

21 Şubat

Sahne Yönetimi

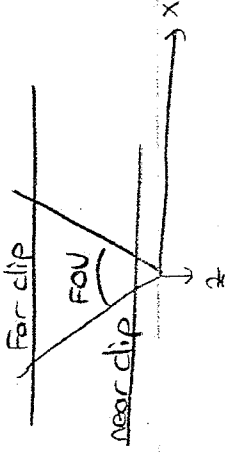
Oyun programının temel aşamalarından biridir. Bulunan sahnedeki karakterlerin modellenmesi, detaylarının verilmesi ve bunların hafızayı yönetim teknikleriyle gerektiği şekilde günclenmesidir.

1) Sahne gereksit olan nesne ve karakterlerin o anki harekete göre çıkarılması ilk işlemidir.

Arkaplan çıkarma (Backpase Culling)

Kamera oculusına girmeyen nesnelerin alınması işlemidir. Normalde (rendering) derinlik kotonadına işlemi yapılmalıdır. Bunun yerine gerçekleştirilecek çıkarma işlemi de bu işlemi yapılabilir.

Kamera ve Hacim Gözün Açısı (Camera View Volumes)



Bilgisayar ortamında nesnenin modellenmesi kamera pozisyonu ve görüş mesafesiyle ilgilidir. 2 boyutlu nesne, 3 boyutlu nesne primitiv yapıyla yapılır.

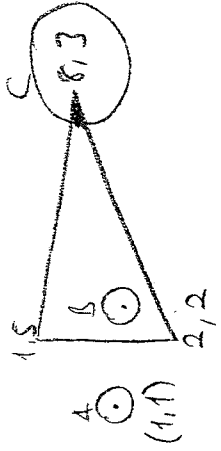
— Bir nesnenin bilgisayar ortamında görülmesinde 3 durum vardır.

- 1) Tamamen görülebilir olanlar, 2) Kısmen görülebilir olanlar, 3) Tamamen görülmeyen yapılar.

* Kesit alma işlemi yarı görülebilir nesneler içindir.

— Bir nesnenin olan içinde olup-olmadığı problemi

— 2 boyutlu bir nesnenin kapalı olan içinde olup-olmadığını bulan program.



Bir nokta B'nin içinde midir?
İçindeyse ne kadar içindedir?

Poltals (Oktomber)

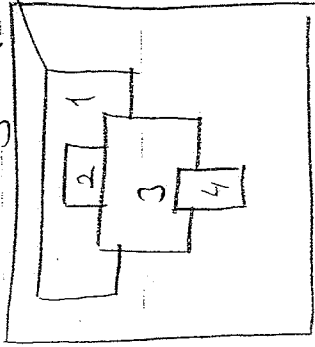
1'den fazla sahneye olan oyunlardaki ortamların oluşturulması aşamasına sona erme gerçekleştirilir. Bir ortamdan bir yada birden fazla ortama geçiş yapılabilir. Bu geçişlerin ortaya konma tekniğinde grafikler (cutgeller) kullanılır.

Görülebilir Nesneler ve Kümeler

Bir nesnenin görülebilmesi kamera pozisyonu ile ilgilidir. Nesnenin ne kadının görüleceğinin ortaya konduğu algoritmalarla bir tanesi z-buffer algoritmasıdır. Benzeri göre işlem yapma algoritmasıdır.

(Geriden - öne doğru) Görülebilir yer öndeki nesnelerdir. (Büyük nesneler için)

Üstten görüldüğü en geride



z-buffer, en gerideki nesneden itibaren en yakındaki nesneye doğru modelleme gerçekleştirilir. Dolayısıyla en fazla götüken nesne en öndekidir.

Seviye Detay Kümeleri

Uzaktaki nesnenin detayları oluşturulmaz. Nesneye yaklaştıkça detaylar verilir. İlerleyen itzoro (mesh) tekniği (Progressive Meshes) kullanılarak detay oluşturulabilir. Bunun için;

a) Matematiksel işlemler yapılır. Alt bölmeleme (Subdivision) tekniği denir. Δ --- \bigcirc nokta sayısını artırarak yapılır.

b) Alt bölmeleme ile oluşturulmuş objeye yapıları kullanılır.

Data sayısı çok fazla olduğunda; geliştirilmiş, iyileştirilmiş alt bölmeleme algoritmlarına başvurulabilir.

Cik Görsel Birlikli Modelleme (Multiresolution modelling)

21 Subat

(2. gösterim)

Bu yöntemin temel amacı utaktaki nesnelerin detaylarını göstermektir. Örneğin, bir dağ yapısı (terrain), ovalar, vadiler gibi doğal yapıların modellenmesinde yaklaşıklık detayların artırılma tekniğidir.

