



YENİ GRAFİK UYGULAMA GELİŞTİRME ARAYÜZLERİNİN KARŞILAŞTIRILMASI

Mustafa TÜRKSEVER, Aybars UĞUR

Ege Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, 35100-Bornova/İzmir

Geliş Tarihi : 16.04.2001

ÖZET

Bu makalede grafik yazılımları geliştirmede kullanılan uygulama geliştirme arayüzleri sınıflandırılarak kısaca tanıtılmıştır. İki boyutlu ve üç boyutlu uygulama geliştirme arayüzlerinin özellikleri belirtilmiştir. Çalışma kapsamında geliştirilen aynı işleve sahip iki yazılım (Kup3B Java2D sürümü, Kup3B Java3D sürümü) kaynak kod satır sayılarına göre karşılaştırılmışlardır. Elde edilen değerler grafiklere dönüştürülmüş ve yorumlanmıştır.

Anahtar Kelimeler : Grafik yazılımları, Yazılım geliştirme, Uygulama geliştirme arayüzleri, Java2D API, Java 3D API

COMPARISON OF NEW GRAPHICS APPLICATION PROGRAMMING INTERFACES

ABSTRACT

In this paper, Application Programming Interfaces (API) which are used for graphics software development are classified and introduced briefly. Properties of 2D and 3D API's are specified. Programs which are implemented in this study (Kup3B Java2D version, Kup3B Java3D version) are compared by number of lines of code written. Charts are constructed by using values obtained and interpreted.

Key Words : Graphics software, Software development, Application programming interfaces, Java 2D API, Java 3D API

1. GİRİŞ

Bilgisayar yazılım ve donanımlarında son yıllarda görülen gelişmeler, bilgisayar grafikleri ve animasyon işlemlerindeki kalite ve hızı oldukça artırmıştır. Bu şekilde günlük yaşamdan değişik bilim dallarına kadar her alanda etkisini göstermeye başlayan bilgisayar grafikleri alanına duyulan ilgi ve gereksinim giderek artmakta; bu alan standartların oluşması ve yeni kavramların eklenmesi ile her geçen gün zenginleşmektedir.

İnternet üzerine üç boyutlu grafiksel içerik ekleme, son günlerde üzerinde çalışılan konulardandır. İnternet2 ve NGI (Next Generation Internet) gibi ileri ağ teknolojilerini oluşturmaya yönelik

projelerden elde edilecek başarılarla, daha yüksek bant genişliği, servis kalitesi sağlanacak ve çok kullanıcı, etkileşimli, gerçek zamanlı uygulamaların beklediği ortam oluşacaktır. Sonuçta üç boyutlu grafiksel içerikler oluşturmayı ve üzerinde işlemler yapmayı sağlayan yazılımların ve yazılım geliştirme ortamlarının önemi daha da artacaktır.

Grafik yazılımları ikiye ayrılmaktadır :

- Grafik Uygulama Geliştirme Arayüzleri (API'ler)
- Grafik Paket Programları (Çizim, boyama, görüntü işleme, modelleme, animasyon, ışın izleme, CAD, oyun, eğitim ve diğerleri)

Üç boyutlu grafiksel yazılımların geliştirilmesinde, kullanılan grafik uygulama geliştirme arayüzünün (API) türü yönünden iki yaklaşım vardır. İki boyutlu bir arayüz kullanıp üç boyutla ilgili kodları (üç boyutlu uzayda nesnelere taşıma, üç boyutlu nesnelere köşe ve yüzey bilgilerini tutarak oluşturma, perspektif izdüşüm gibi), yazılacak uygulamaya yönelik olarak baştan oluşturmak veya doğrudan bu özellikleri sağlayan (hazır gelen) üç boyutlu bir arayüz kullanmak.

