Gambar 1 Blok diagram pengolahan citra

Dasar dari pengolahan citra adalah pengolahan warna RGB pada posisi tertentu. Dalam pengolahan citra warna dipresentasikan dengan nilai hexadesimal dari 0x00000000 sampai 0x00ffffff. Warna hitam adalah 0x00000000 dan warna putih adalah 0x00ffffff. Definisi nilai warna di atas seperti gambar 2.2, variabel 0x00 menyatakan angka dibelakangnya adalah hexadecimal.



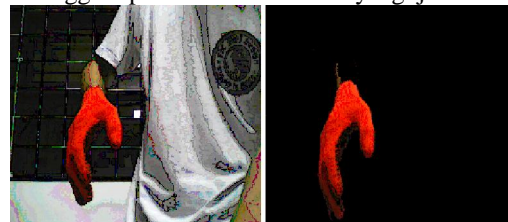
Gambar 2 Nilai warna RGB dalam hexadesimal

Terlihat bahwa setiap warna mempunyai range nilai 00 (angka desimalnya adalah 0) dan ff (angka desimalnya adalah 255), atau mempunyai nilai derajat keabuan $256 = 2^8$. Dengan demikian range warna yang digunakan adalah $(2^8)(2^8)(2^8) = 224$ (atau yang dikenal dengan istilah True Colour pada Windows). Nilai warna yang digunakan di atas merupakan gabungan warna cahaya merah, hijau dan biru seperti yang terlihat pada gambar 2.3. Sehingga untuk menentukan nilai dari suatu warna yang bukan warna dasar digunakan gabungan skala kecerahan dari setiap warnanya.



Gambar 3 Komposisi warna RGB

Pada proses ini cukup menggunakan filtering untuk menghilangkan warna hijau dan biru, sehingga diperoleh warna merah yang jelas.



Gambar 4 Komposisi warna RGB

```

// syntax program
for(j=0;j<bm.bmHeight;j++)
  for(i=0;i<bm.bmWidth;i++)
  {
    w=dcMem.GetPixel(i,j);
    WarnaToRGB(w,&r,&g,&b);
    if (r < (g+b))
      dcMem.SetPixel(i,j,0);
  }
  
```

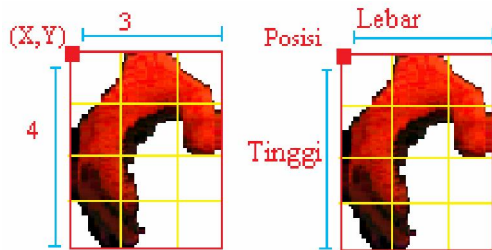
III. METODE PENDETEKSIAN

A. Human Hand Template

Proses ini merupakan metode pemetaan tangan kedalam bentuk koordinat pixel dari warna tangan.

Metode human hand template yang dimaksud adalah metode pemetaan posisi tangan kedalam

bentuk persegi panjang, dengan panjang kedua sisinya yang beraneka ragam tergantung dari banyaknya jumlah pixel yang masuk ke dalam kategori perbandingan warna tangan dengan warna dari pixel yang lainnya. Dengan metode ini, maka software dapat mengetahui koordinat dari posisi tangan manusia dengan membedakannya dengan objek lainnya.



Gambar 5 Human Hand Template

Semua nilai koordinat dari pixel warna di dalam template, menggambarkan banyaknya jumlah gen yang dimiliki oleh suatu individu. Gen tersebut terdiri atas posisi koordinat x dan y, lebar serta tinggi sehingga menghasilkan gambar kotak.

