KARADENİZ TEKNİK ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
BİLGİSAYAR GRAFİKLERİ LABORATUARI

Görünmeyen Yüzey ve Arkayüz Kaldırma

1. Giriş

Bilgisayar grafiklerinin en önemli problemlerinden biri katı nesnelerin görünmeyen yüzeylerinin kaldırılmasıdır. Görünmeyen yüzeylerin kaldırılması, belli bir bakış noktasına göre görüntü düzlemindeki herhangi bir piksele karşılık gelen yüzeylerden en yakın olanını belirleme (diğerlerini kaldırma) işlemidir. Burada yüzeylerin birbirlerini kapatmamaları halinde bakış noktasından görülebilecekleri varsayılmaktadır. Arkayüz kavramı ise bunun tam tersidir. Yani herhangi bir yüzeyin önünde başka bir yüzey olmasa da o yüzey arkayüz veya başka bir deyişle bakış noktasına göre ters çevrilmiş bir yüzey olduğundan görülmesi imkansız olmasıdır. Ters çevrilmişten maksat o yüzeyin normalinin bakış noktasına doğru olmamasıdır. Bu konuda detaylı bilgi Arkayüz Kaldırma konusunda verilecektir.

Gerek görünmeyen yüzeylerin kaldırılması gerekse de arkayüz kaldırma için **Işın İzleme (Ray Tracing)** yöntemi kullanılacaktır. Işın izleme yönteminde bakış noktasından çıkan ve görüntü düzlemindeki piksellerin her birinden geçen ışınlar ile 3D cisimler arasında kesişim testleri yapılır ve ışık kaynakları da dikkate alınarak ekrana görüntüsü çizilecek cismin renk değeri hesaplanır. Herhangi bir ışın birkaç tane cisim ile kesişiyorsa bunlardan bakış noktasına en yakın olanının çizilip diğerlerini atılması işlemine görünmeyen yüzeylerin kaldırılması denir.

2. Vektörel İşlemlerle İlgili Temeller

Işın vektörel bir büyüklük olduğu için vektörel işlemler hakkında bilgiler vermekte fayda vardır:

Herhangi bir vektörün boyunu bulmak için (x,y,z) koordinatlarının karelerinin toplamının karekökü alınır. Örneğin **R=(0,6,8)** vektörünün boyu |**R**|= $\sqrt{0^2+6^2+8^2}=10.$

Herhangi bir vektörün boyunu 1 birim yapma işlemine "normalizasyon" denir ve işlem için (x,y,z) koordinatlarının her biri vektörün boyuna bölünür. Yukarıdaki **(0,6,8)** vektörü normalize edildiğinde $\left(\frac{0}{10},\frac{6}{10},\frac{8}{10}\right)=(\mathbf{0},\mathbf{0.6},\mathbf{0.8})$ bulunur. Normalize edilmiş vektörün boyu hesaplandığında $\sqrt{(0)^2+(0.6)^2+(0.8)^2}=1$ olduğu görülür. Normalize edilmiş vektöre "birim vektör" denir. (Föy boyunca vektörel büyüklükler **koyu**, skaler büyüklükler normal font ile yazılacaktır).

Vektörler arasında vektörel ve skaler olmak üzere iki temel çarpım işlemi vardır. $\mathbf{R_1}$ ve $\mathbf{R_2}$ gibi iki vektörün sırasıyla vektörel ve skaler çarpımı aşağıdaki gibi yapılır:

$$\mathbf{R_1 \times R_2 = (R_{1y}R_{2z}-R_{1z}R_{2y},\ R_{1z}R_{2x}- R_{1x}R_{2z},\ R_{1x}R_{2y}- R_{1y}R_{2x})}$$

$$\mathbf{R_1 * R_2 = R_{1x}R_{2x} + R_{1y}R_{2y} + R_{1z}R_{2z} = |R_1|*|R_2|*Cos(\beta)}$$