Bu çalışmada,

- Aynı özelliklere sahip iki adet program yazılmıştır. Her iki program da, üç boyutlu bir küpün oluşturulmasını ve etkileşimli olarak hareket ettirilmesini sağlamaktadır. İlk programda sadece iki boyutlu bir arayüz olan Java 2D kullanılmış ve üç boyut, matematiği, algoritmaları ve veri yapıları tasarlanıp kodlanarak oluşturulmuştur. İkinci programda, üç boyutlu bir arayüz olan Java 3D kullanıldığından, ve Java 3D ile hazır üç boyut matematiği, algoritmaları, veri yapıları ve üç boyutlu nesne oluşturma komutları geldiğinden, sadece içeriğin oluşturulması, programın tamamlanması için yeterli olmuştur.
- Bu iki yazılım, kaynak kod satır sayılarına göre karşılaştırılmıştır. Böylece, yazılımlarda üç boyutlu grafik kullanımının giderek yaygınlaştığı günümüzde, üç boyutlu grafikler içeren yazılımların geliştirilmesinde, üç boyutlu API kullanımının yazılım geliştirme sürecini ne kadar hızlandırdığı ve kolaylaştırdığı belirlenmiştir. Çalışmada elde edilen değerlerin önemini, yazılım geliştirme maliyetlerine de yansıtacağı dikkate alındığında büyük olduğu açıktır.

2. GRAFİK UYGULAMA GELİŞTİRME ARAYÜZLERİ (API)

Grafik uygulama geliştirme arayüzleri (API), iki bölümde incelenebilir. İki boyutlu grafik API'leri ve üç boyutlu grafik API'leri. İki boyutlu ve üç boyutlu grafikler içeren uygulamaların yazılmasında bu arayüzlerin kullanımı büyük olanaklar ve kolaylıklar getirmektedir.

2. 1. İki Boyutlu grafik API'leri

Grafik yazılımları geliştirmede iki boyutlu grafik uygulama geliştirme arayüzleri uzun yıllardır kullanılmaktadır. İki boyutlu grafikler içeren

yazılımların geliştirilmesinde kullanılmak üzere yeterli derecede işlev ve özellik içermektedirler. Ayrıca üç boyutlu grafikler içeren yazılımlarda da kullanılmaktadırlar.

Kişisel bilgisayarlarda DOS işletim sistemi ortamında Pascal, C, C++ programları ile kullanılmış olan BGI (Borland Graphics Interface) ve Windows ortamındaki Windows API'leri birçok grafik programcısının bildiği arayüzlerdir. Günümüzde Java awt ve Java 2D API de yaygın olarak kullanılmaya başlanmıştır. Java programlama dilinin ağ merkezli olmasının getirdiği avantajlardan da yararlanılmaktadır. Yazılan grafiksel içerikli bir applet, doğrudan internet üzerine yerleştirilebilmektedir. Dünyanın farklı bölgelerindeki kullanıcılar internet üzerinden erişerek bu programları kullanabilmektedirler.

Günümüzde grafik uygulama geliştirme arayüzlerinin özellikleri genişlemiştir. İki boyutlu bir grafik uygulama geliştirme arayüzünün özellikleri aşağıdaki gibi belirtilebilir (Knudsen, 1999) :

- **Temel Şekiller** : Temel iki boyutlu şekilleri çizdirmeyi sağlayan komutlar (nokta, çizgi, dikdörtgen, elips, yay, eğri, çokgen vb.) vardır. Diğer geometrik şekiller, temel şekillerin bileşiminden oluşur.
- **Renk** : Değişik amaçlarla hazır renklerin kullanımını, renklerin elde edilmesini ve renk özelliklerinin ayarlanmasını sağlayan renk komutları kullanılır.
- **Resim** : BMP, GIF, JPEG dahil birçok resim formatındaki resmi yüklemeyi ve kaydetmeyi sağlayan komutlar yardımıyla, herhangi bir grafik programı ile veya tarayıcıdan alınarak oluşturulmuş bir resim grafik programlarında kullanılabilir hale gelmektedir.
- **Çizgi ve Dolgu Biçemi** : Şekilleri biçemlemeyi sağlayan komutlar ve hazır özellikler, nesnelere çeşitlilik katmaktadır.
- **Çizgi biçemi** : Şekillerin sınırları ve çizgiler kesiksiz veya noktalı çizgi türleri ile çizilebildiği gibi herhangi bir kalınlıkta ve renkte de çizilebilir.
- **Dolgu biçemi** : Şekillerin içi sabit bir renkle, desenle, renk basamakları ile veya herhangi bir resim ile doldurulabilir.
- **Metin** : Yazılar, yazı tipleri ve yazılar üzerinde biçemlemeyi sağlayan komutlar, iki boyutlu grafiklerle, grafiksel yazıların beraber kullanılabilmesini sağlar.