B. Algoritma Genetika

Algoritma Genetika adalah algoritma yang memanfaatkan proses seleksi alamiah yang dikenal dengan proses evolusi. Dalam proses evolusi, individu secara terus-menerus mengalami perubahan gen untuk menyesuaikan dengan lingkungan hidupnya. "Hanya individu-individu yang kuat yang mampu bertahan". Proses seleksi alamiah ini melibatkan perubahan gen yang terjadi pada individu melalui proses perkembangbiakan. Dalam algoritma genetika ini, proses perkembang-biakan ini menjadi proses dasar yang menjadi perhatian utama, dengan dasar berpikir: "Bagaimana mendapatkan keturunan yang lebih baik". GA secara khusus dapat diterapkan untuk memecahkan masalah optimisasi yang kompleks. Karena itu GA baik untuk aplikasi yang memerlukan strategi pemecahan masalah secara adaptif.

Algoritma Genetika mensimulasikan proses yang terjadi pada populasi alamiah yang merupakan hal penting dalam proses evolusi. Di alam, individu di populasi saling bersaing untuk memperoleh sumber daya seperti makanan, baju, tempat tinggal dan pekerjaan. Individu yang berhasil akan bertahan hidup sedangkan individu yang tidak, akan mati dan punah. Semua proses algoritma meliputi:

Awal

Inisialisasi

Perulangan

Seleksi

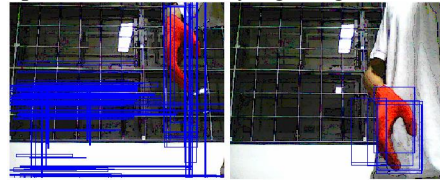
Kawin silang\Crossover

Mutasi

Sampai ditemukan nilai terbaik

Akhir

Inisialisasi merupakan proses untuk membangkitkan tiap-tiap individu. Populasi awal ini dibangkitkan secara random sehingga didapatkan solusi awal. Populasi itu sendiri terdiri dari sejumlah kromosom yang merepresentasikan solusi yang diinginkan.



Gambar 6 Pembangkitan Individu

Dalam membentuk individu terbaik, digunakan tiga operator yang telah disebut di atas yaitu operator reproduksi/seleksi, crossover dan mutasi. Proses ini dilakukan berulang-ulang sehingga didapatkan jumlah kromosom yang cukup untuk membentuk generasi baru dimana generasi baru ini merupakan representasi dari solusi baru.

1). *Seleksi*: Setelah dibangkitkan solusi secara acak, maka selanjutnya dihitung nilai fitness. Berdasarkan persamaan fitness (fitness function) diperoleh individu terbaik hingga individu terjelek. Nilai *fitness* dihitung dengan menggunakan persamaan:

$$f_i^{hd} = f_i^{hd} + C_{skin}^{hd} + (C_1 * C_{skin}^{hd} - C_2 * C_{other}^{hd})$$

Keterangan:

f_i^{hd} = Nilai fitness

C_{skin}^{hd} = Jumlah pixel warna tangan

C_{other}^{hd} = Jumlah pixel warna yang lain

C_1 dan C_2 = Konstanta untuk mempercepat proses

2). *Crossover*: Dilakukan dengan menggunakan pertukaran gen dari 2 induk secara acak. Diharapkan dari pertukaran ini dihasilkan 2 individu yang mempunyai nilai fitness lebih baik.

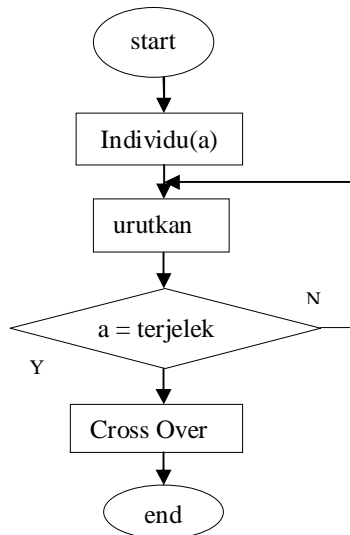
Elite crossover digunakan dengan melakukan pertukaran gen antara individu terbaik dengan individu terjelek. Probability crossover

digunakan untuk menentukan gen/kromosom diganti seluruhnya ataukah diganti hanya sebagian gen/kromosom. Nilai probability antara 0 sampai 1.