Vektörel çarpım yüzey normalinin hesaplanmasında kullanılır. Yüzey normali yüzeye dik olan vektördür. İki vektörün skaler çarpımında (x,y,z) koordinatları çarpılıp toplanır veya vektörlerin boylarının aralarındaki açının kosinüsüyle çarpımı olarak da hesaplanabilir. Dolayısıyla skaler çarpımı yapılan vektörler birim vektör olursa bu vektörlerin skaler çarpımı aralarındaki açının kosinüsünü verir. Skaler çarpım Phong boyamada diffuse ve specular renk bileşenlerinin hesaplanmasında kullanılır. Vektörel çarpım vektörel; skaler çarpım skaler bir değer döndürür.

Köşe noktalarının koordinatları **V0,V1,V2** şeklinde verilen üçgenin yüzey normali **N** vektörü, **(V1-V0)** ve **(V2-V0)** vektörlerinin vektörel çarpımı ile şöyle hesaplanır:

V0=(0,40,120) V1=(30,-40,60) V2=(-30,-40,60)
(V1-V0)=(30,-80,-60) (V2-V0)=(-30,-80,-60)

N=(V1-V0)x(V2-V0)=(-80*-60 - -60*-80, -60*-30 - 30*-60, 30*-80 - -80*-30)

N = (0, 3600, -4800)

N vektörü normalize edilirse **(0,0.6,-0.8)** elde edilir.

3. Işının Tanımı ve Birincil Işınların Üretilmesi

Başlangıç noktası ve doğrultuya sahip vektörel bir büyüklük olan **R** ışını:

$$\mathbf{R} = \mathbf{R_o} + t\mathbf{R_d}$$

olarak ifade edilir. Burada $\mathbf{R_o}$ ışının başlangıç noktası, $\mathbf{R_d}$ doğrultusudur. t ise ışının 3D uzayda $\mathbf{R_d}$ doğrultusu boyunca kaç birim gideceğini belirleyen skaler bir değerdir. Doğrultu vektörü $\mathbf{R_d}$ ‘nin hesaplanabilmesi için 2 noktaya ihtiyaç vardır. Bunlar $\mathbf{R_1}$ ve $\mathbf{R_2}$ olarak alınırsa $\mathbf{R_1}$’den $\mathbf{R_2}$’ye doğru olan doğrultu vektörü $\mathbf{R_d=R_2-R_1}$ ile hesaplanır. Işın izlemede doğrultu vektörleri birim vektör olmalıdır. Dolayısıyla $\mathbf{R_d}$ normalize edilerek boyu 1 birim yapılır.

Işın izlemede ilk adım başlangıç noktasından çıkıp 3D görüntü düzlemindeki piksellerin herbirinden geçecek olan birincil ışınların doğrultularının belirlenmesi işlemidir. Bunun için piksel koordinatlarından başlangıç noktasının koordinatı çıkarılır. Ardından normalize edilerek doğrultunun boyu 1 birim yapılır. Işınlar ile 3D ortamdaki cisimler arasında kesişim testleri yapılarak görüntü düzlemine hangi cismin şeklinin çizileceği belirlenir.