- **Görüntü İşleme** : İki boyutlu gelişmiş arayüzler, resimler üzerinde hazır görüntü işleme komutlarının uygulanabilmesini sağlar.
- **Dönüşüm** : Şekiller, resimler, yazılar küçültülüp, büyütülebilmekte, taşınabilmekte ve döndürülebilmektedir.
- **Kırpma** : Nesnenin bir pencere içinde kalan alanları dışındaki kısımlarını ortadan kaldırma işlemi kolaylıkla yapılabilmektedir.
- **“Antialiasing”** : Çizimlerdeki basamaklı kenarları düzleştirme tekniğidir.
- **Saydamlık** : Nesnelere geçirgenlik verilmesi üst üste gelen resim parçalarında daha gerçekçi görüntülerin oluşturulmasını sağlar.
- **Yazıcı** : Ekrandaki resmin ve şekillerin yazıcı çıktısının alınmasını kolaylaştıran komut ve özellikler önemlidir.

Bu uygulama geliştirme arayüzleri, kullandıkları programlama dilinin tüm özelliklerine grafik özellikleri de katmayı sağlamıştır. Ayrıca donanım ve yazılım sisteminin ve dilin grafik kullanıcı arayüzü özellikleri ile etkin pencereler ve pencere elemanları oluşturulabilmekte, fare ve klavye gibi cihazlarla başarılı bir etkileşim gerçekleştirilebilmektedir.

2. 2. Üç Boyutlu Grafik API’leri

Yıllar boyunca pek çok üç boyutlu grafik standardı geliştirilmiş ve bunlara dayanan uygulama geliştirme arayüzleri kullanıma sunulmuştur. İlk grafik standardı 1977 yılında hazırlanıp, 1979 yılında düzeltilen "Core" grafik standardıdır. Bu tarihten günümüze kadar, 4 kuşak grafik uygulama geliştirme arayüzü (API) ortaya çıkmıştır (Sun Microsystems, 2001).

Birinci kuşak 3B grafik API’lerinden olan “Core” dan sonra, İkinci kuşak API’lerden PHIGS ortaya çıkmıştır. C gibi güçlü bir programlama dilinden çağrılması, iş istasyonlarında kullanılması ve katı nesnelere sisteme eklenmiş olması önemli özelliklerindedir.

OpenGL, kaliteli ve yüksek düzeyli grafikler oluşturulmasını sağlayan, iki ve üç boyutlu grafikleri destekleyen, işletim sisteminden bağımsız üçüncü kuşak bir grafik uygulama geliştirme arayüzüdür. Etkileşimli üç boyutlu grafik uygulamalarında gerekli olan yaklaşık 250 farklı komutu içerir. Birçok platformda çalışması, geniş ve iyi tanımlanmış özellikleri, hızı, sürekli gelişmesi ve ölçeklenebilir olması sayesinde oldukça yaygınlaşmıştır. Güvenilir ve kullanımı kolay bir

donanım soyutlama katmanıdır. 1992 yılında ortaya çıkışından bu yana bilimsel programlardan görsel simülasyonlara, animasyonlardan oyunlara kadar çok geniş bir yelpazedeki uygulamanın yüksek performansla bilgisayar ortamına taşınmasını sağlamıştır.

