

Xiong, Fanlun, Xing Zhao, and Youhua Zhang. 2006. Section 6.6 3-D Animation and Virtual Reality, pp. 425-434 of Chapter 6 Management and Decision Support Systems, in CIGR Handbook of Agricultural Engineering Volume VI Information Technology. Edited by CIGR-The International Commission of Agricultural Engineering; Volume Editor, Axel Munack. St. Joseph, Michigan, USA: ASABE. Copyright American Society of Agricultural Engineers.

Çevirmenler: Sefa TARHAN, Mehmet Metin ÖZGÜVEN ve Abdullah BEYAZ
Çeviri Editörleri: Sefa TARHAN ve Mehmet Metin ÖZGÜVEN

6.6 3D Animasyon ve Sanal Gerçeklik

Yazarlar: F. Xiong, X. Zhao ve Y. Zhang

Çevirmenler: Sefa TARHAN, Mehmet Metin ÖZGÜVEN ve Abdullah BEYAZ

Özet: Bilgisayar grafikleri, bilgi teknolojilerinde aktif bir yer almaya başlamıştır. Bilgisayar grafiklerinin birçok ileri teknolojiye, özellikle 3D animasyon ve sanal gerçeklik teknolojisine adaptasyonu ile parlak bir gelecek vadetmektedir. Bu bölümde, tarımsal bilişim alanında önemli bir konu olan sanal bitkiler konusuna temel düzeyde değinilecektir. Öncelikle 3D animasyon konusunda; popüler modelleme teknolojileri, görselleştirme, hareket kontrol gibi bazı temel kavramlar anlatılacak, sonrasında kısaca sanal gerçeklik konusu tanımlanacaktır. Son olarak, tarımda 3D animasyon ve sanal gerçeklik alanında özellikle sanal bitkilerde başarı, sanal bitkiler için modelleme metodları ve modelleme araçları konusunda bazı uygulamalar anlatılacaktır.

Anahtar Kelimeler: 3D Animasyon, Sanal gerçeklik, Sanal bitki.

6.6.1 Giriş

Bilgisayar grafikleri, bilgi teknolojilerinde aktif olarak yer almaya başlamıştır. Bilgisayar grafikleri; grafiklerin veya görüntülerin gösterilmesi, üretilmesi ve işlenmesi için bilgisayarların kullanıldığı bir uygulama teknolojisidir. Bilgisayar grafikleri; günümüzde endüstri, askeriye, bilimsel araştırmalar, ticaret, eğitim, sanat, eğlence, filmler, TV ve aile hayatı gibi birçok sosyal alanda kullanılmaktadır. Multimedya, görüntü işleme, bilgisayarlı görme, yapay zeka, ağlar gibi ileri teknolojilerin adaptasyonu ile grafik cihazlarının uyumluluğu ve düşük fiyatları, özgün yazılım araçlarının ortaya çıkmasıyla bilgisayar grafikleri daha popüler hale gelmektedir. Sanal gerçeklik ve 3D animasyon parlak bir geleceğe sahiptir.

6.6.2 3D Animasyon

Bilgisayar animasyonları, hareket grafikleri ve görüntü temelli animasyon işlemleri içeren ilginç bir teknolojidir. 2D animasyon ve 3D animasyon olmak üzere iki şekli vardır. 2D animasyon aynı zamanda *bilgisayar destekli animasyon* olarak adlandırılır ve geleneksel el ile yapılan animasyonun tekrarlı iş yükünün (örneğin; çizim, görselleştirme, kopyalama, ölçülendirme, hareket ettirme, çevirme vb.) yerini

almıştır. Görüntü çerçevelerine dayanan 2D teknolojisinde görüntü kareleri arasında otomatik geçiş sağlayarak kullanıcıya gerçekçi ve eş zamanlı bir his vermek zordur. 3D teknolojisi ise geometrik modeller, köşe birleşimleri, 3D derinlik bilgisine sahip çizgi ve yüzeyleri içermektedir. Yazılımlar, multimedya araçları ve grafik teorisi gibi görüntü ve grafik araçlarının gelişmesi ile 3D teknolojisi heyecan verici uygulamalar getirmektedir.

3D animasyonun temel teknolojileri; modelleme, görselleştirme ve hareket kontrolüdür. 3D animasyon için birçok yazılım paketi bulunmaktadır.

Modelleme Teknolojisi

Modelleme, farklı grafiksel nesnelerin temsil edilmesi için uygun modellerin nasıl uygulanacağı anlamına gelmektedir. Bir nesnenin tanımlanması topolojik ve geometrik bilgileri gerektirmektedir. Geometrik bilgi; ölçü, şekil, pozisyon gibi geometrik veriler içerir. Topolojik bilgi ise kenarlar, köşeler ve düzlemlerin birbirleriyle nasıl bağlandığını yansıtan çerçeve bilgileri içerir. 3D modelleme üç farklı temel modelleme teknolojisini içermektedir; bunlar çokgen temelli modelleme, eğri yüzey temelli modelleme, parametre temelli modellemedir. Çokgen temelli modelleme en basit modeldir. Çokgenlerden oluşan bir nesne içerir. Eğri yüzey temelli modelleme; düzgün yüzeyli olmayan bir nesnenin biçimini tanımlamak için silindir, koni, küre, elipsoid gibi kuadrik yüzeyler ve kontrol edilebilir noktalardan oluşan mekansal ızgaralar kullanır. Parametre temelli modellemede ise bir nesnenin geometrik ve yapısal parametreleri tanımlanır.

