

## ANIMASI POHON TERTIUP ANGIN MENGGUNAKAN METODE FRAKTAL

**Janoë Hendarto**

*Departemen Ilmu Komputer dan Elektronika  
Fakultas Matematika dan Ilmu Pengetahuan Alam  
Universitas Gadjah Mada  
email : [jhendarto@ugm.ac.id](mailto:jhendarto@ugm.ac.id)*

### **Abstraksi**

*Fraktal adalah bidang geometri yang mengupas dan dapat menjelaskan hal-hal yang alamiah (natural), yang mana semakin hari semakin berkembang seiring perkembangan teknologi komputer. Dengan menggunakan fraktal dapat dibuat gambar pohon tiruan, yaitu dengan menggunakan sistem fungsi iterasi ( SFI) dimana pemetaan kontraksinya digunakan transformasi affine. Animasi objek alam dengan menggunakan metode fraktal bisa dilakukan dengan cara mengubah-ubah parameter dari SFI nya untuk mendapatkan serangkaian frame gambarnya.*

*Pertama, akan dibahas tentang cara membuat gambar pohon dengan fraktal. Kemudian dibuat rangkaian frame gambar pohon tertiuip angin yang akan digunakan dalam animasinya. Dan akhirnya, dibuat program komputer yang mampu melakukan animasi dengan memperhatikan faktor kecepatan dan arah angin, kemudian diamati hasil animasinya dan dianalisis guna untuk menyempurnakan teknik animasinya. Untuk menggambar pohon tiruan fraktal ada beberapa faktor yang harus diperhatikan yaitu koordinat gambar, gambar awal, rumus transformasi termasuk nilai parameter alfa, pewarnaan dan jumlah iterasi. Sedangkan faktor arah angin pada proses animasi dapat dilakukan dengan mengatur nilai parameter alfa dan faktor kecepatan angin dapat diatur dari kecepatan animasinya. Program komputer yang dibuat telah berhasil melakukan animasi efek tiupan angin pada berbagai macam pohon tiruan dan terdapat fasilitas pengaturan parameter untuk menentukan arah dan kecepatan angin, walaupun masih belum sempurna yaitu gerakannya teratur dan masih kurang alami.*

**Kata kunci:** *Fraktal, Sistem Fungsi Iterasi, Animasi.*

### **Abstract**

*Fractals are a field of geometry that explores and can explain natural things, which are increasingly developing along with the development of computer technology. By using fractals, an image of a tree can be created by using the Iterated Function System (IFS) where the contraction mapping uses affine transformation.. Animation of natural objects using the fractal method can be done by changing the parameters of the IFS to get a series of image frames.*

*First, we will discuss how to create a tree image with fractals. Then a series of frames of a tree like blown by the wind images are created which will be used in the animation. And finally, the computer program is developed that is able to perform animation by considering the wind speed and direction factors, then the animation results are observed and analyzed in order to perfect the animation technique. To draw a fractal tree, there are several factors that must be considered, i.e image coordinates, initial images, transformation formulas including alpha parameter, coloring and the number of iterations. While the wind direction in the animation process can be done by setting the alpha parameter and the wind speed can be set from the animation speed. The computer program that was developed has succeeded in animating the effect of wind blowing on various kinds of trees and there are parameter setting facilities to determine the direction and speed of the wind, although it is still not perfect, that is, the movement is regular and not naturally yet.*

**Keywords:** *Fractal, Iterated Function System, Animation*

## **PENDAHULUAN**

Animasi adalah proses menggerakkan gambar di layar komputer. Di era multimedia ini, gambar bergerak sangat banyak digunakan pada program-program aplikasi. Teknik-teknik animasi telah berkembang dengan

pesat, terutama animasi objek alam yang aplikasinya cukup luas, antara lain untuk pembuatan film, iklan, simulasi dan lain-lainnya.

Fraktal adalah bidang geometri yang mengupas dan dapat menjelaskan hal-hal yang alamiah (natural), yang mana semakin hari semakin berkembang seiring perkembangan teknologi komputer dan keinginan manusia untuk mengungkap rahasia alam semesta. Dengan menggunakan fraktal dapat dirancang gambar-gambar objek alam seperti pohon, gunung, batuan, awan, permukaan bumi maupun planet lain dan lain-lainnya. Perancangan tersebut dilakukan dengan menggunakan suatu sistem yang disebut sistem fungsi iterasi (*iterated function system*) yang disingkat SFI, titik tetap/atraktor dari sistem fungsi iterasi inilah yang berupa gambar yang bisa berbentuk pohon atau objek alam lain yang diharapkan mirip dengan objek sebenarnya.

SFI yang digunakan dalam penggambaran pohon biasanya pemetaan kontraksinya menggunakan transformasi affine. Dengan mengubah-ubah nilai parameter pada transformasi affine dari suatu pohon, maka akan didapat perubahan gambar pohon yang mana perubahannya mirip seperti pohon tertiup angin. Kemiripan ini tidak perlu disangsikan karena fraktal jelas terbukti “dekat” dengan alam. Frame-frame gambar pohon tersebut jika dianimasikan akan didapat suatu animasi pohon tertiup angin yang diharapkan mirip dengan kejadian sebenarnya.