Dördüncü kuşağa örnek olan Java 3D API, Java programlama diline üç boyutlu grafik ve görüntüleme yetenekleri kazandırmak üzere geliştirilmiştir. Java 3D API, etkileşimli üç boyutlu grafikleri içeren uygulamalar için bir dizi nesneye yönelik yüksek performanslı bir arayüz sağlar. Yazılım geliştiriciler, üç boyutlu nesnelere ve görsel ortamlar oluşturmayı, bunlar üzerinde işlemler yapmayı ve davranışlarını denetlemeyi kolaylaştırıp güçlendiren yüksek düzeyli ve kolay programlama modeliyle Java 3D'nin avantajlarından yararlanırlar. Bu model, görüntüleme tekniklerinden çok yazılım içeriğine odaklanmayı sağlar. Sonuçta yazılım geliştiriciler, yüksek kaliteli, ölçeklenebilir ve platform bağımsız üç boyutlu grafikleri Java tabanlı uygulamalar ve applet'ler ile kolaylıkla bütünleştirebilir (Day, 1998; Sowizral and Nadeau, 1999). Web tarayıcılarının birçoğu Java 3D Applet'lerini doğrudan çalıştıramamakla birlikte, Java 3D API yüklendikten sonra yapılan ayarlamalarla bu sorun da ortadan kalkmaktadır. Donanımların gelişmesi ile kısa sürede Java 3D'nin yaygın bir kullanım alanı bulacağı açıktır.

Bunlar dışında VRML ve Open Inventor gibi birçok üç boyutlu grafik uygulama geliştirme arayüzü vardır.

Yüksek düzeyli üç boyutlu uygulama geliştirme arayüzlerinin sağladığı olanaklar ve özellikler şu şekilde belirtilebilir :

- Üç boyutlu sanal geometrik nesnelere tanımlamayı, sahneye eklemeyi, çıkarmayı ve gruplandırma işlemlerini kolaylaştıran altyapıyı sunmaktadır. Matematiksel veri tipleri, veri yapıları ve bunlar üzerinde işlem yapacak çeşitli metotlar tanımlıdır. Programcılar kendi geometrik şekillerini tanımlayabilmek için indeks tabanlı olan veya olmayan geometrileri kullanabilirler.
- Üç boyutlu içerik eklemeyi kolaylaştırır:
- Üç boyutlu değişik grafik yazılımları ile oluşturulmuş dosya biçimindeki üç boyutlu nesnelere eklenmesi (Day, 1999)
- İki ve üç boyutlu yazılar eklenmesi
- Üç boyutlu nesnelere üzerinde değişik dönüşüm işlemleri ve değişik türlerde animasyonlar yaptırılmasını sağlarlar. Üç

boyutlu nesnelere üzerinde özellikle fare ve klavye gibi çevre aygıtları yardımı ile etkileşimli bir şekilde işlemler yapılabilmesini sağlayan olanaklar vardır.

- Üç boyutlu nesnelere niteliklerinin kolaylıkla belirlenmesini ve değiştirilmesini sağlarlar. Geniş bir nesne nitelikleri kütüphanesine sahiptirler.
- Nesnelere üzerine değişik türlerde desenler kaplanmasını kolaylaştırır.
- Değişik türlerde ışık kaynaklarının tanımlanmasını ve aydınlatma ve sis gibi etkilerin verilmesini kolaylaştırır.
- Kameraya yani bakış noktasına ilişkin işlemlerin yapılmasını kolaylaştırır. Üç boyutlu değişik görüntüleme olanakları (paralel, perspektif) gibi esnek kamera özelliklerini sunarlar.
- Üç boyutlu grafik ve animasyonlara ses özellikleri eklemeyi kolaylaştırır.
- Hazır geometri sıkıştırma özellikleri ile etkinlik sağlarlar.
- Güvenlik ve güvenilirliği sağlayıcı özellikleri mevcuttur.
- Altyapıları, grafiklerin oluşturulmasında ve diğer işlemlerde olabilecek en iyi hızı sunmaya çalışmaktadır.

2. 3. Üç Boyutlu Grafik Yazılımları Geliştirmede 2B ve 3B API'lerin Karşılaştırılması