Doğal nesneler düzenli ve düzensiz olmayan nesneler olarak sınıflandırılabilir. Düzenli nesne için öklid geometrisi onun modelini ifade etmek için kullanılabilir ve bu geometri modelleme olarak adlandırılır. Bununla birlikte öklid geometrisi düzensiz nesnelerin tanımlanmasında kullanmak zordur. Düzensiz nesneler için fraktal boyut, biçim temel prensipleri ve parçacık sistemleri gibi prosedürel modelleme metodları uyarlanmaktadır.

Fraktal geometri yaklaşımı; boyutu olmayan noktalar, bir boyutlu çizgiler, iki boyutlu düzlemler, üç boyutlu hacimlere sahip nesneler kullanan klasik geometri yaklaşımından farklılıklar içerir. Cisimler fraktal ölçüler kullanılarak tanımlanır. Fraktal geometri, model oluşturmak için matematiksel eşitliklerin yerine prosedürleri kullanır. Bu yaklaşımda bir nesne sonsuz kendine benzerlikle ifade edilir; bütün ve parçalar arasındaki detaylar sonsuz kere tekrarlanır. Oluşturma prosedürü kısmi detayları üretmek için rekürsif işlem kullanan kesin kuralları esas almaktadır. Fraktal modelleme teknolojisi; kayalar, sahiller, dağlar, su, bitkiler vb. gibi kendine benzer fakat belirsiz nesnelerin tanımlanması için uygundur. Fraktal modellemenin tarımsal uygulamaları; ağaçlar, otlar ve ürünlerin gelişimini tanımlamayı içerir.

Şekil temelli yaklaşım, nesne detaylarının yaratılmasında prosedürel bir yoldur. Bu yaklaşım son yapıyı oluşturmak için orijini esas alan alt yapının değiştirilmesi veya arttırılması için bir grup üretim kuralına dayanır. Bitkileri etkili bir şekilde

tanımlamakta kullanılan L Sistemi, buna tipik bir örnektir. Örneğin bir ağaç, gövde, dallar ve yapraklardan oluşan bir bütün olarak tanımlanabilir ve kendisinin gelişim kuralları doğrultusunda yapraklar, dallar ve gövde özel bir yolla birleştirilebilir ve böylece ağaç simüle edilir.

Parçacık sistem yaklaşımı, düzensiz ve belirsiz nesnelerin simüle edilmesinde diğer bir yoldur. Dinamik ve rastgele özelliklere sahip bir nesneyi mükemmel bir şekilde yansıtabilir, aynı zamanda su, ateş, bulutlar, sis, orman, göller gibi zamana bağlı olarak değişebilen cisimleri çok iyi bir şekilde tanımlar. Bu yaklaşımda temel fikir, basit şekle sahip birçok mikroparçacık kullanarak nesnenin simüle edilmesidir. Bu parçacıklar küresel, elipsoid, küp ve diğer başka biçimlere sahip olabilir. Belirli kurallar altında parçacıkların ölçü, biçim, şeffaflık ve rengi zamanla değiştirilebilir ve parçacıkların hareketleri genel olarak stokastik süreçlerle simüle edilmektedir. Her parçacık doğum, gelişme ve ölüm olmak üzere üç aşamayı geçirir.

Görselleştirme ve Hareket Kontrol

Görselleştirme, meydana gelen görüntüleri gerçek hayata uyarlar. 3D animasyonun, anahtar fonksiyonlarından bir tanesidir. Görselleştirme nesnenin geometrik biçimine, bağlı konumuna ve yüzey özelliklerine göre uygun algoritmalar kullanılarak gerçekleştirilir. Işın atma (raycasting) ve ışın izleme (raytracking); gizli yüzler, materyal, doku haritalama, gölgeleme, gölgelenme ve ışıklandırmanın gerekli olduğu durumlarda dikkate alınan ana algoritmalarıdır.

Bilinen gizli algoritmalar genel olarak ressam algoritması, Z buffer, arka yüz algılama ve Scanline Z buffer'ı içermektedir. Z buffer, çoğu zaman kullanım kolaylığı için tercih edilir. 3D animasyonlarda gerçekliğin sağlanması için görünüşlerin renkleri; onların yansımaları ve çevrim karakteristikleri, ışık modeli, gözlemcinin bakışı gibi birçok faktör tarafından belirlenir. Nesne yüzeyi, belirli materyallerle sıklıkla kaplanır ve tekstür haritalama ile işlenmelidir. Gölgeleme genel olarak Lambert modellemesi, Gourand modellemesi, Phong modellemesi vb. yöntemler içerir. Yansıtıcı, ortam ve yansımalar görselleştirilen nesnenin gerçekliğini etkilemektedir.