	individu	fitness
terbaik	00111001 =	0.984
	10010010 =	0.562
	01000101 =	0.234
	11010010 =	0.168
terjelek	11000010 =	0.015

Elite Crossover:

00111001 =	0.984 induk 1
11000010 =	0.015 induk 2
00111001 =	0.984 anak pengganti induk 2



Gambar 7 Crossover

3). *Mutasi*: Proses mutasi adalah proses pertukaran gen didalam individu itu sendiri. Caranya adalah dengan memberikan nilai inversi pada gen atau menggeser nilai gen pada gen yang terpilih untuk dimutasikan. Bila pada crossover, maka individu dengan nilai fitness minimal digantikan dengan gen/ kromosom dari individu dengan nilai fitness maksimal. Maka pada proses mutasi, gen yang berubah bersifat random.

Individu	fitness
00101001 =	0.653 induk
11010110 =	0.862 induk

Berikut persamaan dari proses mutasi:

$$m[n]_{+} = \text{GaussRand()} * (m_{[best]} - m_{[n]}) + \text{GaussRand()} * m_{[n]}$$

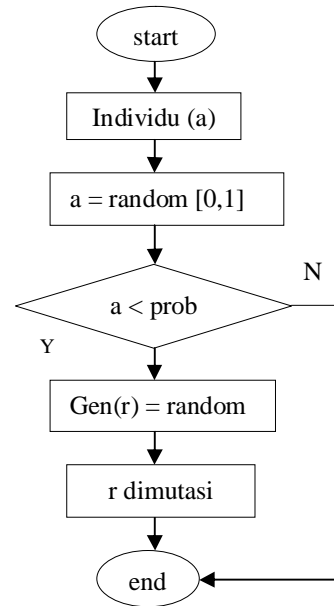
Keterangan:

= konstanta

$m[n]$ = parameter individu ke-n

$m_{[best]}$ = parameter individu terbaik

GaussRand() = fungsi gaussian



Gambar 8 Mutasi

Proses ini akan mengevaluasi setiap populasi dengan menghitung nilai *fitness* setiap kromosom dan mengevaluasinya sampai terpenuhi kriteria berhenti. Bila kriteria berhenti belum terpenuhi maka akan dibentuk lagi generasi baru dengan mengulangi langkah 2. Beberapa kriteria berhenti yang sering digunakan antara lain:

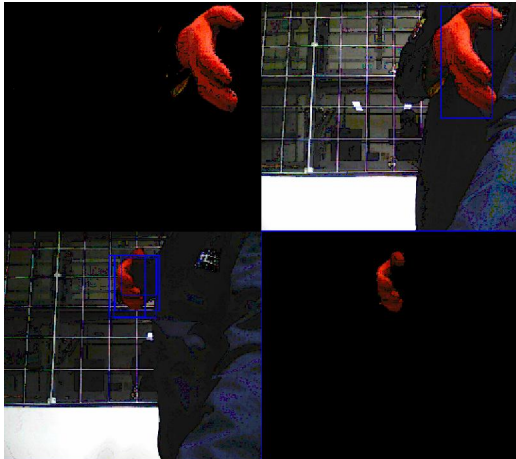
- Berhenti pada generasi tertentu.
- Berhenti setelah dalam beberapa generasi berturut-turut didapatkan nilai *fitness* tertinggi tidak berubah.
- Berhenti bila dalam n generasi berikut tidak didapatkan nilai *fitness* yang lebih tinggi.

Mutasi dapat dilakukan dari semua gen yang ada dengan probabilitas mutasi tertentu. Nilai probability antara 0 sampai 1.

Jika bilangan random yang dibangkitkan kurang dari probabilitas mutasi yang ditentukan maka ubah gen tersebut menjadi nilai kebalikan yang dalam hal ini, binary encoding 0 diubah 1, dan 1 diubah 0.

Bila mana probabilitas mutasi adalah (1/12) maka sebanyak 1 gen akan dimutasi dari kromosom yang terdiri dari 12 gen (bits). Pada

algoritma genetika yang sederhana, nilai probabilitas mutasi adalah tetap selama evolusi.