3B grafik yazılımları geliştirmede 3B API'ler hazır olanaklar sağlamaktadır. Buna karşın, 3B yazılımların geliştirilmesinde, 2B API'lerin kullanımı ileri derecede matematik ve grafik bilgisi gerektirmektedir. İki boyutlu ilkeler yardımı ile üç boyutlu nesnelere oluşturulması, oluşturulan nesnelere için veri yapılarının tasarlanması, izdüşüm yöntemlerinin uygulanması, görünen yüzeylerin belirlenmesi, aydınlatma ve ışık kaynaklarının durumuna göre nesnelere yüzeylerindeki renklerin değerlerinin hesaplanması, yapılması gereken işlemlerden sadece birkaçıdır (Watt, 2000). Üç boyutlu grafik yazılımları geliştirmek bu nedenle yazılım geliştiriciler için zor olmuştur. Donanım ve yazılımlardaki gelişmeler sonucunda üç boyutlu uygulama geliştirme arayüzlerinin kişisel bilgisayar ortamına taşınması grafik yazılımları geliştirme sürecini hızlandıracak ve kolaylaştıracaktır. Bu nedenle bu çalışma kapsamında aynı işleve sahip biri 2B API biri de 3B API kullanılarak iki program geliştirilmiştir. Kod satır sayıları dikkate alınarak karşılaştırmalar yapılmıştır.

3. KUP3B YAZILIMI

Bu çalışma kapsamında geliştirilen ve internet üzerine yerleştirilen Kup3B yazılımı (web tarayıcıları ile "bornova.ege.edu.tr/~ugur" internet adresinden ulaşılarak çalıştırılabilmektedir) üç temel işleve sahiptir :

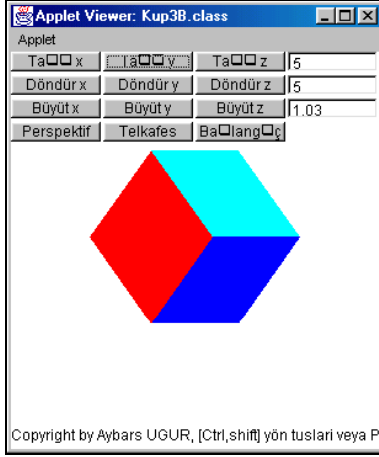
- Bir küp üzerinde kullanıcının temel dönüşüm işlemlerini (taşıma, döndürme ve ölçeklendirme) yapabildiğini sağlamak,
- Paralel/perspektif izdüşüm arasında geçiş yapılabilmesini sağlamak,
- Katı/telkafes görünüm arasında geçiş yapılabilmesini sağlamak.

Yazılım ölçümü, yazılımın kalitesini anlamak ve yazılım geliştiricilerin verimliliğini belirlemek gibi amaçlarla yapılır (Sommerville, 2000). Yazılım ölçütleri fiziksel ölçütler gibi iki çeşittir : Doğrudan ölçüler ve dolaylı ölçüler. Boyut uyarlı ölçütler, yazılım mühendisliği çıktısının ve kalitesinin doğrudan ölçülmesinde kullanılır. Fonksiyon uyarlı ölçütler, dolaylı ölçütlerdir. İnsan uyarlı ölçütler, yazılım geliştiren kişilerle ilgilidir. Yazılım mühendisliği işlemindeki doğrudan ölçüler, maliyet ve harcanan çabadır. Ürünün doğrudan ölçüleri arasında, üretilen kaynak kodun satır sayısı, işletim hızı, belirli bir süre içinde tespit edilen hata sayısı sayılabilir. Ürünün dolaylı ölçüleri, fonksiyonelliği, kaliteyi, karmaşıklığı, etkinliği, güvenilirliği ve diğer özellikleri içerir. Verimlilik ölçütleri, yazılım mühendisliği işleminin çıktısına odaklanır, kalite ölçütleri, müşteri gereksinimlerinin (yazılımın kullanılabilirliği) uygunluğunun göstergesidir. Teknik ölçütler ise yazılımın mantıksal karmaşıklığı gibi yazılımın karakterine odaklanır.

Bu çalışma kapsamında aynı işlevlere sahip fakat farklı arayüzlerle geliştirilen iki yazılım kod satır sayılarına göre karşılaştırılmışlardır. Yazılımların değişik bölümlerinin de satır sayıları dikkate alınmıştır. Doğrudan ölçütlerden olan kod satır sayılarının ölçülerek karşılaştırılması ile, grafik yazılımları geliştirme sürecindeki hızlanma ve yazılım geliştirmede harcanan çaba ve maliyetlerdeki azalma da ölçülmüş olmaktadır.

3. 1. Kup3B (Java2D Sürümü)

Java 2D kullanılarak geliştirilen Kup3B programında dönüşüm ve geçiş işlemleri paneldeki düğmeler veya klavye tuşları yardımı ile yaptırılmaktadır (Şekil 1).



