

Görüntüleyici: Bakış Açısına Göre Basitleştirme Yapan Bir Model Görüntüleme Sistemi *

Okan Arıkan [†] ve Uğur Güdükbay [‡]

[†] University of California
Department of Electrical Engineering and Computer Science
Computer Science Division
387 Soda Hall 1776
Berkeley, CA 94720-1776, USA

[‡] Bilkent Üniversitesi
Bilgisayar Mühendisliği Bölümü
06533 Bilkent Ankara

okan@cs.berkeley.edu, gudukbay@cs.bilkent.edu.tr

Özetçe

Gelişigüzel üçgenel yüzey modelleri, üçgenlerden oluşan ve hiçbir şekil ve sınır kısıtlaması olmayan modellerdir. Aşamalı yüzey modeli, üçgenel modellerin değişik detay seviyelerindeki versiyonlarını tanımlayan çok çözünürlüklü modelleme yöntemidir. Aşamalı yüzey modeli işlenerek modelin orjinal hali ile en basit hali arasında bir versiyonu elde edilebilir. Aynı zamanda, aşamalı yüzey modeli bakış açısına göre basitleştirilerek görüntü kalitesi orjinal modele yakın basitleştirilmiş modeller elde edilebilir. Bu makalede, poligonal modellerin bakış açısına göre basitleştirilerek gösterilmesini sağlamak amacı ile geliştirilen bir yüzey gösterme sistemi anlatılmaktadır.

1 Giriş

Bilgisayar grafiğinde modeller üç boyutlu tarayıcılar kullanılarak ya da otomatik olmayan yöntemlerle oluşturulmaktadır. Bu şekilde oluşturulan modeller çok sayıda primitif yüzeyler (poligonlar, spline yüzeyler, vb) içerebilmektedirler. Bu primitiflerin bazıları matematiksel olarak çok karmaşık bir şekilde modellendiklerinden bilgisayar kullanılarak kolay bir şekilde gösterilememektedir. Bu nedenle, primitifler poligonal yüzeylere dönüştürülerek gösterilmektedir. Bu amaçla, bu modeller için düzlemsel poligon (genellikle üçgenler) kümeleri oluşturulmaktadır. Poligonal yüzeylerin doğrusal yapısı, bu modellerin gösterimi ve geometrik olarak işlenmesi için büyük kolaylıklar sağlamaktadır. Bunun bir nedeni de poligon boyama genellikle günümüzde kullanılan bilgisayarlarda donanımda yapılmaktadır. Bu da böyle modellerin hızlı bir şekilde boyanmasına imkan sağlamaktadır.

Poligonal modeller temel olarak köşeler ve poligonal yüzeylerden oluşmaktadır. Her poligonal yüzey bu köşelerden bir kaçını içermektedir. Bu da modelin pozisyon ve geometrisini belirlemektedir. Poligonal yüzeylerin düzlemsel olması kolay işlenebilirlik açısından gereklidir. Her poligonal yüzey istenildiği kadar köşeye sahip olabilir. Ancak, modellerde sadece üçgenel yüzeyler kullanmak sıklıkla kullanılan bir yöntemdir. Bunun nedeni, üçgenel yüzeylerin düzlemsel ve konveks yapısı bu yüzeyleri gerçek zamanlı boyama için en uygun yüzeyler haline getirmektedir. Üç köşeden daha fazla köşe içeren yüzeyler, üçgenleştirme algoritmaları kullanılarak üçgenel modellere dönüştürülebileğinden bu durum sunulan yöntemlerin genelliğinden birşey kaybettirmemektedir.

*Bu araştırma projesi Türkiye Bilimsel ve Teknik Araştırma Kurumu (TÜBİTAK) tarafından EEEAG 198E018 proje numarası ile desteklenmektedir. Qs1im isimli yüzey basitleştirme programı için Michael Garland'a teşekkür ederiz. Tavşan ve Buda heykeli modelleri Stanford Üniversitesi Bilgisayar Grafiği Laboratuvarı'ndan sağlanmıştır.