Gambar 9 Hasil Pengambilan

B. VIDEOCX TOOLS

Untuk memudahkan mengaplikasikan proses pengolahan gambar saat online atau saat kamera langsung terhubung ke PC, dipergunakan VideoOCX Tools untuk mengakses VideoOCX sebagai bagian dari fungsi ActiveX Control pada microsoft visual studio 6.0.

```
// Basic syntax yang digunakan
m_ocx.SetMode();
m_ocx.SetDriver (0);
m_ocx.Init ();
m_ocx.GetColorImageHandle();
m_ocx.SetPreview (TRUE);
m_ocx.Start ();
m_ocx.Close()
```

V. HASIL PENGUJIAN

Berikut ini merupakan proses pengujian software yang meliputi: kemampuan software dalam menentukan posisi tangan manusia dan membedakannya dengan objek lainnya.

Dari pengujian berulang-ulang diperoleh hasil yang cukup baik, dimana software dapat menentukan posisi tangan manusia dengan presisi dengan tingkat kesalahan sebesar 7%.

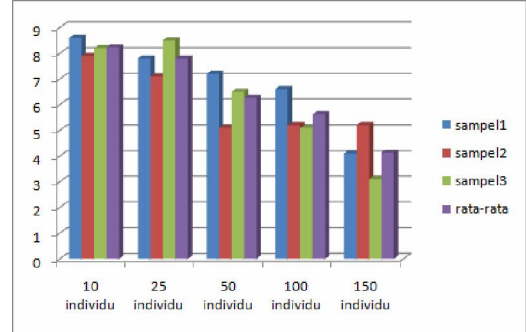
1). *Pengujian 1:* Dilakukan juga pengujian untuk mengetahui waktu optimasi dari sistem, pada tiap-tiap perubahan nilai jumlah populasi individu yang dibangkitkan.

Parameter pengujian yang digunakan:

- Probabilitas crossover = 0.7
- Probabilitas mutasi = 1.0
- Intensitas tetap

TABEL I
Waktu Pencapaian Tiap Populasi

waktu (detik)	10 individu	25 individu	50 individu	100 individu	150 individu
sampel1	8,6	7,8	7,2	6,6	4,1
sampel2	7,9	7,1	5,1	5,2	5,2
sampel3	8,2	8,5	6,3	5,1	3,1
rata-rata	8,23333333	7,8	6,26666667	5,63333333	4,13333333



Grafik 1 Hubungan antara Jumlah Populasi dengan Waktu Pencapaian

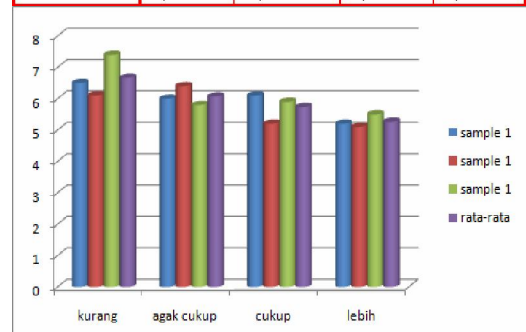
2). *Pengujian 2:* Pengujian kedua dilakukan untuk mengetahui pengaruh intensitas dalam proses pencapaian titik optimasi. Dimana pada saat yang sama, software mampu mendeteksi posisi tangan manusia.