Şekil 1. Kup3B yazılımı (Java 2D sürümü) applet penceresi açılış görünümü

Küpü üç boyutlu uzayda x,y,z eksenleri etrafında döndürmek için yön tuşları ile PageUp ve PageDown tuşları veya düğme panelindeki üç tane (üç eksen için) Döndür düğmesi kullanılır.

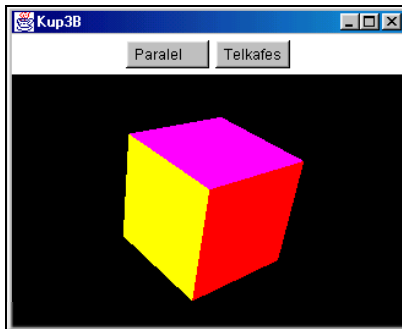
Küpü üç boyutlu uzayda x,y,z eksenlerinde taşımak için yön tuşları ile PageUp ve PageDown tuşları (Ctrl tuşu basılı iken) veya düğme panelindeki üç tane (üç eksen için) Taşı düğmesi kullanılır.

Küpü üç boyutlu uzayda x,y,z eksenlerinde ölçeklendirmek için yön tuşları ile PageUp ve PageDown tuşları (Shift tuşu basılı iken) veya düğme panelindeki üç tane (üç eksen için) Büyüt düğmesi kullanılır.

Paralel/Perspektif izdüşüm dönüşümü ve Katı/Telkafes görüntü dönüşümü, paneldeki düğmeler yardımı ile yapılır.

3. 2. Kup3B (Java3D Sürümü)

Java 3D kullanılarak geliştirilen Kup3B programında dönüşüm işlemleri fare yardımı ile, geçiş işlemleri paneldeki tuşlar yardımı ile yapılabilmektedir (Şekil 2).



Şekil 2. Kup3B yazılımı (Java 3D sürümü) applet penceresi açılış görünümü

Farenin sağ tuşuna basılarak x ve y eksenlerinde taşıma yapılabilmektedir. Alt tuşu ve farenin sol tuşu basılı iken fare yukarı/aşağı hareket ettirilerek z ekseninde taşıma gerçekleştirilmektedir. Fare, sol tuşu basılı iken değişik yönlerde hareket ettirilerek küp x, y ve z eksenleri etrafında döndürülebilmektedir.

4. SONUÇLAR

Benzeri işlemlere sahip olan ve bu çalışma kapsamında Java 2D ve Java 3D ile geliştirilen iki yazılımın bölümlerine göre kod satır sayıları Tablo 1'de verilmiştir. Tablodaki bölümlerde dikkate alınan etkinlikler aşağıdaki gibidir;

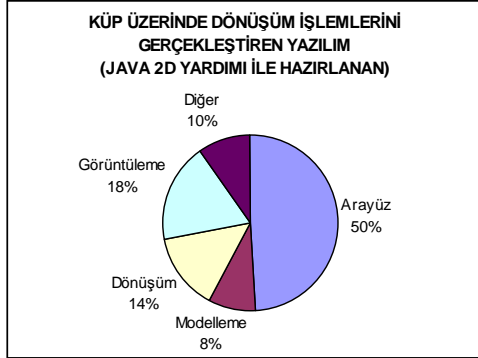
- Arayüz : Ekran pencerelerinin hazırlanması, düğmeler, metin kutuları ve diğer pencere bileşenlerinin oluşturularak eklenmesi, bu tür bileşenler ile, klavye ve fare ile etkileşim için oluşacak işlem kodları
- Modelleme : Veri yapılarının oluşturulması, sahnenin ve üç boyutlu nesnelerin oluşturulması ve veri yapılarına yerleştirilmesi
- Dönüşüm : Matris çarpım metotları, taşıma, ölçeklendirme ve döndürme gibi dönüşüm işlemlerini gerçekleştiren metotlar ile dönüşüm grupları ile ilgili işlemler
- Görüntüleme : Üç boyutlu nesnelerin veri yapılarından alınarak görüntülenmesi için yapılan tanımlama ve işlem kodları
- Diğer : Genel açıklamalar, standart programlama dili deyimleri, genel amaçlı veya sınıflandırılmayan deyim satırları.