Yüzey basitleştirme, bir yüzey modelinin köşe ve poligon sayısını yüzeyin genel geometrisine sadık kalarak azaltma işlemi olarak tanımlanabilir. Yüzey basitleştirme yöntemlerinden bir tanesi bu proje için diğerlerine göre daha önemlidir: *kenar yokederek basitleştirme*. Bu yöntemde, her safhada bir kenar seçilerek bu kenar bir köşeye indirgenmektedir. Her safhada, seçilen kenar yüzeyden atılarak yerine yeni bir poligon köşesi konulmakta ve poligonlar da buna göre güncellenmektedir. Bu işlem yeni poligon köşesi yerine yokedilen kenar tekrar yerine konularak geri alınabilir. Bu işleme de *köşe ayırma* adı verilmektedir. Sonuç olarak eğer yokedilen kenar içeride bir kenar ise her safhada iki poligon yüzeyi (üçgen) ve bir poligon köşesi poligonal yüzeyden atılmaktadır.

Bazen bir poligonal modelin sadece bir versiyonunu kullanmak işlemci zamanının gereksiz şekilde harcanmasına neden olabilir. Örneğin, ekranda gösterilecek model kameradan çok uzakta ve sadece birkaç piksellik bir alanı kapsıyorsa modelin basit bir versiyonu ekranda gösterilerek performans arttırılabilir. Aynı modelin farklı detay seviyelerindeki versiyonlarını değişik durumlarda kullanmak sıklıkla başvurulan bir yöntemdir. Bu yöntemde *çok çözünürlüklü modelleme* adı verilmektedir.

Etkileşimli uygulamalarda kullanılan diğer bir yöntem ise modelin boyama işlemi sırasında dinamik olarak istenilen detay seviyesine indirgenmesidir. Bu amaçla geliştirilmiş olan yöntemlerden bazıları modelleri bakış açısına göre seçici olarak basitleştirmekte, böylece modelin bazı kısımları diğer kısımlara göre daha fazla basitleştirilerek istenilen görsel kalite elde edilmekte ve işlem zamanından tasarruf sağlanmaktadır. Örneğin, modelin bir kısmı kameranın görüş alanı dışında kalabilmekte ya da bakış açısına göre başka noktalara bakabilmektedir. Bu durumda, bu kısımlar görüntüye fazla bir katkıda bulunmadığından basitleştirilerek gösterilmektedir. Aynı zamanda, modelin kameraya yakın olan kısımlarının diğer kısımlara göre daha detaylı gösterilmesi gereklidir. Kullanıcının çok fazla dikkat etmediği bölgelerin detay seviyesi azaltılabilir.

Makalenin devamı şu şekilde organize edilmiştir. İkinci kısımda sistemin genel yapısı ve bakış açısına göre model arındırma anlatılmış, gerçekleştirme detaylarından bahsedilmiştir. Üçüncü kısımda sistemin kullanımı ile elde edilen görüntüler sunulmuştur. Dördüncü kısım ise sonuçları vermektedir.