Pada penelitian ini dielaborasi dan dikaji (a). Bagaimana cara membuat gambar pohon yang mirip dengan pohon sebenarnya. (b). Bagaimana cara mengubah nilai parameter sedemikian hingga didapat rangkaian frame gambar pohon tertiup angin. c). Bagaimana program komputer untuk melakukan animasi pohon tertiup angin dengan memperhatikan faktor kecepatan dan arah angin.

Hasil tinjauan pustaka tentang animasi pohon dengan menggunakan metode fraktal dapat dilihat pada Tabel 1.

Peneliti	Topik Penelitian, Metode yang digunakan	Kekurangan/kelemahan
Cao, W. dan Zeng, L., 2009	Melakukan pemodelan pohon secara interaktif, yaitu membuat sistem yang pemodelannya menggunakan dekomposisi bagian cabang yang menggunakan metode fraktal dan bentuk keseluruhan pohon dengan menambahkan daun yang dibuat dengan algoritma global. Pengguna dapat mengubah nilai parameter untuk mengontrol struktur pohon.	Gambar pohon fraktalnya cukup bagus, akan tetapi belum membahas tentang animasinya dan juga tidak membahas perancangan rangkaian gambar pohon tertiup angin.
Darmanto, T. dkk., 2013	Mensimulasikan animasi metamorfosis dari pohon fraktal dengan efek anyaman, menggunakan SFI berupa transformasi affine dan interpolasi.	Belum menyinggung tentang animasi pohon dengan efek tiupan angin.
Hadzieva, E. dan Petkoski, J., 2015	Menyajikan aplikasi interaktif yang memungkinkan pembangkitan dan pemodelan gambar fraktal berupa pohon dua dimensi menggunakan SFI, memodifikasi parameter pada SFI dan membuat animasi dengan mengubah satu atau beberapa koefisien dari SFI. Pemodelan dapat dilakukan dengan menentukan tiga titik yang tidak segaris.	Animasi pohonnya bukan animasi pohon dengan efek tiupan angin.
Xue, F. dan Xu, F., 2018	Struktur spasial hutan dibangun dengan menghitung hamburan elektromagnetik yang koheren dimana pohon-pohon dihasilkan berdasarkan algoritma yang berbasis aturan pertumbuhan. Batang, dahan, dan dedaunan di pohon-pohon itu disederhanakan sebagai sekelompok silinder dan disk, pendekatan silinder digunakan untuk menghitung hamburan, koefisien hamburan dari satu pohon dan hutan keduanya dihitung, yang biasa dilakukan untuk mensimulasikan gambar PolinSAR,	Dapat menghasilkan banyak gambar pohon fraktal, akan tetapi belum menyinggung tentang animasi pohonnya.

Kumar, A. dan Pharwaha, A.,P.,S. , 2019	Mengusulkan penggunaan pohon fraktal Pythagoras yang dimodifikasi dalam pemodelan antena patch dimana pohon fraktal yang digunakan hingga iterasi ketiga. Selanjutnya antena yang diusulkan adalah dibuat kompatibel dengan sirkuit terpadu gelombang mikro monolitik, kemudian disimulasikan dengan hasil cukup mirip dan desainnya cocok untuk eksplorasi bumi pada layanan satelit.	Pohon fraktal digunakan untuk pemodelan antena, belum menyinggung tentang animasi pohonnya.
Tao, X. Dkk., 2021	Mengeksplorasi peran berbagai parameter dari transformasi affine dalam SFI untuk mendapatkan metode yang menghasilkan gambar fraktal yang bebas dan dapat dikontrol. Kemudian menganalisis pengaruh generasi setiap transformasi affine pada tampilan gambar fraktal dan mensimulasikan berbagai bentuk pohon fraktal.	Bentuk pohon fraktal yang dibangun dapat dikontrol dengan menggunakan parameter dari transformasi affine, akan tetapi belum membahas animasi pohonnya.
Wang, W. dkk, 2021	Membandingkan 2 model simulasi pohon listrik yang berdasarkan teori fraktal yaitu model NPW dan model WZ. Kemudian pengaruh parameter kedua model terhadap hasil simulasi dibandingkan, hasilnya menunjukkan bahwa dimensi fraktal pohon listrik lebih kecil pada material dengan kekuatan medan tembus yang tinggi.	Pohon fraktal digunakan untuk simulasi listrik, tidak membahas animasi pohon, hanya mengamati dimensi dari pohon fraktal.