Hareket kontrol, 3D animasyonun bir diğer anahtar teknolojisidir. Hareket kontrol; animasyonun etkileşimli olarak nasıl kontrol edileceğinin yanında anahtar çerçeve animasyonu, kinematik, dinamik, dönüşüm animasyonu, rol animasyonu, yazı animasyonu, kamera hareketleri, vb. gibi unsurları içermektedir.

Ara çerçeveleri oluşturmak için anahtar çerçeve animasyonunda sıklıkla enterpolasyon yapılır. Anahtar çerçeve animasyonu şekil temelli ve parametre temelli olmak üzere iki çeşittir. Anahtar çerçeve animasyonu genel olarak parametre değişimi ve hareket yakalama ile animasyon etkilerini üretir. Kinematik teori veya dinamik teoriyle kontrol edilen animasyonlar sırasıyla kinematik animasyon ve dinamik animasyon olarak adlandırılır. Çoğunlukla nesnelerin hareket ve hızlandırılmalarını tanımlamak için kullanılırlar. Abartma etkileri için dönüşüm

animasyonu, nesneyi dönüştürür. Rol animasyonu, vücut dilini veya animasyonu kişiselleştirme hareketini kullanır. Yazı animasyonu ise nesnelere anahtar çerçeve hareketini tanımlamak için yazı dilini kullanır.

3D Yazılım Araçları

Piyasada 3D animasyon için birçok yazılım aracı bulunmaktadır. Popüler olanlardan bazıları 3D Studio, 3D Studio Max, Maya, Softimage/3D, PowerAnimator'dür. Bunlar arasında 3D Studio Max, Maya, Softimage/3D yazılımları kişisel bilgisayarlar ve iş istasyonları için popüler yazılımlardır. Bu araçlar güçlü fonksiyonlara ve kullanıcı dostu arayüzlere sahiptir, dolayısıyla kullanıcılar 3D modelleri kolaylıkla oluşturabilmekte, hareket takibini kurabilmekte ve bir nesnenin yapı ve ışık modellerini tanımlayabilmektedir. Bu yazılımların hepsi hızlı görselleştirme ve çözünürlüğün kolayca ayarlanması, renk derinliği, alfa kanalı ve elde edilecek nihai animasyon için keskinliği yumuşatma (antialiasing) özelliklerine sahiptir. Ayrıca bu programlar değişik animasyon formatlarını ve depolama için veri sıkıştırma teknolojilerini kullanıcıların kullanımına sunmaktadır.

İlave olarak Maya yazılımı MEL adında tekrarlı işlemleri otomatik hale getiren bir işletim arayüzü içermektedir. Dikkate değer bir performans elde etmek için 3D Studio Max, 3D görüntü kartı ile hızlandırılabilen OpenGL ve Direct 3D gibi popüler 3D grafik arayüzlerini destekleyebilmektedir. 3D Studio Max, çok işlemcili (SMP, Symmetric Multiprocessor) bilgisayarlarda çalışabilen ve ağda paralel görselleştirme yapabilen simetrik çok işlemci kabiliyetli çok kanallı bir yazılımdır.

6.6.3 Sanal Gerçeklik

Sanal gerçeklik (SG); bilgisayarları ve ilgili multimedya ekipmanlarını (örneğin video, ses ve dokunmatik cihazlar) kullanarak kullanıcıların görebileceği, duyabileceği, değiştirebileceği, sanal çevre ile ilişkiye geçebileceği ve arka planından beslenen gücü hissedebilecekleri üç boyutlu sanal bir dünya oluşturmayı hedeflemektedir. Sanal gerçeklik; bilgisayar grafikleri, görüntü işleme, desen algılama, bilgisayarlı görme, akıllı kontrol, simülasyon, otomasyon, sensör, ses işleme ve belirleme, mekanizmalar, optik, ağ, paralel işleme vb. gibi birçok yüksek teknolojiyi içerir.

Sanal gerçeklik sisteminde kullanılan donanım cihazları şunları içermektedir:

- *görüntüleme ve görüntü tarama sistemleri*; geleneksel bilgisayar ekranı, başa takılan gösterge, pozisyon izleyici (mekanik armatürler, ultrasonik sensörler, manyetik izleyiciler, optik pozisyon izleyiciler vb.) içermektedir;
- *ses üretileri ve izleyicileri*; ses sentezleyicileri, ses belirleyiciler ve ses kaynağı belirleyicileri içermektedir;
- *dokunsal ve kuvvet geri besleme cihazları*; geleneksel fare, iztopu veya kumanda kolundan, kuvvet topları, manipulatörler, enstrümanlı eldivenler,

kuvvet geri besleme sistemleri, dokunsal geri besleme etkileşim sistemleri, vb.'ye kadar geniş bir aralığı içermektedir.

Piyasada birçok SG donanımı bulunmaktadır. Bununla birlikte yüksek hızlı görüntü işleyiciler gibi birçok mevcut cihazın performansı SG'nin ihtiyaçlarını henüz karşılayamamaktadır. İnsan psikolojisinin ve fizyolojisinin sanal gerçekliğe nasıl uygulanacağı hakkında yapılan araştırmalar henüz açık sonuçlar vermemiştir. İlave olarak, SG'nin yaygın kullanımının önündeki bir diğer engel ise pahalı donanımlar olabilir.