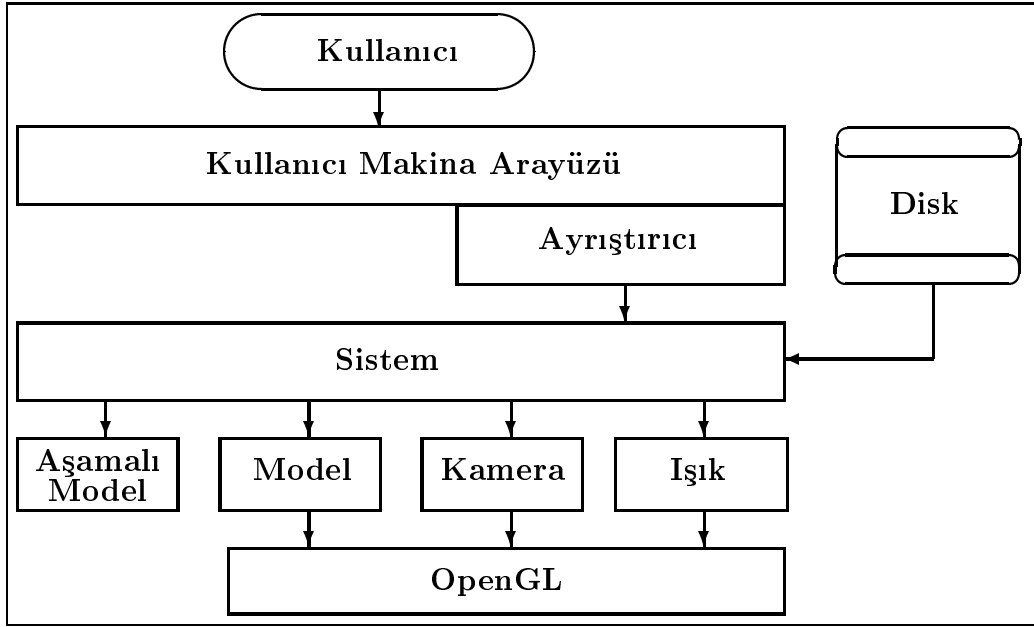
2 Sistemin Genel Yapısı

Görüntüleyici adı verilen sistemin genel yapısı Şekil 1'de verilmektedir. Sistem, temel olarak Hugues Hoppe [3] tarafından geliştirilen bakış açısına göre basitleştirme yöntemini kullanmaktadır. Aşamalı yüzey modeli oluşturmak için Michael Garland [1] tarafından geliştirilmiş olan yüzey basitleştirme algoritması kullanılmaktadır. Sistem temelde OpenGL adı verilen grafik yazılım kütüphanesi üzerine kurulmuştur. Sistemi oluşturan temel nesnel veri yapıları şunlardır:

- **Işık:** Işık kaynaklarının bilgisini tutar.
- **Kamera:** Kamera ile ilgili bilgilerin tutulması ve üzerindeki işlemlerin yürütülmesini sağlar.
- **Model:** Bu veri yapısı bir poligonal modelin depolanması ve ekrana çizilmesini sağlar.
- **Aşamalı Model:** Bu veri yapısı bir modelin aşamalı model yapısını depolar. Bakış açısına göre arındırma ve bakış açısından bağımsız arındırma işlemleri için gerekli metodları sağlar.
- **Sistem:** Bu veri yapısı diğer yapılar için bir platform teşkil eder ve programın tamamının çalışmasını kontrol eder.
- **Kullanıcı Makina Arayüzü:** Kullanıcı arayüzünü tanımlar ve veri alma ve çıktı üretmekten sorumludur. Ayrıca, kullanıcıdan komut konsolu aracılığı ile alınan komutları parçalayarak anlamaya yarayan **Ayrıştırıcı** altsistemi de kullanıcı makina arayüzünün önemli bir kısmıdır.

2.1 Bakış Açısına Göre Arındırma

Bir poligonal yüzeyin aşamalı model yapısı, bir temel yüzey ve bu yüzeye uygulanacak olan köşe ayırma işlemlerinin kümesi olarak tanımlanabilir [2]. Köşe ayırma işlemleri, sıra ile basit modele uygulandığında orjinal model elde edilir. Bu işlemler bakış açısına göre seçici bir şekilde de uygulanabilir. Böylece sadece modelin görünen kısımlarında detay seviyesi arttırılabilir. Arındırma (detaylandırma ya da basitleştirme) hızlı bir şekilde aşamalı model yapısı kullanılarak yapılabilir. Arındırma sırasında bir köşeyi ayırmak (detaylandırmak) için aşağıdaki üç kritere bakılır [3, 4]:



Şekil 1: Görüntüleyici sisteminin genel yapısı

```

procedure view_dependent_refine()
  for each i in {1 .. number_of_pcols}
    if (pcol[i].v1.active) and (pcol[i].v2.active)
      if refine(pcol[i])
        Execute(pcol[i])
  