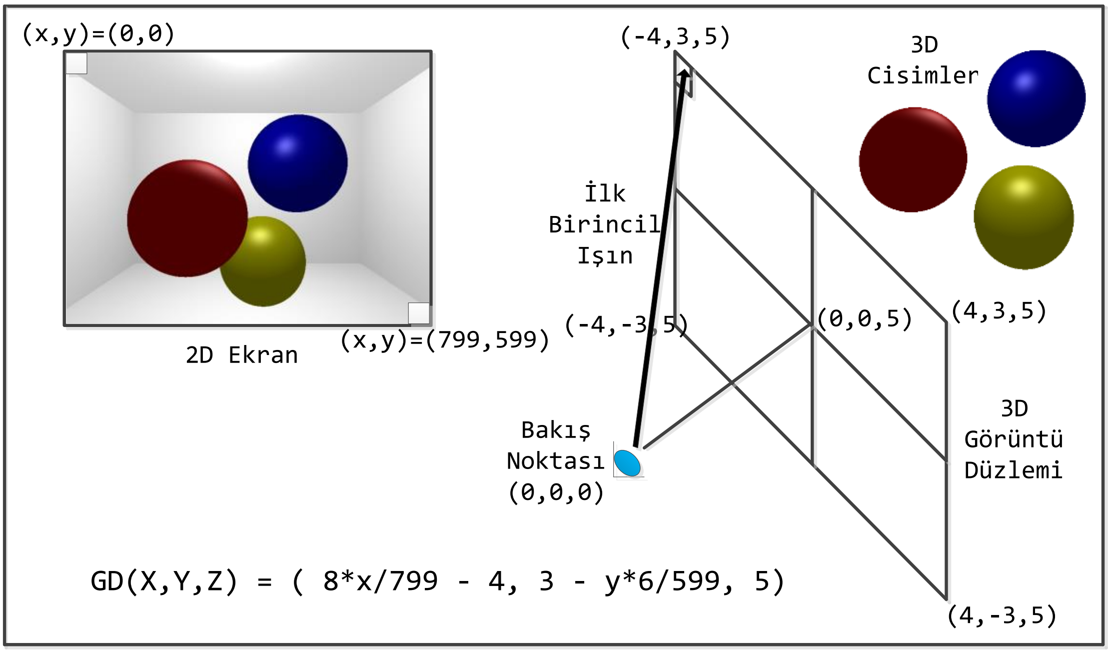
Şekil 1’den görüldüğü gibi ışınların 3D görüntü düzleminden geçtikleri pikseller ile en son ekranda üretilen görüntüdeki pikseller farklı koordinat sistemlerini kullanmaktadır. Örneğin bilgisayar ekranındaki 800x600 çözünürlükteki bir görüntünün (x,y) koordinatları sol üst köşede (0,0) sağ alt köşede de (799,599)’dur. Aynı çözünülürlükteki bir görüntü düzleminin sol üst köşesinin koordinatları (-400,300,500) olmalıdır (görüntü düzleminin bakış noktasına uzaklığı 500 birim alınmıştır). Dolayısıyla ekrandaki herhangi bir pikselin görüntü düzlemindeki karşılığını bulmak için bir dönüşüm yapmak gerekir. Ekrandaki 800x600 çözünürlüğünde bir görüntünün (x,y) koodinatlarının 3D Görüntü Düzlemi **GD(X,Y,Z)**'deki karşılığı aşağıdaki ifade ile bulunur:

GD(X,Y,Z) = (x - 399, 299 - y, 500)

Yukarıdaki dönüşümün pratikte kullanılması pek tercih edilmez. Çünkü çizilecek görüntünün çözünürlüğü değiştirilmek istenildiğinde sadece 399, 299 gibi değerleri değiştirmek yeterli olmaz. 3D cisimlerin koordinatlarını da değiştirmek gerekir. Dolayısıyla 3D cisimlerin koordinatlarından bağımsız dönüşüm imkanı sağlayan aşağıdaki ifade kullanılmalıdır:

GD(X,Y,Z) = (8*x/799 - 4, 3 - y*6/599, 5)

Burada görüntü düzlemi (8x6) boyutunda seçilmiştir. Üretilecek görüntünün çözünürlüğü 800x600'den farklı mesela 1024x768 olduğunda sadece 799'u 1023 ve 599'u 767 yapmak yeterlidir.



Şekil 1: 2D Ekran ve 3D Görüntü Düzlemi Arasındaki İlişki

4. Işın-Üçgen Kesişim Testi

3D cisimler çoğunlukla üçgenler ile temsil edilirler. Burada anlatılacak olan ışın-üçgen kesişim testi iki aşamadan oluşmaktadır:

- Işın ile üçgenin tanımladığı yüzey arasında kesişim testi.
- Işın yüzey ile kesişiyorsa kesişim noktasının üçgenin içinde olup olmadığını belirleme.