Bazı SG yazılımları çalışması için bir iş istasyonuna ihtiyaç duyarken, çoğu SG yazılımı yüksek performanslı hızlandırıcı kartlar ekleyerek kişisel bilgisayarlarda çalışabilir. SG yazılımına bir örnek olarak Dimension International şirketi tarafından kişisel bilgisayarlar için geliştirilen VRT yazılım araç kutusu verilebilir. VRT sanal bir dünya oluşturmak için kullanılacak güçlü bir geliştirme ortamıdır. Diğer sanal gerçeklik programları örnekleri; Human Interface Technology Laboratory tarafından geliştirilen Virtual Environment Operating Sell (VEOS), Sense&Corperation tarafından geliştirilen World Tool Kit (WTK), Alberta Üniversitesi tarafından geliştirilen Minimal Reality (MR), Gemini tarafından geliştirilen Generic Visual System (GVS), SGI tarafından geliştirilen Sky Writer and Reality Engine, Division Şirketi tarafından geliştirilen Division Provision, Straylight Şirketi tarafından geliştirilen Photo VR şeklindedir.

Sanal gerçeklikteki bir diğer önemli gelişme; hiper metin için HTML (Hypertext Markup Language) gibi bir web tarayıcısında interaktif 3D animasyonlar yapmak için kullanılan bir programlama dili olan VRML (Virtual Reality Modeling Language)'dir. VRML; 3D modelleme, görüntü yapısı, animasyon, ses ve sensör fonksiyonlarını ayarlamaya olanak sağlamaktadır. 3D modelleme yapmak için yorumlama amacıyla VRML, bir web tarayıcısına veya ücretsiz indirilebilecek web yazılım eklentisine (plug in) ihtiyaç duymaktadır.

6.6.4 Tarımda 3D Animasyon ve Sanal Gerçeklik Uygulamaları

Bitkiler için yapılan bilgisayar modelleri ve görselleştirme tarımsal bilişim alanında önemli bir konudur. *Sanal bitki*, üç boyutlu bir uzayda bir bitkinin büyüme sürecinin ve yapısal gelişiminin bilgisayar simülasyonudur. Bir organizma olarak bir bitkinin gelişme sürecini ve çevre koşullarından etkilenmesini simüle etmek öyle karmaşıktır ki, aslına uygun bir şekilde simülasyonu yapmak zordur. Fakat araştırmacılar bu alanda önemli ilerlemeler kaydetmişlerdir.

Sanal Bitkiler için Bilgisayarda Modelleme Metodları

Bitkilerin grafik veya görüntülerinin üretilmesine yönelik olarak bilgisayarlar, botanikçiler ve matematikçiler tarafından birçok model geliştirilmiştir. Geliştirilen bu modellerle sadece sanal bitkiler değil, sanal alanlar ve çevreler de üretilebilmektedir. Bu modeller arasında L Sistemi, A Sistemi, tekrarlama

fonksiyonu sistemi (IFS, Iteration Function System), parçacık sistemi ve otomasyon yer almaktadır.

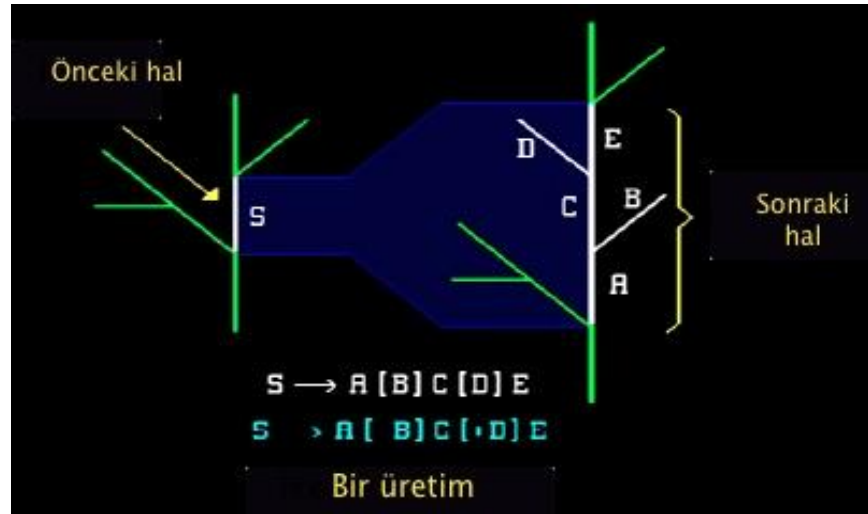
L Sistemi

L sistemi ilk defa 1968’de Lindenmayer [1] tarafından tanıtılmıştır ve sonra Prusinkiewicz tarafından bitki modelleme amacıyla geliştirilmiştir. L sistemi aslında, yaratılan sonlu rekürsif işlemlerle bir dizi oluşturmak için aksiyom ve bir üretim kümesi kullanan bir dizi yeniden yazma sistemidir. Bu dizi bitkinin yapısı olarak yorumlanabilir. L sistemi bitkilerin morfolojik yapılarının tanımlanması için uygun bir sistemdir.