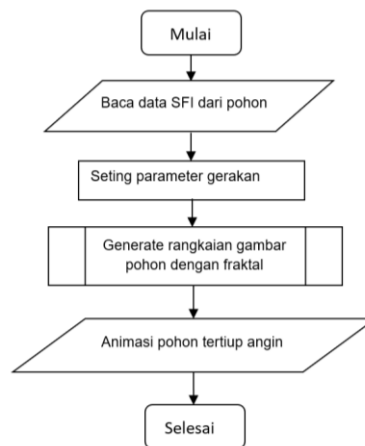
**Tabel 1. Tabel hasil tinjauan pustaka**

Dari penelitian-penelitian di atas, sebagian besar membahas pembuatan gambar pohon tiruan menggunakan metode fraktal dengan menggunakan transformasi affine sebagai sistem fungsi iterasi (SFI) dan memanfaatkan parameter-parameter dari transformasi affine untuk memodifikasi pohonnya, akan tetapi belum banyak yang mencari parameter yang dapat menghasilkan rangkaian frame gambar pohon yang dapat digunakan dalam animasi pohon tertiuip angin. Pada penelitian ini dibahas bagaimana membuat rangkaian gambar pohon tiruan yang dapat digunakan dalam animasi pohon dengan efek seperti pohon tertiuip angin dengan arah dan kecepatan anginnya dapat ditentukan oleh pengguna, diharapkan animasinya cukup realistis dan dapat digunakan untuk keperluan yang lebih luas.

## METODE PENELITIAN

Langkah-langkah yang dilakukan dalam penelitian ini adalah:

1. Studi pustaka, mulai dari referensi buku hingga mempelajari paper-paper mengenai fraktal secara umum dan penggambaran pohon dengan metode fraktal.
2. Analisis permasalahan dan analisis terhadap algoritme penggambaran pohon dengan menggunakan metode fraktal, terutama permasalahan pembuatan rangkaian gambar pohon tertiuip angin.
3. Perancangan aplikasi animasi pohon tertiuip angin menggunakan metode fraktal yaitu pertama dicari SFI dengan transformasi affine yang atraktornya berupa gambar pohon, kemudian dicari parameter yang berpengaruh pada bentuk gambar pohonnya yang seolah olah seperti tertiuip angin dan akhirnya dibuat program komputer untuk membuat gambar pohon dan yang mampu melakukan animasi dengan memperhatikan faktor kecepatan dan arah tiupan angin. Secara garis besar langkah-langkah animasi pohon tertiuip angin menggunakan metode fraktal dapat dinyatakan dengan diagram alir pada Gambar 1.
4. Mengimplementasikan aplikasi dalam bentuk program komputer, dengan menggunakan bahasa pemrograman delphi.
5. Menguji kebenaran program dan menganalisis hasil program meliputi efisiensi algoritme dan kualitas animasi dari aplikasi animasi pohon tertiuip angin menggunakan metode fraktal.



Gambar 1. Diagram alir animasi pohon tertiuip angin menggunakan metode fraktal

## HASIL DAN PEMBAHASAN

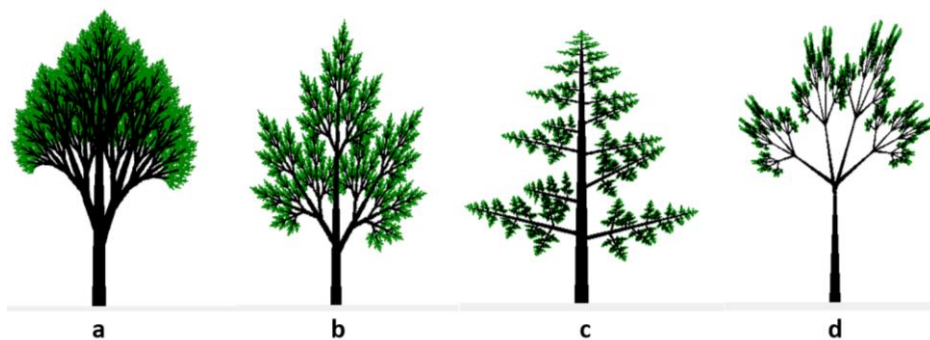
Penelitian ini dilakukan pertama dibahas tentang bagaimana cara membuat gambar pohon tiruan dengan metode fraktal, kemudian bagaimana membangun rangkaian gambar pohon yang akan digunakan dalam animasi pohon tertiuip angin dan akhirnya bagaimana melakukan animasi dengan memperhatikan faktor arah dan kecepatan angin.

### A. Pembuatan gambar pohon dengan metode fraktal.

Pohon adalah termasuk objek alam, sehingga untuk menggambarnya lebih cocok digunakan pendekatan geometri fraktal. Pada penelitian ini gambar pohon yang dibahas adalah gambar pohon 2 dimensi. Gambar pohon fraktal merupakan atraktor dari suatu SFI dimana pemetaan kontraksinya digunakan transformasi affine sebagai berikut :

$$W_i : \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_i & b_i \\ c_i & d_i \end{bmatrix} * \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \end{bmatrix} \quad \text{untuk } i=1,2, \dots, N$$

dengan N adalah banyaknya transformasi affine, pada penelitian ini digunakan beberapa nilai N yaitu 4, 5 dan 6. Nilai dari  $a_i$ ,  $b_i$ ,  $c_i$ ,  $d_i$ ,  $e_i$ ,  $f_i$  dan N bergantung pada bentuk pohon yang diinginkan. Beberapa gambar pohon tiruan hasil penggambaran dengan metode fraktal dapat dilihat pada Gambar 2.