Birinci aşama için üçgenin tanımladığı yüzeyin denklemini çıkarmak gerekmedir. Bilindiği gibi yüzey denklemi Ax+By+Cz+D=0 ‘dır. Burada **(A,B,C)** yüzey normalidir. Yukarıda **V0,V1,V2** şeklinde verilen üçgenin yüzey denklemini çıkaralım:

Daha önce üçgenin normali **N=(0,0.6,-0.8)** olarak hesaplanmıştı. Dolayısıyla **N=(A,B,C)** biliniyor. Yüzeyin üzerinde olduğu için yüzey denklemini sağlayacağından üçgenin köşe noktalarından herhangi biri D ‘nin hesabı için kullanılabilir. Dolayısıyla Ax+By+Cz+D=0 ‘daki **(x,y,z)** yerine köşe noktalarından herhangi birinin mesela **V0** ‘ın **(x,y,z)** ‘sini yazıp sıfıra eşitlersek D ‘yi :

Ax + By + Cz + D = 0
0*0 + 0.6*40 + -0.8*120 + D = 0
D = 72

olarak buluruz. Dolayısıyla yüzey denklemi 0.6y - 0.8z + 72 = 0 ‘dır.

Işın yüzey ile kesişiyorsa üçgenin köşe noktalarında olduğu gibi ışının yüzey üzerindeki koordinatları da yüzey denklemini sağlamalıdır. Dolayısıyla şöyle yazabiliriz:

$A(\mathbf{R_{0x}}+t\mathbf{R_{dx}})+B(\mathbf{R_{0y}}+t\mathbf{R_{dy}})+C(\mathbf{R_{0z}}+t\mathbf{R_{dz}})=0$

Yukarıdaki denklem t ‘ye göre düzenlenirse:

$$t = -\frac{AR_{ox} + \mathrm{B}R_{oy} + CR_{oz} + \mathrm{D}}{AR_{dx} + AR_{dy} + AR_{dz}} = -\frac{N * R_o + D}{N * R_d}$$

t>0 ise ışın yüzey ile kesişiyor demektir. t<0 ise kesişmiyor, t=0 ise paralel demektir.

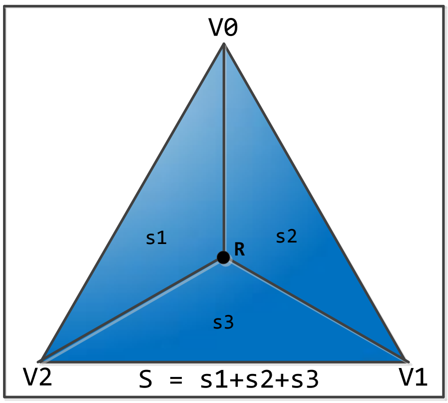
Yukarıda **V0,V1,V2** üçgeni için yüzey denklemi hesaplanmıştı. $\mathbf{R_o}$**=(0,0,0)** başlangıç noktasından **Rd=(0,0,1)** doğrultusu boyunca giden **R** ışınının bu yüzey ile kesişip kesişmediğini t değerini hesaplayarak belirleyebiliriz.

$$t = -\frac{N * R_o + D}{N * R_d} = -\frac{72}{-0.8} = 90$$

Işının yüzey üzerindeki koordinatları $\mathbf{R}=\mathbf{R_o}+t\mathbf{R_d}$**=(0,0,0)**+90**(0,0,1)=(0,0,90)** olarak bulunur. Böylece kesişim testinin I. Aşaması tamamlanmış oldu.