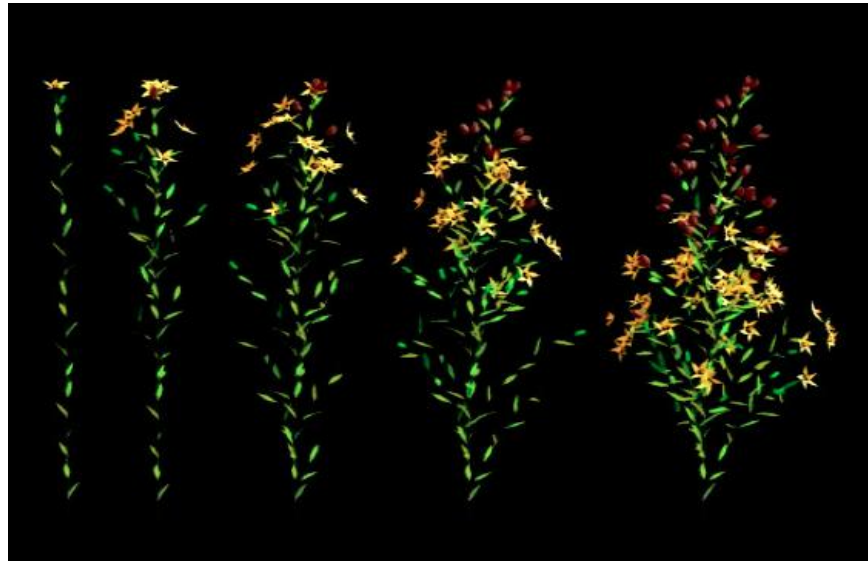
Basit bir L sistemi *DOL sistemi* olarak adlandırılır ve bu sistemde “D0” deterministik anlamına gelmektedir. DOL sistemi bağlamdan serbest olup fraktal görüntüyü oluşturmak için kaplumbağa yorumlama algoritması kullanmaktadır. Bir kaplumbağa ileriye doğru, geriye doğru, sağa doğru ve sola doğru hareket edebilmektedir. Kaplumbağanın durumu dört sembolle tanımlanır: I, +, -, [,and]. I sembolü herhangi bir yönde hareketin bir birim uzunluğu anlamına gelmektedir. Yön + ve – sembolleri ile belirlenir. + sembolü (saatin tersi yönünde) sola dönme anlamındayken, – sembolü (saat yönünde) sağa dönme anlamındadır. [,and] sembolü küme anlamı taşımaktadır. [sembolü kaplumbağanın mevcut durumunu kümeye iterken,] sembolü ise kümeden yeni bir durum çıkarmaktadır ve bunu kaplumbağanın mevcut durumu olarak atamaktadır. Örneğin $S \rightarrow A [B] [C] [+D] [E]$ modeli kullanıldığında Şekil 1’in sağ parçası şeklin solunda bulunan önceki S ana şekli tekrarlı işlemle elde edilen son halini vermektedir.

Bağlam hassas L sistemi, birbirine komşu bitki parçaları arasındaki etkileşimler için geliştirilmiştir. Bağlam serbest L sistemi, genel bağlam hassas L sisteminin sadece özel bir halidir. Gelişim sürecine rastgelelik ekleyebilmek için stokastik L sistemi önerilmiştir. Gelişimin tahmin edilemediği veya aynı aksiyom ve kurallarla yeniden oluşturulamadığından bu sistem her zaman farklı çıktılar üretebilmektedir. Bir bitki ile çevresi arasındaki karşılıklı etkileşimi simüle etmek için Mech ve ark. tarafından açık L sistemi önerilmiştir.

Devam eden olayları simüle etmek için parametrik L sistemi geliştirilmiştir. Şekil 2’de bir bitkinin gelişim süreci ve çiçeklenme aşamaları gösterilmektedir. Prusinkiewicz ve ark. zaman ayarlı L sistem ve diferansiyel L sistemini, devamlı gelişim sürecindeki dallanma modellerinin yapılması için formal bir taslak olarak önermiştir.



Şekil 1. L-sistemi prensibi örneği (Prusinkiewicz).



Şekil 2. Bir bitkinin çiçek sıklığı ve gelişme işlemi.

L sistemi fraktal nesne yapmak için güçlü bir araçtır, fakat emek yoğun yazım kurallarından dolayı genel kullanıcılar onu anlamakta zorlanmaktadır. Aynı zamanda bu yöntemle gerçeğe yakın bitki gelişiminin simüle edilmesi oldukça karmaşıktır.