Gambar 2. Beberapa gambar pohon tiruan

Gambar pohon tiruan **a** pada Gambar 2 dibentuk dengan memberi nilai  $N = 4$  dan nilai dari parameter transformasi affinenya adalah sebagai berikut :

$$W_1 : a_1 = 1; b_1 = 0; c_1 = 0; d_1 = 1; e_1 = 0; f_1 = 0;$$

$$W_2 : a_2 = 0,6; b_2 = 0; c_2 = 0; d_2 = 0,6; e_2 = 0; f_2 = 3,3;$$

$$W_3 : a_3 = 0,6495; b_3 = 0,225; c_3 = -0,225; d_3 = 0,6495; e_3 = 0; f_3 = 1,6;$$

$$W_4 : a_4 = 0,5734; b_4 = -0,2294; c_4 = 0,2294; d_4 = 0,5734; e_4 = 0; f_4 = 2,0;$$

Sedangkan gambar pohon tiruan **d** pada Gambar 2 dibentuk dengan memberi nilai  $N = 5$  dan nilai dari parameter transformasi affinenya adalah sebagai berikut :

$$\begin{aligned} W_1: & a_1 = 1; b_1 = 0; c_1 = 0; d_1 = 1; e_1 = 0; f_1 = 0; \\ W_2: & a_2 = 0,15; b_2 = -0,433; c_2 = 0,2598; d_2 = 0,25; e_2 = 0; f_2 = 2,5; \\ W_3: & a_3 = 0,2954; b_3 = -0,1042; c_3 = 0,0521; d_3 = 0,5908; e_3 = 0; f_3 = 2,3; \\ W_4: & a_4 = 0,2819; b_4 = -0,2223; c_4 = 0,1026; d_4 = 0,6108; e_4 = 0; f_4 = 2,4; \\ W_5: & a_5 = 0,1721; b_5 = -0,4096; c_5 = 0,2457; d_5 = 0,2868; e_5 = 0; f_5 = 2,5; \end{aligned}$$

Konsep pembuatan gambar pohon tiruan dimulai dengan membuat gambar awal sebuah batang, kemudian ditransformasikan menggunakan  $N$  transformasi affine untuk menghasilkan cabang. Jika hal ini dikerjakan berulang-ulang (proses iterasi), maka akan dihasilkan sebuah gambar pohon tiruan yang mempunyai batang, cabang, ranting, sampai daun. Ada beberapa hal yang harus diperhatikan termasuk penentuan beberapa parameter terlebih dahulu agar menghasilkan pohon tiruan seperti yang diharapkan.

Pohon yang akan digambar didefinisikan pada sistem koordinat kartesius 2D, sedangkan gambar yang tampak dilayar komputer mengikuti sistem koordinat layar. Hal pertama yang harus ditentukan adalah *window*, yaitu daerah segiempat pada sistem koordinat kartesius dimana pohon tiruan didefinisikan. Agar tampilan di layar komputer (*viewport*) sesuai dengan window maka diperlukan transformasi dari window ke viewport. Gambar pohon yang dihasilkan akan ditampilkan di layar monitor, sehingga harus ditentukan window dan posisi viewport di mana pohon tersebut akan ditampilkan. Berikut contoh pemberian nilai parameter-parameter dari viewport dan window :

$$\begin{aligned} \text{Viewport} : & \quad x_{vmin} = 0; \quad x_{vmax} = 450; \quad y_{vmin} = 0; \quad y_{vmax} = 500; \\ \text{Window} : & \quad x_{wmin} = -4.5; \quad x_{wmax} = 4.5; \quad y_{wmin} = 0; \quad y_{wmax} = 10; \end{aligned}$$

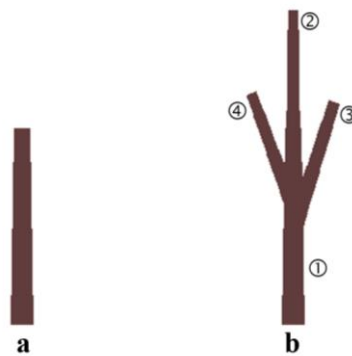
Semua proses transformasi affine dilakukan pada window, sedangkan viewport digunakan untuk menampilkan pohon pada layar monitor. Transformasi window ke viewport dinyatakan dengan sebuah prosedur sebagai berikut :

```
procedure Tform1.WV(xw,yw:real;var xv,yv:integer);
begin
  xv:=xvmin+round((xw-xwmin)*(xvmax-xvmin)/(xwmax-xwmin));
  yv:=yvmax-round((yw-ywmin)*(yvmax-yvmin)/(ywmax-ywmin));
end;
```

Sebuah gambar pohon yang dihasilkan dengan berbagai macam bentuknya berawal dari sebuah batang sebagai gambar awal. Gambar awal bentuknya dapat bermacam-macam, bisa sebuah garis, polygon, lingkaran dan lain lain. Bentuk batang ini menentukan bentuk gambar pohon yang dihasilkan, karena pada dasarnya pembuatan pohon adalah proses transformasi dari batang yang dilakukan secara berulang-ulang.