```

Şekil 2: Bakış açısına göre arındırma algoritması

- **Bakış Piramidi:** Eğer bir köşe ve bu köşe ayırma işlemine tabi tutulduğunda elde edilecek köşelerin tümü bakış piramidinin dışında kalıyorsa, bu köşenin ayırma işlemine tabi tutulması elde edilecek görüntüye bir katkıda bulunmaz. Buna karar vermek için bu köşe ve ayırma işlemi sonucunda elde edilecek köşeleri içeren ve bu köşeyi merkez kabul eden küre bakış piramidinin kenarları ile karşılaştırılır. Eğer küre piramidin dışında ise köşe ayırma işlemi uygulanmaz.
- **Yüzey Oryantasyonu:** Eğer bir köşe ve bu köşe ayırma işlemine tabi tutulduğunda elde edilecek köşelerin tümü bakış açısının tersi yöne bakıyorsa, bu köşenin ayırma işlemine tabi tutulması da elde edilecek görüntüye bir katkıda bulunmaz. Bunu anlamak için sözkonusu köşenin normalini eksen kabul eden ve ayırma işlemi sonucunda elde edilecek köşelerin normallerini kapsayan bir piramid hesaplanarak bakış açısı ile karşılaştırılır.
- **Ekran Uzayı Hata Toleransı:** Eğer bir köşe ayırma işlemi uygulandığında elde edilecek görüntüde belirgin bir sapma oluyorsa, işlemin uygulanması modelin geometrisinin değiştiği izlenimini verebilir. Bu durumda, sapmanın toleransını dinamik olarak değiştirerek detay seviyesi ile ekrandaki hata oranı kontrol edilebilir. Bu kriter modelin bakış noktasına olan uzaklığını da (dolayısı ile modelin ekranda kapladığı alanı) dikkate alır.

Kullandığımız bakış açısına göre arındırma algoritması Şekil 2’de verilmektedir. Arındırma fonksiyonunu uygulamak için köşe çifti yoketme kayıtları (`pcol`) sıra ile taranmaktadır. Yoketme işlemine tabi tutulacak köşe çiftine ait köşeler aktif ise arındırma fonksiyonu (`refine`) yukarıda anlatılan üç kriteri kontrol ederek yoketme işleminin uygulanıp uygulanmayacağına karar vermektedir. Her köşe çifti yoketme işlemi sırasında iki köşe yokedilmekte ve bu iki köşenin yerine bir köşe yerleştirilmektedir. Orijinal modeldeki köşe sayısı çok yüksek olduğundan, arındırma işlemi için gerekli önceden hesaplanmış veriler köşe çifti yoketme kayıtlarında tutulmaktadır.

2.2 Gerçekleştirme Detayları

Aşamalı model yapısını oluşturmak için kullandığımız model basitleştirme yöntemi temel olarak köşe çiftlerini yoketmeye yönelik bir yöntemdir. Bu yöntemde bir köşe çiftini yoketmek için aralarında kenar olma zorunluluğu olmadığından, basitleştirme yapıldığında birbirine bağlı olmayan kısımlar birleşebileceğinden modelin topolojisi değişebilmektedir. Bu özellik, bakış açısına göre basitleştirmede aranılan bir özellik olmaktadır. Örneğin, model birbirlerine bağlı olmayan birkaç parçadan oluşuyorsa ve bunların bazıları bakış piramidinin dışında kalıyorsa bu parçalar tamamen atılabilmektedir.

Gerçekleştirdiğimiz sistemde aşamalı model yapısı olarak orjinal model ile bunu basitleştirmek için gerekli köşe çifti yoketme işlemleri tutulmaktadır. Bu yapı, basit model ile onu detaylandırmak için gerekli köşe ayırma işlemlerini tutan orjinal aşamalı model yapısından farklı olup bakış açısına göre basitleştirme/detaylandırma işlemi sırasında basit modele köşe ayırma işlemlerini uygulayarak detaylandırma yerine orjinal modele köşe çifti yoketme işlemleri uygulanarak basitleştirme yapılmaktadır. Bu durumda hafıza kullanımı artmakla birlikte şu kazanımlar elde edilmektedir. Bizim kullandığımız yöntemin uygulaması daha basit ve hızlıdır. Hoppe'un yönteminde [3], her köşe ayırma işlemi sırasında iki yeni köşe oluşmakta ve bir köşe yokedilmektedir. Ayırma işlemine tabi tutulacak olan köşeleri tespit etmek amacı ile arındırma fonksiyonunun aktif köşe listesini taraması gerekmektedir. Bu durumda bazı köşeler birden fazla defa arındırma işlemine tabi tutulabilmektedir. Ayrıca, arındırma fonksiyonu bazı koşulları kontrol etmek zorunda olduğundan daha karmaşıktır ve köşe ayırma ve poligon veri yapıları bu kontroller için daha fazla bilgi tutmak durumundadır. Bizim kullandığımız yöntemde, poligonal model üzerinde karmaşık geometrik kontroller yapmaya gerek olmadığından ve ortaya çıkan köşelerin hangi yüzlere ait olduğunu belirleme ihtiyacı olmadığından model daha hızlı gösterilebilmektedir.