Parameter pengujian yang digunakan:

- Jumlah populasi 50 Individu
- Probabilitas crossover = 0.7
- Probabilitas mutasi = 1.0

TABEL II
Waktu Pencapaian Pada Tiap Intensitas Kondisi Ruangan 1

waktu (detik)	kurang	agak cukup	cukup	lebih
sample 1	6,5	6	6,1	5,2
sample 1	6,1	6,4	5,2	5,1
sample 1	7,4	5,8	5,9	5,5
rata-rata	6,6666667	6,06666667	5,7333333	5,2666667

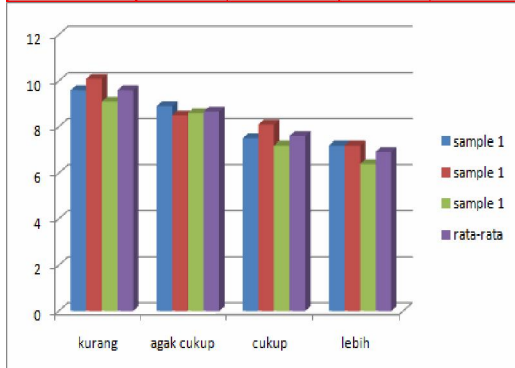


Grafik 2 Hubungan Antara Intensitas Cahaya Dengan Waktu Pencapaian 1

- Jumlah populasi 10 Individu
- Probabilitas crossover = 0.7
- Probabilitas mutasi = 1.0

TABEL III
Waktu Pencapaian Pada Tiap Intensitas Kondisi Ruang 2

waktu (detik)	kurang	agak cukup	cukup	lebih
sample 1	9,6	8,9	7,5	7,2
sample 1	10,1	8,5	8,1	7,2
sample 1	9,1	8,6	7,2	6,4
rata-rata	9,6	8,66666667	7,6	6,9333333



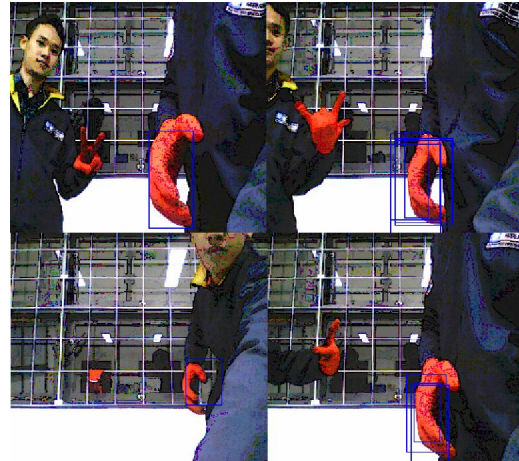
Grafik 3 Hubungan Antara Intensitas Cahaya Dengan Waktu Pencapaian 2

Dari hasil pengujian di atas kita dapat mengetahui bahwa jumlah populasi yang dibangkitkan berpengaruh pada peningkatan waktu pencapaian posisi tangan. Semakin banyak jumlah individu yang dibangkitkan, maka semakin cepat pencapaian posisi tangan.

VI. KESIMPULAN

Dari hasil analisa pengujian dapat diketahui adanya pengaruh intensitas cahaya dan juga perubahan jumlah populasi individu yang dibangkitkan terhadap waktu pencapaian sistem dalam menentukan posisi tangan.

- Banyaknya intensitas cahaya berbanding lurus dengan waktu pencapaian posisi tangan.
- Jumlah populasi individu yang dibangkitkan berbanding lurus dengan waktu pencapaian posisi tangan.



Gambar 10 Hasil Perbandingan Objek

DAFTAR PUSTAKA

- [1] **Gen Mitsuo, Runwei Cheng**, *Genetic Algorithms and Engineering optimization*, Wiley Computer Publishing 2000.
- [2] **Parker J.R.**, *Algorithm For Image Processing And Computer Vision*, Wiley Computer Publishing 1997
- [3] **Parker, J.R.** (1993). *Practical Computer Vision Using C*, John Wiley & Sons Inc, New York.
- [4] **Sams, A.T.**, *Teach Yourself Visual C++ in 21 Days*.
- [5] **Sigit Riyanto**, *Praktikum Pengolahan Citra Digital (Image Processing)*, PENS ITS 2002.
- [6] **Gonzalez, R.C, Woods, R.E.** (2002). *Digital Image Processing Second Edition*, Prentice Hall, New Jersey.
- [7] **VideoOCX.chm**, Microsoft HTML help control 6.0