Transformasi affine merupakan komponen utama dalam pembuatan gambar pohon dengan metode fraktal. Transformasi inilah yang akan menentukan bentuk dari pohon yang dihasilkan. Transformasi affine dari sebuah gambar pohon bisa terdiri dari transformasi geometri dasar seperti translasi, skala dan rotasi, dengan menggunakan beberapa parameter, seperti parameter sudut alfa (dalam derajat) digunakan untuk membuat kemiringan pohon. Untuk menentukan transformasi yang sesuai memang tidak mudah, memerlukan "*sense of art*" dan geometri. Percobaan beberapa kali sebaiknya dilakukan untuk menghasilkan sebuah gambar pohon yang mendekati harapan. Gambar 3 adalah contoh bentuk gambar awal (**a**) dan gambar hasil (**b**) setelah dikenai 4 transformasi affine berikut :

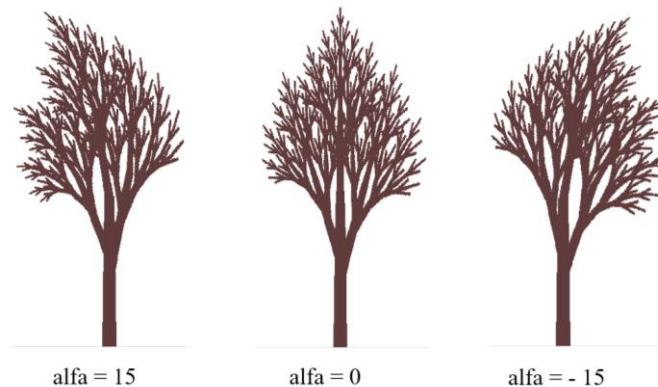
$$\begin{aligned} W_1: & a_1 = 1; b_1 = 0; c_1 = 0; d_1 = 1; e_1 = 0; f_1 = 0; \\ W_2: & a_2 = 0,6\cos(\alpha); b_2 = -0,5\sin(\alpha); c_2 = 0,2\sin(\alpha); d_2 = 0,6\cos(\alpha); e_2 = 0; f_2 = 3,3; \\ W_3: & a_3 = 0,75\cos(\alpha/2,5-30); b_3 = -0,45\sin(\alpha/2,5-30); c_3 = 0,45\sin(\alpha/2,5-30); \\ & d_3 = 0,75\cos(\alpha/2,5-30); e_3 = 0; f_3 = 1,6; \\ W_4: & a_4 = 0,7\cos(\alpha/2+35); b_4 = -0,4\sin(\alpha/2+35); c_4 = 0,4\sin(\alpha/2+35); \\ & d_4 = 0,7\cos(\alpha/2+35); e_4 = 0; f_4 = 2,0; \end{aligned}$$



**Gambar 3. Gambar awal (a) dan gambar hasil iterasi pertama (b)**

Transformasi  $W_1$  menghasilkan bentuk dan posisi yang mirip dengan gambar awal. Perlu diperhatikan bahwa bentuk yang dihasilkan bukan merupakan gabungan antara gambar awal dengan gambar hasil transformasi. Setelah dilakukan transformasi, bentuk awal akan hilang dan digantikan dengan bentuk hasil untuk keperluan iterasi berikutnya. Untuk itu diperlukan sebuah transformasi untuk mempertahankan keberadaan bentuk awal pada bentuk hasil, itulah peran dari transformasi  $W_1$ . Transformasi  $W_2$  menghasilkan cabang yang menjulang ke atas. Transformasi  $W_3$  menghasilkan cabang yang miring ke kanan, sedangkan transformasi  $W_4$  menghasilkan cabang yang miring ke kiri dengan pangkal cabang berada agak ke atas bergantung dari nilai parameter  $f$ .

Pada transformasi affine di atas ada parameter sudut alfa yang dapat diubah nilainya untuk menghasilkan bentuk kemiringan pohon yang berbeda yang dapat dimanfaatkan untuk mengilustrasikan efek tiupan angin pada pohon. Pada Gambar 4 terlihat 3 gambar pohon hasil iterasi ke-5, masing-masing dengan nilai alfa = 15, 0 dan -15.



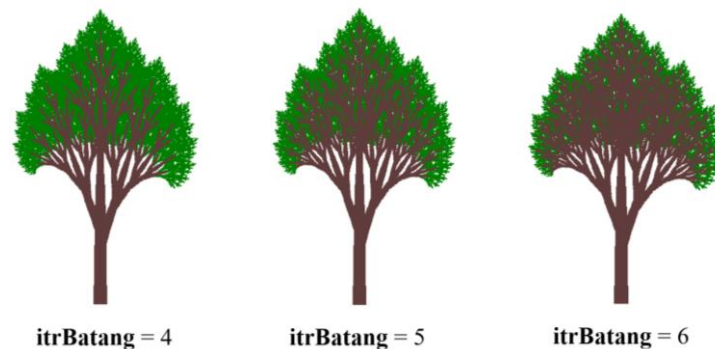
**Gambar 4. Gambar beberapa pohon dengan nilai alfa berbeda**

Pohon asli mempunyai banyak macam warna, untuk dapat menghasilkan gambar pohon yang realistis minimal diperlukan 3 komponen warna yaitu warna dasar, warna batang dan warna daun. Warna dasar merupakan warna latar belakang, artinya warna dari titik (pixel) dalam viewport yang tidak termasuk pohon. Untuk membuat pohon terlihat menarik 2 warna lainnya yaitu warna batang adalah coklat dan warna daun adalah hijau. Dalam proses transformasi yang dikenai adalah pixel-pixel yang termasuk pohon, sedangkan pixel lainnya yaitu warna latar belakang diabaikan. Sementara pengecekan tetap dilakukan terhadap semua pixel dalam viewport.