NOT: Önemli!

1) Sahnedeki önemli bir karakter gölgeleridir. Gelişmiş oyun programlarının kullandığı gölgeleme tekniği: "Phong" gölgelendirir.

2) Ortam haritalarının belli bir sırayla sahneye yerleştirilmesi.

3) Görüntü işleme teknikleri yardımıyla da görsel efektlerin, diğer efektlerin sahneye dahil edilmesidir.

NOT: Realflow programı

Çarpışma Tespit Algoritmaları

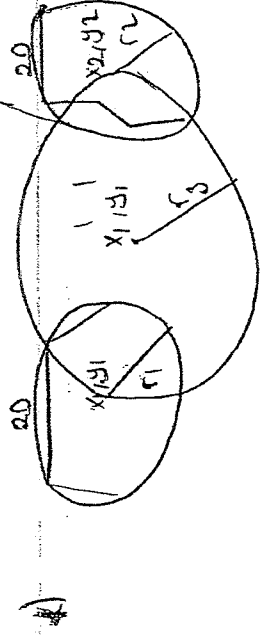
Tanım: 2 boyutlu ya da 3 boyutlu ortamda 2 yada daha fazla nesnenin temas edip, etmediğini bulma işlemidir.

— Oyun motorları içinde "collider" yazılımları ile bu işlem gerçekleştirilebilir.

— Kormasik nesnelerin temas nokta ve yüzeylerinin bulunması problemidir.

1) — 2 boyutlu uzayda temas edip, etmediğini bulmak için nesnelerin tamamını kapsayacak şekilde bilinen bir geometrik şekil içine nesne yerleştirilir.

Akıl teması



β sabit, A hareketli.

HAREKET CHIT, JUNE

$t=0$, x_1, y_1

$t=1$, $x_1 + \Delta x$, $y_1 + \Delta y$

— Yarıçap kontrolü (nesneler birbirine değişir mi?, değişiyor mu?)

(+) Avantajı: Hız meselesidir.

(-) Dezavantajı: Gereksiz abni işgal ettiği için gereksiz temas noktası içerir.

→ A nesnesi ile B nesnesinin kesişiminin bulunması çarpışma tespitidir.

→ Çarpışmanın tespiti hız ve ivme ile ilgilidir.

→ Yapılan işlem, nesnenin tamamını kapsayacak şekilde, basit geometrik şekil içine yerleştirilebilir.

NOT: Nesnenin hızı yolda yavaş olması Δx , Δy yi etkiler.

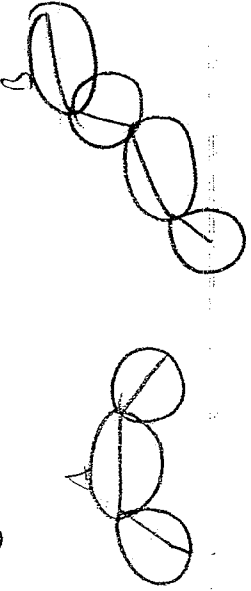
NOT: Çemberdeki sıkıntı gereksiz alanların nesneye dahil edilmesidir. Avantajı hızlı işlem sağlar. (Dikdörtgen içine de alınabilir.)
Bu da 2 olabilir!

***3

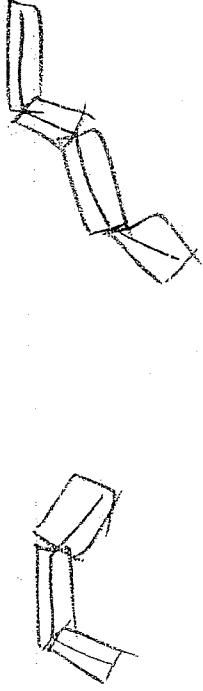
a) → parça, parça nesneyi tanımlayan geometrik şekiller kullanmak.

(+) Avantajı: Gereksiz alanlar atılır.

(-) Dezavantajı: Birden fazla çemberin olması ve kontrol edilecek değerlerin sayısının artması.



b) Her bir parçayı dikdörtgen şeklinde göstermek.
Bu yöntem en çok kullanılan yöntemdir. Özellikle oyunlarda ve animasyonlarda en fazla kullanılan yöntemdir.