A sistemi

Honda [3], dallanmanın açı ve uzunluklarını kullanarak bir ağaç şekli tanımlamıştır. Hondanın çalışmasına dayanarak, Anon [4] bitki şekilleri için bir simülasyon modeli olan A sistemini geliştirmiştir. Sonrasında ilave olarak GMT1, GMT2, GMT3, GMT4 modellerini geliştirmiştir. A sistem; dallanma yapılarının özelliklerini ve dalların açılarını gelişigüzel seçerek, farklı bitkilerin görsel olarak simülasyonunu yapabilir ve ayrıca cazibe algoritmaları (attractor algorithms)

kullanarak bitkinin şekli üzerine ışığın, rüzgarın ve yer çekiminin etkilerini de simüle edebilir.

Tekrarlama Fonksiyonu Sistemi

Tekrarlama fonksiyonu sistemi (TFS), 1981’de Hutchinson tarafından tanıtılan etkili bir bitki simülasyon sistemidir. Barnsley ve Demko [5], TFS’yi eğrelti otunun yapraklarını tam kendine benzerlik ile üretmek için uyarlamışlardır. Sonrasında Barsley ve ark., bitki parçaları arasında farklı kendine benzerlikleri somutlaştırabilen yinelemeli TFS’yi önermişlerdir. Onların yaptıkları ilave çalışma, herhangi bir boyutta fraktal geometrileri yapabilmek için TFS’yi uygun bir yol haline getirmiştir ve TFS’nin hızlıca gelişiminin önünü açmışlardır. 1991’de Prusinkiewicz ve Hammel [6], uygulanabilir dönüşüm sıralamalarını sınırlayan bir araç sağlayarak TFS’nin orijinal tanımını genelleştiren dil sınırlı TFS’yi geliştirmiştir.

Parçacık Sistemi

Reeves [7], parçacık sistemini çimen ve orman simülasyonunda kullanmıştır. Bu stokastik modelde, yapısal parçacık sistemi karmaşık doğal senaryoları tanımlanması için uygulanmıştır. Bir ağaç oluşturulduğunda gövde ve ana dallar klasik geometri modeli ile üretilmektedir, fakat yapraklar gibi küçük organlar ise genel olarak parçacık sistem yaklaşımını kullanmaktadır. Bu durum bütün bitkilerin tanımlanmasında uygun olmayabilir. Şekil 3, Reeves [7] tarafından yapılan bir orman senaryosu simülasyonunu göstermektedir.

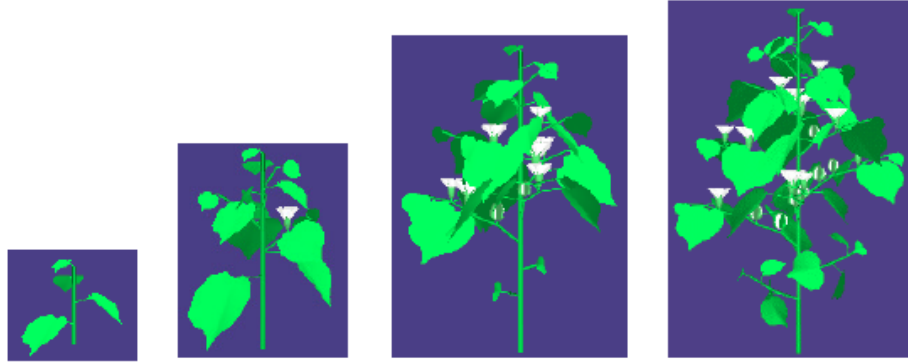


Şekil 3. Reeves [7], tarafından yapılan bir orman senaryosu simülasyonunu.

Otomasyon

De Reffye ve ark. [8], sonlu otomasyon kullanarak bitki morfogenez modelini geliştirmişlerdir. Bitkinin büyümesi, dormansisi ve ölüm süreçleri, Markov zincirleri ve durum geçiş grafikleri kullanılarak sonlu otomasyon modeli ile tanımlanmaktadır. Çok ölçekli bir model olan bitki topolojisi modeli (BTM), Godin ve ark [9] tarafından Reffy'nin çalışmalarına dayanılarak tanıtılmıştır. BTM modeli, farklı zaman ölçüleri kullanarak topolojik yapıyı tanımlayabilmektedir. Zhao ve ark. [10], bitki gelişimini simüle etmesi için çift ölçekli otomasyon isimli bir model geliştirmişlerdir. Çift ölçekli otomasyon makro ve mikro durumlar içermektedir. Bu modelde makro ve mikro durumların kombinasyonu doğrultusunda, bitkinin topolojik yapısı görselleştirilmektedir. Botanik ve tarım temelli olduğu için bitki gelişimini daha doğru simüle edebilmektedir. Şekil 4'de çift ölçekli otomasyon sistemi ile yapılan pamuğun gelişim süreci örneği görülmektedir.

İlave olarak Oppenheimer [11], bitki ve ağaçların gerçek zamanlı simülasyonu için fraktal metodu kullanmıştır. Viennot [12], bitki yapısı modeli için dallanma matrisi modelini önermiştir. Önerdiği bu matrisi kullanarak dallanma noktaları sayısını ve onların ilişkilerini tanımlamıştır ve ayrıca tekrarlama ile bitkinin fraktal yapısını geliştirmiştir. Greene [13], bitki ve engelleyicisi arasındaki etkileşimi simüle etmek için voksel uzayını kullanmıştır ve ayrıca voksel uzayında sarmaşık gülünün gelişigüzel büyüme süreçlerini modellemiştir.