Berikut contoh kode program yang mengatur komponen warna ke dalam konstanta :

```
WarnaDaun = clGreen; //warna daun adalah hijau
WarnaBatang = $3c3c60; //warna batang adalah coklat kemerahan
WarnaDasar = clWhite; //warna latar belakang adalah putih
```

Parameter terakhir berkaitan dengan pembuatan pohon dengan metode fraktal adalah banyaknya iterasi. Parameter ini menentukan tingkat kelebatan dari pohon. Semakin besar banyaknya iterasi, maka pohon yang dihasilkan semakin lebat. Namun sampai pada batas tertentu, peningkatan banyaknya iterasi tidak akan memperlihatkan perubahan yang signifikan. Untuk mendapatkan gambar batang (coklat) dan daun (hijau) perlu ditentukan sejauh mana hasil iterasi dianggap sebagai batang dan hasil iterasi berikutnya dianggap sebagai daun, sehingga ada dua parameter iterasi, yaitu iterasi pohon (**itrPohon**) yaitu total banyaknya iterasi dan iterasi batang (**itrBatang**) yaitu banyaknya iterasi untuk mendapatkan batang saja, sehingga iterasi daunnya adalah **itrPohon** dikurangi **itrBatang**. Gambar pohon dengan **itrPohon** = 8 dengan **itrBatang** = 4, 5 dan 6 dapat dilihat pada gambar 5.

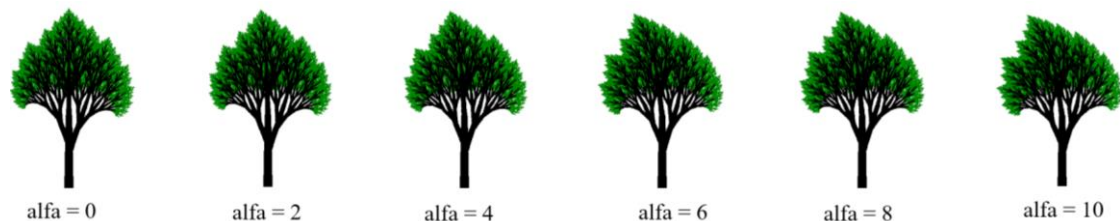


Gambar 5. Gambar beberapa pohon dengan nilai **itrBatang** berbeda

## B. Animasi Pohon Tertiuip Angin menggunakan Metode Fraktal.

Setelah gambar pohon tiruan berhasil dibuat dengan menggunakan metode fraktal, langkah berikutnya adalah membangun animasi pohon tertiuip angin. Teknik animasi yang digunakan dalam penelitian ini adalah metode klasik, yaitu *frame by frame*. Untuk itu ada beberapa langkah yang harus dilakukan agar menghasilkan animasi sebuah pohon tertiuip angin.

Langkah pertama adalah membuat frame-frame gambar yang akan digunakan pada animasi. Frame gambar yang dibuat disimpan pada memori agar kecepatannya aksesnya cepat sehingga diharapkan menghasilkan animasi yang baik. Frame-frame gambar pohon fraktal dibuat berdasarkan nilai variabel alfa pada transformasi affine, yaitu untuk nilai alfa = awal sampai dengan nilai alfa = akhir dengan interval atau step tertentu. Sebagai gambaran dapat dilihat pada Gambar 6 yaitu untuk nilai alfa = 0 sampai dengan alfa = 10 dengan step = 2. Semakin kecil nilai step semakin banyak frame gambarnya sehingga gerakan animasinya diharapkan semakin halus.



Gambar 6. Rangkaian frame gambar pohon untuk beberapa nilai alfa

Proses animasi secara umum adalah menampilkan rangkaian frame-frame gambar tersebut dengan mengatur posisi gambar dan waktu jeda (kecepatan animasi). Pada kasus animasi pohon tertiuip angin ini, letak posisi gambar tidak berubah karena posisi pohon tidak berpindah hanya bergoyang saja. Sehingga faktor yang perlu diperhatikan adalah kecepatan animasi yang nantinya berhubungan dengan kecepatan angin, sedangkan untuk arah angin berhubungan dengan pengaturan frame-frame gambarnya yaitu berdasarkan nilai alfa.

Pada penelitian ini frame gambar pohon yang dibuat adalah sebanyak 31 frame gambar pohon yaitu untuk nilai alfa = -15 sampai dengan alfa = 15 dengan step = 1. Angka 15 diambil berdasarkan uji hasil program dan nilai maksimum alfa sebenarnya bergantung dari jenis atau kekerasan pohonnya. Untuk pohon besar dan keras efek tiupan angin tidak sampai mencapai 15 derajat dan yang bergerak hanya sekitar daun dan tidak sebagian besar batang.