4) IDEAL:

3 Hafta

- İdeali, gerçek datanın kendisiyle doğrudan çalışmaktır.
- Fakat, dataların çokluğundan dolayı utun zaman alabilen durumlar ortaya çıkar.
- 3 boyutlu nesneler polinomlar şeklinde modellenmişinde özellikle eğrisel bölgelerde nokta ve yüzey sayısının çokluğu sıkıntılara yol açar.
- Nesnenin yer değişim miktarının bulunması ve bunun modellenmesi kritiktir.
- +0 anında (ilk pozisyonda), nesnenin durumu kaydedilir.
- +1 anında nesnenin yeni pozisyon bilgisi tutularak işlem yapılır.

Üçgenin Köşe Koordinatlarının Normalinin Bulunması:

- 2 vektörün vektörel çarpımı bu 2 vektörün oluşturduğu düzleme dik olan vektörü verir. Buna "yüzey normali" denir.
- (Vektörel çarpım) $\vec{A} \times \vec{B} \neq \vec{B} \times \vec{A}$
(Dışa doğru normal vektör) \neq (İçe doğru normal vektör)
- Normal vektörleri nesnenin hareket yönünü belirlemede kullanılır.
- 2 vektör arasındaki açı "cosinüs teoremi" ile bulunur. (Skaler çarpım)



$$A \cdot B = |A| \cdot |B| \cdot \cos \alpha$$
$$A \cdot B = \frac{A \cdot B}{|A| \cdot |B|} = \frac{A_x \cdot B_x + A_y \cdot B_y + A_z \cdot B_z}{\sqrt{A_x^2 + A_y^2 + A_z^2} \cdot \sqrt{B_x^2 + B_y^2 + B_z^2}}$$

Üçgen - Üçgen Kesişimi

- Normalde 2 boyutlu ya da 3 boyutlu nesnelerin modellenmesinde üçtey şekli olarak ve üçtey parçası olarak üçgen ya da dörtgenler kullanılır.
- Özellikle eğrisel yüzlerde üçgen yapısını tercih etmek faydalıdır.
- Detay olan bölgelerde üçgen boyutunu küçültüp, sayıyı arttırmak faydalıdır.

Hareketli Parçalar

2 durum söz konusudur.

→ 1.) Nesnelerden birinin hareket etmesi, diğerinin sabit durması.

- Bu durumda, hareket eden parçaya göre işlem yapılır.
- Bu durumda problem; üçgen, kare ve dükkene kesişimi problemidir.

→ 2.) Her iki nesnenin birbirine doğru hareket etmesi problemidir.

• Hızları eşit ya da farklı olabilir.

Optimizasyon Stratejileri

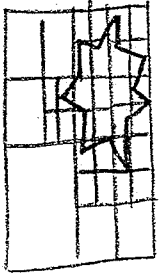
NOT Optimizasyon Stratejilerinin Amacı: Uygun bir veri yapısı kullanarak hızlı bir şekilde işlem yapmaktır. Bunun için çeşitli ağaç yapıları geliştirilmiştir.

yarar



1) Oc-trees!

- Amaç! Nesne sınırlarını takip edecek şekilde mümkün olduğunca kadar büyük düzgün parçaları şeklinde modellemektir.
- Böylece, sınır-daki temsil edilebilen nokta sayısı azaltılmış olur.
- Nesne sınırlarını takip edecek şekilde büyük bir karenin parçalara bölünerek sınırlara yerleştirilmesidir.



2) KD Trees! Amaç! Nesne sınırlarını takip edecek şekilde veriyi otaltmaktır.

3) BSP Trees!

Amaç! Bu ağaçların amacı, nesnenin alanını basit düğümlerle tanımlamaktır.

- Galısmo şekli; Nesne kenarındaki çizgilerin birbiriyle kesişecek şekilde düğüm oluşturmalarıdır. Böylece nesne parçaları şeklinde modellenir.

4) K - Ops!

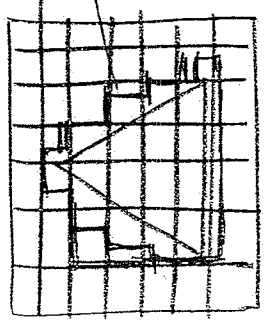
Amaç! Nesne bölgelerini kapsayacak şekilde ağaç yapısı oluşturmaktır.

- Böylece, nesnenin kendisi basit bir ağaç olarak tespit edilir.
- Fazla olanlar bu algoritma ile oluşabilir.

5) Uniform Grids (Yaylı Izgara) Kullanmak

Amaç Nesnenin tamamını kapsayacak şekilde eşit aralıklı ve eşit boyutlu izgara kullanmaktır.

→ Grid boyutu ne kadar küçükse o kadar düşük nesne sınırı elde edilmiş olur.