Şekil 4. Dört farklı dönem için pamuğun gelişim simülasyonu.

Sanal Bitkiler için Bilgisayar Modelleme Araçları

Sanal bitkiler, bitkinin gelişim hızı gibi bitki gelişimine ait bilgilere göre Visual C, Delphi, Visual Basic, vb. gibi genel programlama dilleri kullanılarak oluşturulabilmektedir. Bu diller, OpenGL ve Direct3D gibi 3D grafik arayüzlerini çağırabilme yeteneğine sahiptir. Sanal bitkiler yapmanın diğer yolu 3D Studio Max, Maya vb. gibi 3D animasyon platformları kullanmaktır. Bu programlar birçok alanda kullanıcıların 3D animasyon ve modeller oluşturabilmesi için kullanışlı arayüzlere ve güçlü fonksiyonlara sahiptir. Bununla birlikte bu metodlar ağırlıklı olarak kullanıcı

ve makina arasındaki diyaloga bağı bir şekilde deęişmektedir. Bunun anlamı sonuçlar oldukça subjektiftir ve nadiren bitkinin orijinal yapısına sadık kalır. Son yıllarda Xfrog, SpeedTree, AMAP, GOSSYM/COMAX, CottonPlus, PlantStudio, VirtualPlants, Digimation-Tree Factory, Tree Druid vb. gibi bitki simülasyonlarına yönelik özel araçlar içeren yazılımlar çıkmıştır. Burada örnek olarak üç araçtan bahsedilecektir.

Xfrog

Xfrog bitki modelleme yazılımı, GreeWorks Gbr (Almanya)'den Bernt Linterman [14] tarafından geliştirilmiştir. Ağaçlar, çalılar vb. gibi dallanabilen çok sayıda nesnelere bu yazılım tarafından güvenli bir şekilde oluşturulabilmektedir. Yazılım bitkinin yapısal ve geometrik bileşenlerini üç grup altında tanımlamaktadır. Birinci grup; gövde, yapraklar, dallar ve diğer geometrik temel yapılar gibi grafiksel nesnelere oluşturmaktadır. İkinci grup, dalların yerleşimi ve taç yaprağının yeri vb. gibi bitkinin yapısını ifade eder. Üçüncü grup, oluşturulan bitkinin bütün şeklini etkileyen küresel deęişkenleri belirtmektedir. Her bileşen basit bir geometriyi tanımlayan temel parametreler kümesini içerir. Kullanıcılar grafiksel diyaloga tıklayarak bitkinin görüntüsünü yapılandırabilir ve modelleme süreci üzerine kontrol sahibidir.

Xfrog ışık ve yerçekimine yönelim gibi Maya, Ryshade tarzı yazılımlara çıktı verebilen güzel özelliklere sahiptir. Ancak bitkilerin yapısını ve gelişimini biyolojik olarak simüle edemediği gibi yaprakların damar yapılarını da destekleyememektedir. Bu sebeple gerçek 3D yaprakları üretmek zordur. Aynı zamanda Y birleşimleri yerine sadece silindireleri birleştirdiği için bu yazılım tarafından oluşturulan ağaçlar doğru olmayan dallanma geometrilerine sahiptir.

Arktik İzleme ve Deęerlendirme Programcısı (AMAP)

Gelişen tomurcukların doğum ve ölümlerine bağı olarak, De Reffye ve ark. [8], bazı parametreler ile bitki gelişimini kontrol etmemize izin veren prosedürel bir model geliştirmişlerdir. CIGR (Fransa) tarafından geliştirilen AMAP (Arctic Monitoring and Assessment Programmer); bitkilerin ve ürünlerin mimarisi, fonksiyonellięi, büyümesi ve üretilmesini simüle etmek, analiz etmek ve ölçmek için gerekli metodları sağlamaktadır. AMAP, bitkilerin büyüme mekanizması modelleri ve görselleştirme modellerinin birleştirilmesinde büyük bir ilerleme kaydetmiştir.

Farklı fonksiyonları sağlayabilmek için AMAP; AMAPMod, AMAPsim, AMAPpara, AMAPhydro gibi birçok alt modele sahiptir. AMAPmod modeli yeterli uzaysal bilgiyi esas alarak bitkinin gelişim modelinin yapılmasında çoğunlukla kullanılmaktadır. AMAPmod'da geliştirilen bitki modeline göre AMAPsim modeli, bitkilerin gelişim süreçlerini 3D görselleştirmeyle simüle etmektedir. AMAPpara modeli, bitkinin ekofizyolojik fonksiyonu ve mimari gelişimi arasındaki döngüsel etkileşimin kümülatif çıktısı olarak uzun dönemli bitki gelişimini tanımlamaktadır.

AMAPhydro modelinde, bitkinin büyümesine bitki mimarisini etkisini ve bitki içerisinde besinlerin dağıtım işlemiyle birlikte bitki büyümesini simüle etmek için bir bitki hidrolik modeli yapılmaktadır.