Proses animasi diawali dengan menyiapkan variabel bertipe *bitmap* untuk menyimpan frame gambar, kode programnya sebagai berikut :

```
procedure TForm1.FormCreate(Sender: TObject);
var i:shortint;
begin
  bm:=tbitmap.create;
  bm.Width:=xvmax-xvmin;
  bm.Height:=yvmax-yvmin;
  for i:= -15 to 15 do
  begin
    bmList[i]:=tbitmap.create;
    bmList[i].Width:=xvmax-xvmin;
    bmList[i].Height:=yvmax-yvmin;
  end;
end;
```

Varibel *bm* digunakan untuk menyimpan gambar pada proses pembuatan frame gambarnya, kemudian gambar yang disimpan di *bm* dicopy ke variabel array *bmList*, proses ini dapat dilihat pada kode program berikut :

```
procedure TForm1.Button3Click(Sender: TObject);
var i:shortint;
begin
  for i:= -15 to 15 do begin
    BuatGambarPohon(bm);
    bmList[i].Canvas.CopyRect(Rect(0,0,bm.width,bm.height),
      bm.Canvas, Rect(0,0,bm.width,bm.height));
  end;
end;
```

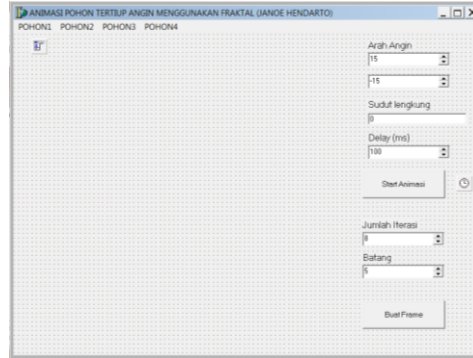
Setelah frame-frame gambar siap maka selanjutnya melakukan animasi dengan mengatur frame gambar mana saja yang digunakan dan bagaimana urutannya serta mengatur kecepatan animasinya. Proses animasinya bisa menggunakan fasilitas *timer* pada borland delphi, kode programnya sebagai berikut :

```
procedure TForm1.Timer1Timer(Sender: TObject);
var i:shortint;
begin
  Timer1.Tag:= 1; {step = 1 }
  Timer1.Interval:= waktuDelay;
  i:= StrToInt(Edit1.Text);
  Canvas.CopyRect(RECT(0,0,bmList[i].width,bmList[i].height),
    bmList[i].Canvas,RECT(0,0,bmList[i].width,bmList[i].height));
  i:= i+Timer1.Tag;
  if (i > 15)or(i < -15) then Timer1.Tag:= -1 * Timer1.Tag
  else Edit1.Text:= IntToStr(i);
end;
```

Untuk mengatur arah angin, yaitu arah angin ke kiri dan kekanan dilakukan dengan memilih frame gambarnya. Untuk arah angin ke kiri dipilih frame gambar dengan nilai alfa = 0 sampai dengan alfa = 15, sedangkan untuk arah angin ke kanan dipilih frame gambar dengan nilai alfa = -15 sampai dengan alfa = 0. Untuk arah angin tidak menentu ( ke kiri dan ke kanan ) dipilih frame gambar dengan nilai alfa = -15 sampai dengan alfa = 15. Sedangkan untuk mengatur kecepatan angin, dapat dilakukan dengan cara mengatur nilai waktuDelay yang nilainya dalam milidetik. Untuk waktuDelay = 0 berarti kecepatan angin maksimal, sedangkan untuk waktuDelay = 100 kecepatan angin sedang atau pelan.