3 Deneysel Sonuçlar

Bu kısımda sistemin çalıştırılması ile elde edilen görüntüler sunulmaktadır. Sistemin çalıştırılması ile elde edilen bakış açısına bağlı görüntüleme animasyonları değişik animasyon formatlarında http://www.cs.bilkent.edu.tr/~gudukbay/view_dependent.html adresindeki web sitesinden görülebilir.

Bu kısımdaki görüntülerde modellerin sadece bir kısmını içeren dikdörtgensel alanlar ya da piramid şekilleri, fantom kamera (ikinci bir bakış açısı) için bakış piramidini (diğer bir deyişle bu kameranın görüş alanını) tanımlamaktadır. Bunun amacı bakış açısına göre arındırma işleminin etkisini göstermektir. Eğer kullanıcı ikinci bir kamera tanımlamazsa bakış açısına bağlı basitleştirme görüntü düzlemi tüm pencere olan normal bakış açısına göre yapılmaktadır.

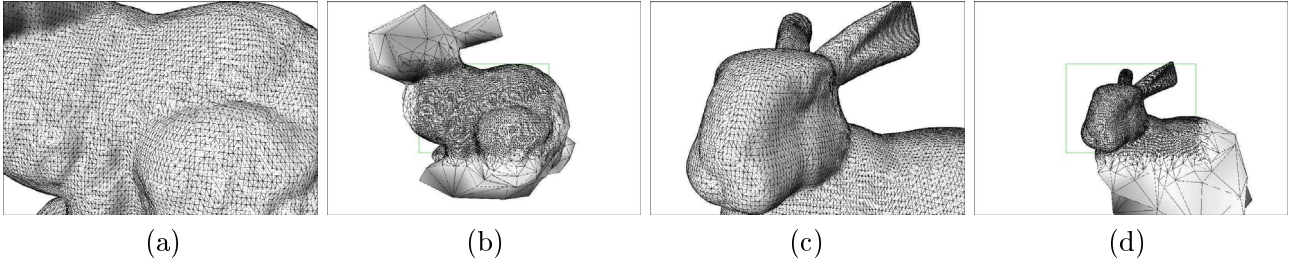
Şekil 3 bakış piramidinin bakış açısına göre arındırma işlemindeki etkisini göstermektedir. Şekil 3'de görüldüğü gibi, piramidin dışında kalan alanlar detaysız gösterilmektedir. Orjinal model 70000 poligon (üçgen), Şekil 3 (a) ve (b)'deki versiyonlar 32558 üçgen, (c) ve (d)'deki versiyonlar 32330 üçgen içermektedir. Sadece piramidin dışında kalan alanlar detaysız gösterildiğinden, normal kamera bakış açısına göre model gösterildiğinde orjinal modelle arasında görüntü kalitesi açısından fark yoktur.

Şekil 4 yüzey oryantasyonunun bakış açısına göre arındırma işlemindeki etkisini göstermektedir. Şekil 4'de görüldüğü gibi, modelin bakış noktasının tersi yönüne bakan kısımları detaysız olarak gösterilmektedir. Şekildeki arındırma işlemi modeldeki üçgen sayısını 70000'den 28949'a düşürmüştür. Basitleştirilmiş modelle orjinal model arasında görüntü kalitesi açısından fark yoktur.