II. aşamada **(0,0,90)** noktasının üçgenin içinde olup olmadığına karar verilmelidir. Bunun için değişik yöntemler denenebilir. Burada “alan testi” yöntemi kullanılacaktır. Buna göre Şekil 2‘den de görüldüğü gibi yukarıda hesaplanan kesişim noktasından üçgenin köşelerine doğrular çizerek 3 alt üçgen oluşturulur. Bu alt üçgenlerin alanları toplamı büyük üçgenin alanına eşitse kesişim noktası üçgenin içinde demektir.

V0,V1,V2 üçgeninin alanı 0.5*|**(V1-V0)x(V2-V0)**| ile hesaplanabilir. **R=(0,0,90)** noktası kullanılarak oluşturulan s1, s2 ve s3 alt üçgenlerin alanları ve **V0,V1,V2** üçgeninin S alanını hesaplandığında S=3000, s1=750, s2=750 ve s3=1500 çıkar. S=s1+s2+s3 olduğundan kesişim noktası üçgenin içindedir.



Şekil 2: Alan Testi

5. Görünmeyen Yüzeylerin Kaldırılması (Hidden Surface Removal)

Giriş bölümünde de bahsedildiği gibi görünmeyen yüzeylerin kaldırılmasından maksat aynı ışın doğrultusu boyunca kesişen cisimlerden en yakın olanın belirlenip diğerlerinin atılması (kaldırılması) dır. Bunu bir örnekle açıklayalım:

U0(0, 30, 40)	**U1(40, -30, 120)**	**U2 (-40, -30, 120)**
Y0(-50, 30, 124)	**Y1(50, 30, 124)**	**Y2 (0, -30, 44)**
Z0(-30, 0, 37)	**Z1(30, 40, 117)**	**Z2 (30, -40, 117)**

Sırasıyla kırmızı, yeşil ve mavi renklere sahip **U0,U1,U2** üçgeni **Y0,Y1,Y2** üçgeni ve **Z0,Z1,Z2** üçgeninin köşe noktalarının koordinatları yukarıda verilmiştir. Başlangıç noktası **R_o=(0,0,0)** 'dan çıkan ve görüntü düzleminde **(0,0,5)** noktasındaki pikselden geçen ışın ile bu üçgenler arasında kesişim testleri yapılırsa t_U=80, t_Y=84, t_Z=77 uzaklık değerleri hesaplanır. Uzaklıklar sıralandığında mavi renkli Z üçgeninin bakış noktasına daha yakın olduğu görülür. Dolayısıyla ışının geçtiği piksel mavi renge boyanır. Kırmızı renkli U ve yeşil renkli Y üçgenleri görüntü düzleminde **(0,0,5)** noktasındaki pikselden görünmeyen üçgenlerdir.

6. Arkayüz Kaldırma (Backface Culling)

Giriş bölümünde de bahsedildiği gibi arkayüz olan yüzey (veya üçgen) bakış noktası ile arasında başka yüzeyler olmasa bile ters durduğu için görünmeyen yüzeydir. Yüzeyin veya üçgenin ters durması ne demektir?

Y=0 yüzeyini düşünelim. Bu yüzeyin +Y ve –Y eksenlerine bakan iki farklı yüzü vardır. Y=0 yüzeyi üzerinde bir üçgen tanımlayalım:

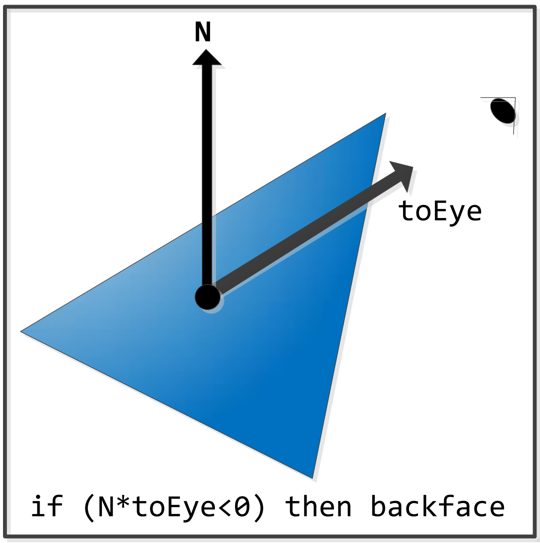
U0(0, 0, 40) **U1(40, 0, -40)** **U2 (-40, 0, -40)**