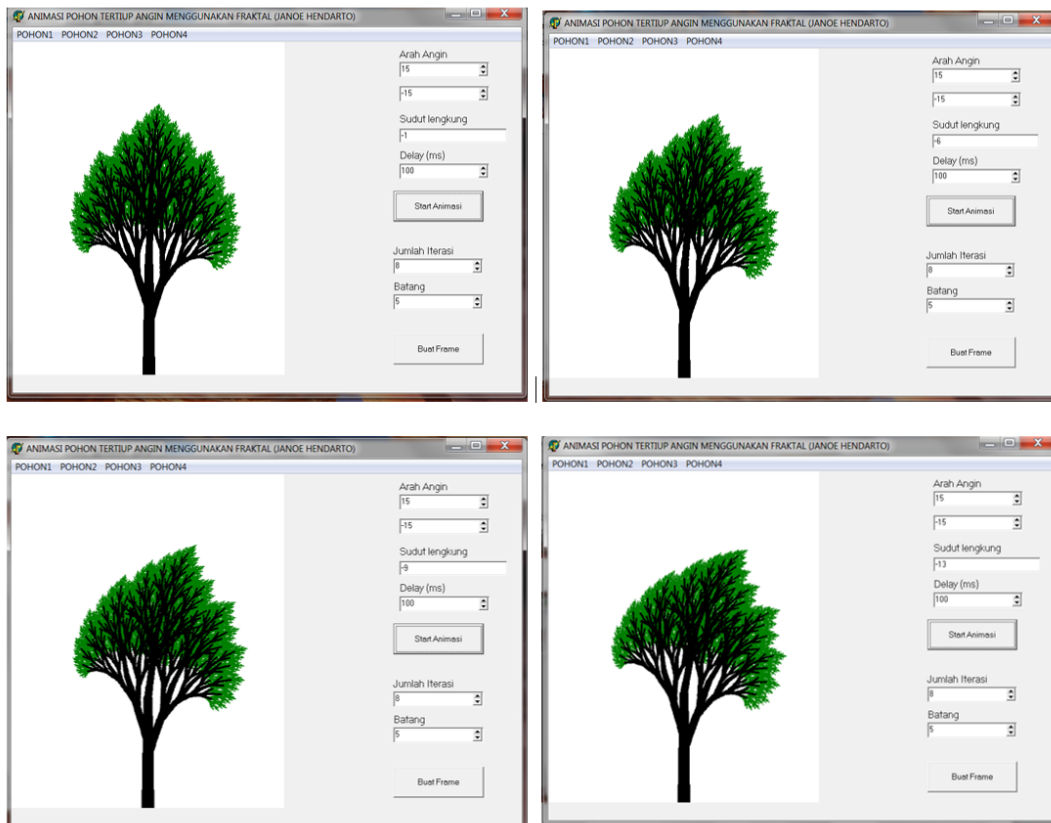


Program komputer untuk melakukan animasi pohon tertiuip angin menggunakan metode fraktal telah dibuat secara interaktif dengan pilihan 4 macam pohon tiruan, antarmuka programnya dapat dilihat pada Gambar 7.



Gambar 7. Antarmuka program animasi pohon tertiuip angin

Program dapat membuat frame gambar yang akan digunakan dalam animasi berdasarkan jumlah iterasi pohon dan iterasi batangnya, kemudian pengguna dapat menentukan arah angin dan kecepatan atau waktu jedaanya, beberapa gambar tangkapan layar dari hasil program animasi pohon tertiuip angin menggunakan metode fraktal dapat dilihat pada Gambar 8.



Gambar 8. Hasil tangkapan layar proses animasi pohon tertiuip angin

## KESIMPULAN

Dari penelitian yang dilakukan dapat ditarik beberapa kesimpulan sebagai berikut :

1. Metode fraktal cukup memadai untuk digunakan dalam pembuatan gambar pohon, walaupun untuk mendapatkan gambar pohon yang realistik masih ada beberapa kesulitan.
2. Untuk menggambar pohon fraktal ada beberapa faktor yang harus diperhatikan yaitu bentuk gambar awal, rumus transformasi affine, pewarnaan dan jumlah iterasi.
3. Rangkaian frame gambar untuk animasi pohon tertiuip angin didapat dari perubahan nilai parameter alfa pada rumus transformasi affinenya.
4. Faktor arah angin pada animasi pohon tertiuip angin dapat dikontrol dengan mengatur nilai parameter alfa dan faktor kecepatan angin dapat diatur dari waktu jedah pada proses animasinya.

## DAFTAR PUSTAKA

- [1]. Cao, W. dan Zeng, L., 2009, "Natural Tree Modeling Based on Fractal", 978-0-7695-3735-1/09 \$25.00 ©2009 IEEE.
- [2]. Darmanto, T. dkk., 2013, " Weaving Effects in Metamorphic Animation of Tree-like Fractal based on a Family of Multi-transitional Iterated Function System Code", 978-1-4799-1078-6/13/\$31.00 ©2013 IEEE.
- [3]. Hadzieva, E. dan Petkoski, J., 2015, "An Interactive Application for Modeling Two-Dimensional IFS Fractals", 978-1-4799-8673-6/15 \$31.00 ©2015 IEEE.
- [4]. Kumar, A. dan Pharwaha, A.,P.,S. , 2019, "Design and Analysis of Modified Pythagorean Fractal Tree Patch Antenna for Multiband Applications", 978-1-7281-2530-5/19/\$31.00©2019 IEEE.
- [5]. Tao, X. Dkk., 2021, "Algorithm of Controllable Fractal Image Based on IFS Code", 978-1-7281-9004-4/21/\$31.00 ©2021 IEEE.
- [6]. Wang, W. Dkk, 2021, "Numerical Simulation of Electric Trees Based on Fractal Theory", 4th International Electrical and Energy Conference (CIEEC) 978-1-7281-7149-4/21/\$31.00 ©2021 IEEE.
- [7]. Xue, F. dan Xu, F., 2018 , "Coherent Scattering and PolinSAR Imaging Simulation of Fractal Trees", 2018 China International SAR Symposium (CISS) IEEE.

## Biodata Penulis

**Drs. Janoe Hendarto M.Kom.**, memperoleh gelar Sarjana Matematika (Drs.), Program Studi Matematika FMIPA UGM, lulus tahun 1986. Tahun 1992 memperoleh gelar Magister Komputer (M.Kom) dari Program Studi Ilmu Komputer Program Pascasarjana UI. Saat ini sebagai Staf Pengajar program studi S1 Ilmu Komputer Departemen Ilmu Komputer dan Elektronika FMIPA Universitas Gadjah Mada.