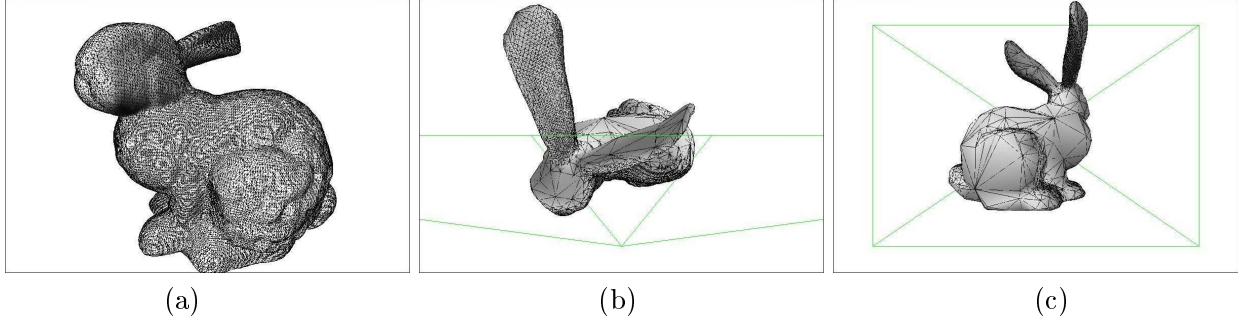
Şekil 5 ekran uzayı hata toleransının bakış açısına göre arındırma işlemindeki etkisini göstermektedir. Modelin Şekil 5 (a), (b) ve (c)'deki versiyonları sırası ile 1657, 7385 ve 13292 üçgen içermektedir. Modelin silüetini belirleyen kenar kısımlarında detay seviyesi diğer kısımlara göre yüksektir. Bunun nedeni ekran uzayı hata toleransının bu kısımlardaki etkisinin yüksek olmasıdır.

Şekil 6 basitleştirme işlemi sırasında gerektiğinde modelin topolojisinin değişebileceğini göstermektedir. Şekil 6 (a)'daki dört parçadan oluşan model topolojiyi koruyan bir algoritma ile (b)'deki modele dönüşürken, bizim kullandığımız algoritma ile (c)'deki tek parçalı model elde edilmektedir. Bu da daha basit modellerin elde edilebilmesine imkan sağlamaktadır. Orjinal model birbirine bağlı olmayan parçalardan oluşuyorsa, görüntüye katkısı olmayan parçalar tamamen yokedilebilmektedir.

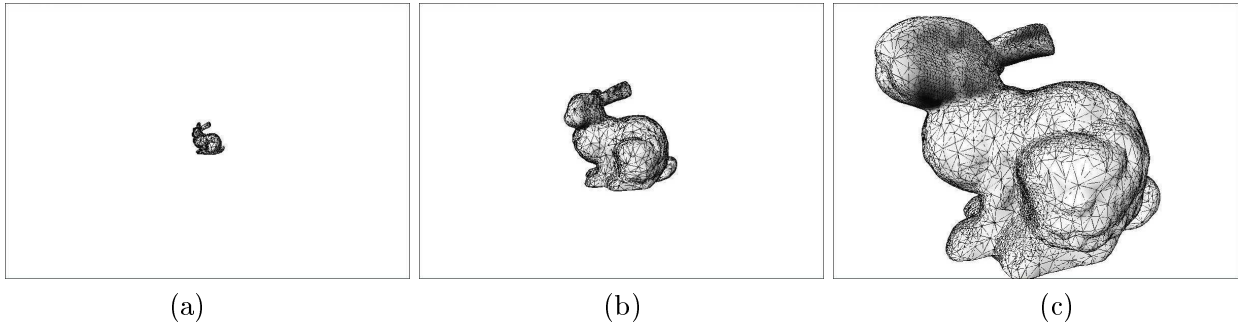
Şekil 7 sistemin bir Buda heykeli modelini görüntülemesini vermektedir. Sistem 1088000 üçgen içeren bu modeli rahatlıkla gerçek zamana yakın hızlarda görüntüleyebilmektedir.