Bu üçgenin yüzey normali hesaplanırsa **(0, 6400, 0)** bulunur. Normalize edilirse **(0,1,0)** olur. Şimdi köşe noktalarının koordinatlarını farklı sırada yazalım:

U0(40, 0, -40) **U1(0, 0, 40)** **U2 (-40, 0, -40)**

Tekrar yüzey normali hesaplanırsa **(0, -6400, 0)** bulunur. Normalize edilirse bu sefer **(0,-1,0)** olur. İlk hesaplanan yüzey normali **(0,1,0)** +Y eksenine doğru şimdiki ise **(0,-1,0)** –Y eksenine doğru çıktı. Dolayısıyla köşe noktalarının sırası değişince yüzey normalinin doğrultusu da değişmektedir. Köşe noktaları 3D kartezyen koordinat sisteminde Z ekseni (0,0,0) 'dan ileriye doğru pozitif artıyor ise (+Z) saat yönünde (ClockWise-CW); negatif olarak artıyor ise (-Z) saat yönünün tersi sırada (CounterClockWise-CCW) tanımlanmalıdır. Bu kurallara sırasıyla sol el ve sağ el kuralı denir. Sol el kuralında sol elin 4 uzun parmağı +X eksenini gösterirken +Y eksenini gösterecek şekilde katlandığında baş parmağın doğrultusu +Z eksenini gösterir. Aynı işlem sağ el ile yapıldığında baş parmak yine +Z 'i gösterir. DirectX 3D kartezyen koordinatlar için + eksenleri sol el; OpenGL de sağ el kuralına göre belirler.

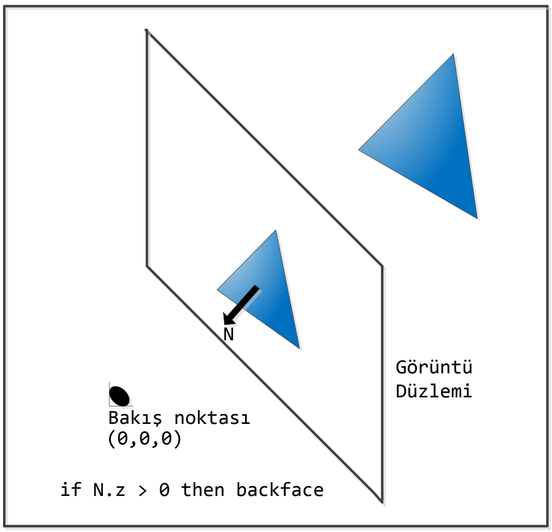
Yüzey normalinin doğrultusunun köşe noktalarının sırasına bağlı olarak değişmesi ile arkayüz olması arasında ne ilişki var? Bunu U üçgeni içindeki **(0,0,0)** noktasının diffuse katsayısını hesaplayarak açıklayalım:



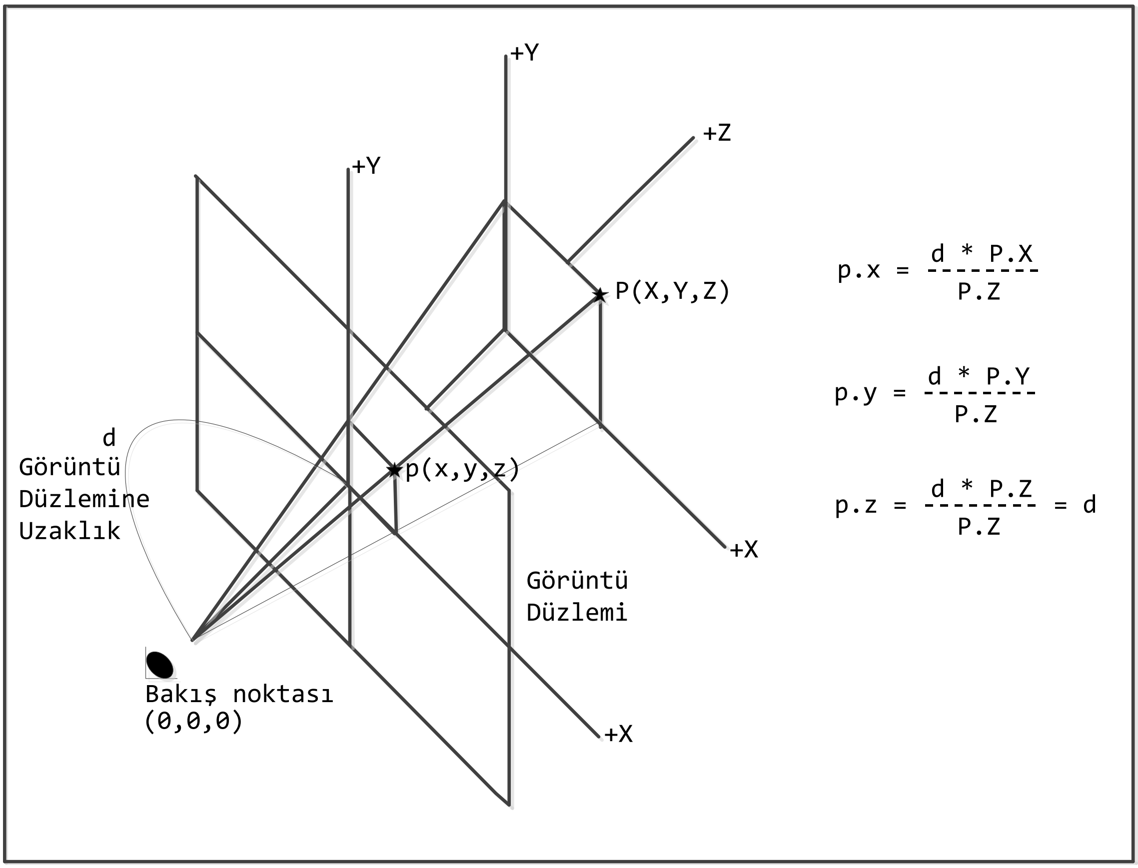
Şekil 3: Skaler çarpımla arkayüz kaldırma

Işık kaynağı **(0,60,80)** noktasında olsun. Işık kaynağına doğru olan vektör **(0,60,80)-(0,0,0)=(0,60,80)** normalize edilirse **(0,0.6,0.8)** bulunur. Diffuse kaysayıyı bulmak için bu vektör ile yüzey normali skaler çarpılırsa 0.6 bulunur. Aynı işlemler köşe noktalarının sırası değiştirilerek normali **(0,-1,0)** olan üçgen için yapılırsa bu sefer -0.6 bulunur. Renk değerinin negatif olması imkansız olduğundan -0.6 diffuse katsayı olarak kullanılamaz. Başka bir deyişle bu yüzey ışık kaynağı tarafından aydınlatılmıyor demektir. Demek ki yüzeyin ışık kaynağı tarafından aydınlatılıp/aydınlatılmadığına normalin doğrultusuna bağlı olarak karar veriyoruz. Benzeri şekilde yüzeyin bizim tarafımızdan görülüp görülmediğine de yine normaline göre karar veririz. Eğer bakış noktasına doğru olan vektör ile normalin skaler çarpımı negatif çıkıyorsa yüzeyin bakış noktasına göre görülmesi imkansız demektir. Yani arkayüzdür. Böylece arkayüz belirlemede kullanılan birinci yöntem açıklanmış oldu. Mesela yukarıda ikinci olarak tanımlanan U üçgeni bakış noktası **(0,60,80)** alındığında arkayüz olmaktadır. Bakış noktasına doğru olan vektör ile yüzey normalini skaler çarparak arkayüz kaldırma Şekil 3'te gösterilmiştir.