6) Nokta Gıptı Aaltma İstemı

Çarpıma tespıtinde dıkkat edılecek dığer bır işlem verıyı aaltmaktır.

Amaç, Hızlı bır şekilde işlem yapmaktır.

Fizik Simülasyonu

Gerek dünyadaki değerlerin fiziksel etkileri bilgisayar ortamına cesitli deęistkenler ve fonksiyonlar olarak oktarıldığı "yapılar" simülasyonlar (benzetimler)" dir.

Kinematik: Herhangi bir nesneye uygulanan kuvvet gösterdi edilerek, bu nesnedeki hareket çalışmalarını bilimidir.

Dinamik: Fiziksel hareket çalışma disiplini dir.

1) Forward (ileri) Dinamik: Uygulanan kuvvetin sonucu olarak hareketin hesaplanmasıdır. (Hedef yok, sadece kuvvet var)

2) Inverse (Ters) Dinamik: Gerekli hareketin sağlanabilmesi için uygulanması gereken kuvvet miktarının hesaplanması disiplini dir. Özellikle simülasyon ve animasyon işlemlerinde daha çok kullanılır.

Parçacıklar: Büyük boyutlu nesnelerle işlem yapmak matematiksel açıdan sıkıntılıdır. Bunun yerine 2 ve 3 boyutlu uteda daha küçük parçalı nesnelerle çalışılır.

Yaygın olarak 2 boyutlu cember, 3 boyutlu küre yapısı kullanılır.

Bunun nedeni, orijin noktası ve yarıçapının bilinmesinin yeterli olmasıdır.

→ Her bir parçacığın 3 bileşeni vardır.

→ Pozisyon: x

→ Hız: $v = dx/dt$

→ İvme: $a = dv/dt$

→ Hızın Değeri: $a = a_0$

$$v = \int a_0 dt = a_0 t + v_0$$

$$x = \int v dt = \frac{1}{2} a_0 t^2 + v_0 t + x_0$$

NOT $v_0 = 0$ ise parçacık, (x_0) başlangıç noktasındadır.

→ Kütle: m

→ Momentum: $p = m \cdot v$

→ Kuvvet: $f = dp/dt = m \cdot (dv/dt) = m \cdot a$
↪ $f = m \cdot a$

→ YERÇEKİMİ KUVVETİ?

→ $f = m \cdot g$ → $g = -9.8 \text{ m/s}^2$ olarak kabul edilir.

ama Dünyanın her yerinde yerçekimi kuvveti aynı değildir.

Kutuplarda en fazla, Ekvatorda en azdır.

→ Sürtünmenin götardı edildiđi ortamlarda yerçekimi kuvveti uygulamalara dahil edilmelidir.

Kuvvet Gesitleri

1) Yay (Sönümleyici) Kuvveti: Aşağı - yukarı, sağa - sola hareket eden yayların hareketi ve sönümasyonu ile ilgilidir. Yerelekinsiz ortamda yayda sadece belli katsayılarla göre basit hareket beklenir. Bu katsayı k'sönümüdür.

2) Sönümme Kuvveti: Hareketi engelleyen kuvvettir.

Statik ve Dinamik türleri vardır.

Gerçekçi olanı dinamik sönümme kuvvettir.

Özellikle çarpışma ve sonrasında nesnelerin hareketinin modellenmesinde kritiktir.

NOT) Oyun programlamada çarpışma ortamlarını, kuvvetlerin vektör ya da impuls (birim darbe) şeklinde olduğu kabul edilir.

Bu darbelerin belli zaman aralıklarıyla nesneye etki ettiği düşünülür.

NOT) Kuvvetin sürekliliği nesne üzerindeki etkiyi, momentumu etkiler.

KATI NESNELER

→ Atalet Meselesi :

- Nesnelerin fiziksel yapısından kaynaklanan iç enerjiye "atalet" denir.
- Nesnenin kendi iç dinamiklerinden kaynaklanan bir tür enerjisidir.
- Atalet, nesnenin başlangıç enerjisine, başlangıçta uygulanan kuvvete bağlıdır. Ve bu durum nesnenin hem arızal hem de düzensel hareketlerini etkiler.
- Özellikle yuvarlanan nesneler için önemlidir, değerlidir.

Katı Nesnelerin Garpisması

2 farklı daronış söt konusudur.

Y.1.) 2+ yönlere hareket.

Y.2.) Kırılmalar.

* Nesneyi farklı boyutlarda küçük parçalara bölerek modellemek esastır. Böylece kırılma anında her bir parçaya etki edecek kuvvete göre bir dağılım, yere sacılma işlemi olarak tır.