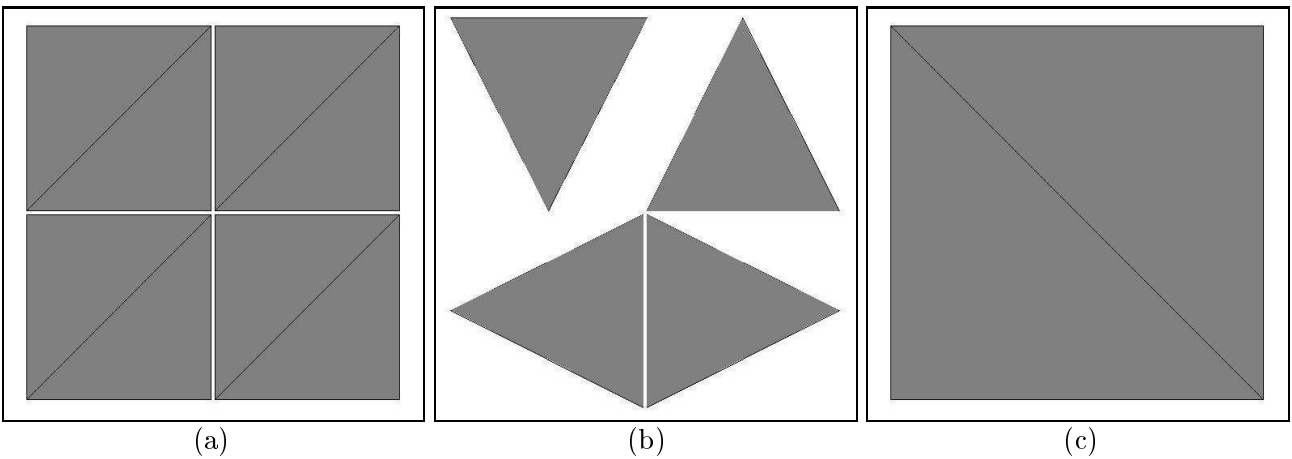
Şekil 3: Bakış piramidinin bakış açısına göre arındırma işlemindeki etkisi.



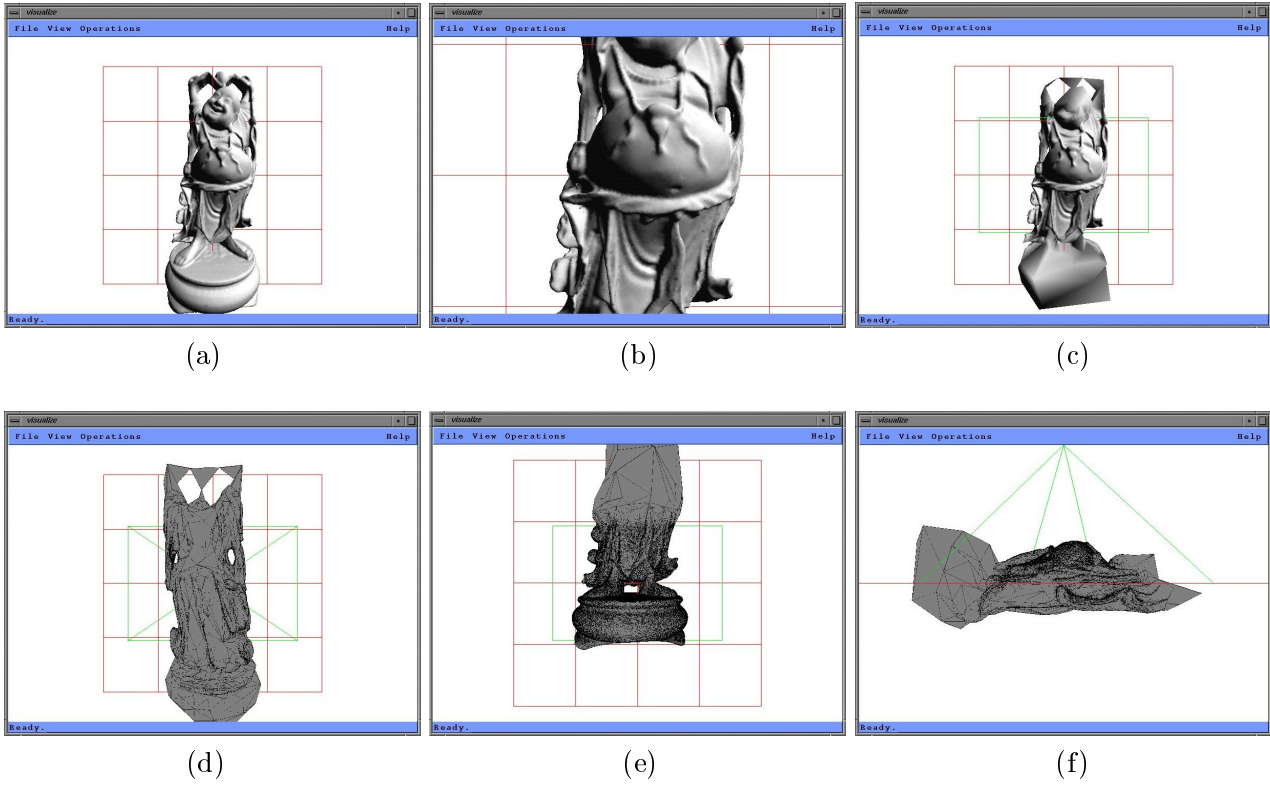
Şekil 4: Yüzey oryantasyonunun bakış açısına göre arındırma işlemindeki etkisi.