Arkayüz kaldırmada kullanılan diğer bir yöntemde üçgenin görüntü düzlemine izdüşümü alınır ve normali hesaplanır. Eğer normalin Z bileşeni >0 ise arkayüzdür. Yukarıdaki yöntemde skaler çarpma ile arkayüz belirlendiğinden "skaler çarpımla arkayüz kaldırma" olarak isimlendirildi. Burda da izdüşüm sonrası vektörel çarpımla normal hesaplandığından bu yöntem "vektörel çarpımla arkayüz kaldırma" olarak isimlendirilecektir. Vektörel çarpımla arkayüz kaldırma Şekil 4'te gösterilmiştir.



Şekil 4: Vektörel çarpımla arkayüz kaldırma



Şekil 5: Perspektif Dönüşüm ile İzdüşüm

Herhangi bir noktanın benzer üçgenler yardımıyla görüntü düzlemine perspektif dönüşüm ile izdüşümünün nasıl yapıldığı Şekil 5'te gösterilmiştir.

7. Deney Hazırlığı

- Görünmeyen Yüzeylerin Kaldırılması bölümündeki örnekte hesaplanan t değerlerini bulmak için gerekli ara işlemleri yazıp deneye getiriniz.
- Görünmeyen yüzey ile arkayüz arasında ne fark vardır? Karşımızda duran bir küpün görmediğimiz yüzeyleri görünmeyen yüzey midir yoksa arka yüz müdür?
- Vektörel çarpım ile arkayüz kaldırırken izdüşüm sonrası neden normalin X veya Y değil de Z bileşenine bakılır? Bu bileşenin neden sıfırdan küçük olması ile değil de büyük olması ile yüzey arkayüz olur?
- Önce hangisi kaldırılmalıdır? Görünmeyen yüzey mi arkayüz mü?
- Işın üçgen kesişim testinde kullanılan alternatif yöntemlerden biri de crossings testidir. Kaynak kodların olduğu klasördeki **CrossingTest.pdf** isimli belgeyi inceleyiniz.

8. Deney Tasarımı ve Uygulaması

V0(50, -30, 40)   V1(0, 30, 120)   V2(-50, -30, 40)

- Yukarıda köşe noktaları V0,V1,V2 olarak verilmiş üçgenin arkayüz olup olmadığını hem skaler çarpım hem de vektörel çarpım yöntemlerine göre belirleyiniz. Bakış noktası **(0,0,0)** 'dır. Yüzey üzerindeki nokta olarak köşe noktalarından herhangi birini alabilirisiniz. Hem skaler hem de vektörel çarpımda sonucun değeri değil işareti (pozitif mi negatif mi) önemli olduğundan yüzey normali ve bakış noktasına doğru olan vektörleri normalize etmenize gerek yoktur.
- Deneyde anlatılan arkayüz kaldırma, ışın-yüzey ve ışın-üçgen kesişim testlerinin kodlarını yazınız.
- Işın üçgen testinde noktanın üçgenin (poligonun) içinde olup olmadığının belirlenmesinde kullanılan en hızlı tekniklerden biri de crossings testidir. Bu yöntemde, noktadan herhangi bir doğrultuda ışın gönderilir. Eğer poligonla tek sayıda kesişirse poligonun içinde, çift sayıda kesişirse de dışında olduğuna karar verilir. Bunun için size verilen **CrossingTest.pdf** isimli belgede yalancı dilde yazılmı algoritmayı inceleyerek mantığını anlamaya çalışınız. İlgili algoritmayı kodlayarak programı test ediniz.