Bitki Stüdyosu

Bitki stüdyosu Kurtz-Fernhout [15] tarafından Almanya da geliştirilmiştir. Bu stüdyoda kullanıcılar birçok bitki çeşidi üretebilmektedir. Bir bitkinin meristem, dallanma, antotaksis (anthotaxis), yaprak, boğum arası, çiçek ve meyve gibi özelliklerini kullanıcının sanal bir ağaç kolayca oluşturmasında seçebilmesi için bileşenler şeklinde önceden tanımlanabilir. Bitki stüdyosu sonuçları 3DS, DFX, WRL, OBJ, POV, LWO vb. gibi birçok bilindik değiştirme formatlarında çıktı olarak alınabilmekte, 3D araçları ile daha sonra işlenebilmektedir.

6.6.5 Sanal Bitkilerin Geleceği

Sanal bitkilerle ilgili büyük gelişmeler kaydedilmiş olmasına rağmen gerçek bitki büyümesinin simüle edilmesi henüz mümkün olamamıştır. Gerçek bitki büyümesinin simülasyonu; bitki topolojisi ve geometrik model bilgisinin yanında bitki fizyolojisi, bitki ekolojisi ve çevresel etkinin karmaşık faktörlerinin dikkate alındığı ekofizyolojik model bilgisine de ihtiyaç duymaktadır. Farklı disiplinler ve teknolojilerin birleştirilmesinin, sanal bitkilerin gelişmesini ve pratik uygulamalarını ilerleteceğine inanılmaktadır.

Sanal bitkiler araştırmalarda ve tarımsal gelişimde önemli roller oynayacaktır. Örneğin zamandan ve paradan tasarruf sağlamak için sanal bitkiler, sanal deneyleri gerçekleştirebilir. Bu sanal deneyler, bitkilerin gelişim kurallarının keşfedilmesine yardımcı olacak ve çiftlik yöneticilerinin uygun kararlar almasını kolaylaştıracaktır. Sanal bitkiler aynı zamanda bitki fizyolojisi, ekolojisi vb. gibi alanlarda yapılan araştırmaları ve bunların tarımdaki gelişmelerini destekleyebilir. Dahası sanal bitkiler, tarım teknisyenleri ve çiftçilere ileri teknoloji ve bilgi aktarmada önemlidir. Sanal bitkilerdeki gelişmeler tarımsal bilişimde gelişim rolü oynayacağı ve böylece sanal tarla, sanal peyzaj, sanal çiftlikler, sanal bonzai ve sanal çevreler vb. gibi uygulamaların performansının hızla iyileşeceği tahmin edilebilir.

Kaynaklar

1. Lindenmayer, A. 1968. Mathematical models for cellular interaction in development, Parts I and II. *J. Theoretical Biology* 18: 280-315.
2. Prusinkiewicz, P., and A. Lindenmayer. 1990. *The Algorithmic Beauty of Plants*, 40-50. New York, NY: Springer Verlag.
3. Honda, H. 1971. Description of the form of trees by the parameters of the tree-like body: Effects of the branching angle and the branch length on the shape of the tree-like body. *J. Theoretical Biology* 31: 331-338.
4. Anon, M., and T. L. Kunii. 1984. Botanical tree image generation. *IEEE Computer Graphics and Applications* 4: 10-34.

5. Barnsley, M. F., and S. Demko. 1985. Iterated function systems and global construction of fractals. *Proc. Roy. Soc. London*, Vol. A 399, 243-275.
6. Prusinkiewicz, P., and M. Hammel. 1991. Automata, language, and iterated function systems. *Fractal Modeling in 3D Computer Graphics and Imagery*, ACM SIGGRAPH, Course Note C14, eds. J. C. Hart and F. K. Musgrave, 115-143.
7. Reeves, W. T., and R. Blau. 1985. Approximate and probabilistic algorithms for shading and rendering structured particle systems. *Computer Graphics* 19: 313-322.
8. De Reffye P., C. Edelin, J. Francon, M. Jaeger, and C. Puech. 1988. Plant models faithful to botanical structure and development. *Computer Graphics* 22: 151-158.
9. Godin, C., and Y. A. Carglio. 1998. A multiscale model of plant topological structures. *J. Theoretical Biology* 84(191): 1-46.
10. Zhao, X., P. de Reffye, F. Xiong, and B. Hu. 2001. The dual-scale automation model of virtual plant development. *J. Computers* Vol.24.
11. Oppenheimer, P. E. 1986. Real time design and animation of fractal plants and trees. *Computer Graphics* 20: 55-64.
12. Viennot, X. G., and G. Eyrolles. 1989. Combinatorial analysis of ramified patterns and computer imagery of trees. *Computer Graphics* 23: 31-39.
13. Greene, N. 1989. Voxel space automaton: Modeling with stochastic growth processes in voxel space. *Computer Graphics* 23: 175-184.
14. Lintermann, B., and O. Deussen. 1999. Interactive modeling of branching structures. *IEEE Computer Graphics & Applications* 19: 56-65.
15. www.Kurtz-Fernhout.com