Şekil 5: Ekran uzayı hata toleransının bakış açısına göre arındırma işlemindeki etkisi.



Şekil 6: Topolojinin değiştirilebilmesinin basitleştirme işlemine etkisi.



Şekil 7: Buda heykelinin *Görüntüleyici* sistemi ile görüntülenmesi.

4 Sonuçlar

Bu makalede aşamalı model yapılarının bakış açısına göre arındırılması için geliştirilmiş bir sistem anlatılmaktadır. Sistem, temel olarak Hugues Hoppe tarafından geliştirilen bakış açısına göre basitleştirme yöntemini kullanmaktadır. Aşamalı yüzey modeli oluşturmak için modelde topolojik değişikliklere olanak veren bir yüzey basitleştirme yöntemi kullanılmaktadır. Bu yöntem kenarlar yanında aralarında kenar bulunmayan birbirlerine yakın köşe çiftlerini de yok edebilme özelliğine sahiptir. Bu nedenle, basitleştirme işlemi sırasında modelin topolojisi değişebildiğinden birkaç parçadan oluşan modellerin bazı parçaları basitleştirme sırasında diğer parçalarla birleşerek kaybolabilmektedir.

Gerçekleştirdiğimiz sistemde aşamalı model yapısı olarak orjinal model ile bunu basitleştirmek için gerekli köşe çifti yoketme işlemleri tutulmaktadır. Bu yapı ile bakış açısına göre arındırma işlemi sırasında basit modele köşe ayırma işlemlerini uygulayarak detaylandırma yerine orjinal modele köşe çifti yoketme işlemleri uygulanarak basitleştirme yapılmaktadır. Bu durumda yüzeylerin komşuluk ilişkileri ile ilgili karmaşık geometrik kontrollere gerek kalmamakta ve arındırma işlemi basitleştigiğinden daha hızlı yapılabilmektedir.

Kaynakça

- [1] Garland M. ve Heckbert P.S., “Surface simplification using quadric error metrics”, *ACM Computer Graphics (Proc. of SIGGRAPH’96)*, Ağustos 1996, s. 209–216.
- [2] Hoppe H., “Progressive meshes,” *ACM Computer Graphics (Proc. of SIGGRAPH’96)*, Ağustos 1996, s. 99–108.
- [3] Hoppe H., “View-dependent refinement of progressive meshes,” *ACM Computer Graphics (Proc. of SIGGRAPH’97)*, Ağustos 1997, s. 189–198.
- [4] Luebke D. ve Erikson C., “View-dependent simplification of arbitrary polygonal environments”, *ACM Computer Graphics (Proc. of SIGGRAPH’96)*, Ağustos 1996, s. 199–208.