Tablo 1. Kup3B Yazılımlarının Bölümlerine Göre Kaynak Kodlarındaki Satır Sayıları

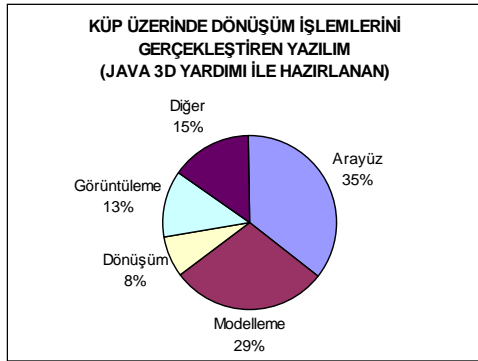
| Kup3B Yazılımının Bölümleri (Java 2D Sürümü) | |
|--|--------------|
| Bölüm | Satır Sayısı |
| Arayüz | 220 |
| Modelleme | 38 |
| Dönüşüm | 65 |
| Görüntüleme | 82 |
| Diğer | 44 |
| Toplam | 449 |

| Kup3B Yazılımının Bölümleri (Java 3D Sürümü) | |
|--|--------------|
| Bölüm | Satır Sayısı |
| Arayüz | 42 |
| Modelleme | 34 |
| Dönüşüm | 9 |
| Görüntüleme | 15 |
| Diğer | 18 |
| Toplam | 118 |

Şekil 3'te Java 2D, Şekil 4'te Java 3D sürümü için Kup3B kaynak kodundaki bölümlerin satır sayısı yüzdeleri gösterilmektedir.



Şekil 3. Kup3B yazılımı kaynak kodundaki bölümlerin satır sayısı yüzdeleri

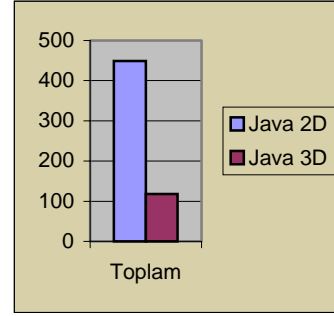
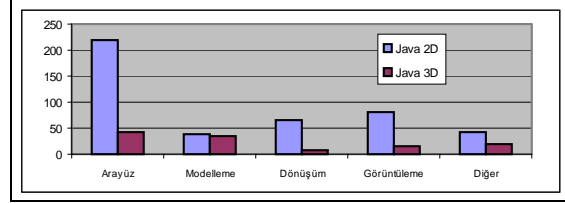


Şekil 4. Kup3B yazılımı kaynak kodundaki bölümlerin satır sayısı yüzdeleri

Aynı işlevlere sahip grafiksel yazılımların geliştirilmesinde, yazılan kod satır sayıları dikkate alındığında bu çalışma için $449/118 = 3.8$ oranı ortaya çıkmaktadır. Başka bir deyişle, grafik yazılımları geliştirmek için harcanan çaba yüksek düzeyli üç boyutlu bir uygulama geliştirme arayüzünden yararlanıldığında % 74 azalarak, % 26'sına inmiştir (Şekil 5). İki boyutlu uygulama geliştirme arayüzleri yerine üç boyutluların tercih edilmesinin getireceği bu yarar maliyetlere ve yazılım geliştirme süresine de doğrudan yansiyacaktır.

Fare etkileşimi, temel nesnelere ve üzerlerindeki işlemler ile görüntüleme, Java 3D API ile hazır geldiğinden, arayüz ve görüntüleme için yazılan kaynak kodu satır sayıları oldukça büyük oranlarda azalmıştır (Şekil 5). Çalışmadan elde edilen sonuçlara göre, üç boyutlu API kullanımı *arayüz oluşturmak* için harcanan çabayı % 81, *görüntüleme* için gereken çabayı % 82 ve *dönüşüm işlemleri* için harcanan çabayı yaklaşık % 86 azaltmıştır.

Yazılımda tek nesne olduğu için, *modelleme* kısmında önemli bir kod azalması görülmektedir.



Şekil 5. Java 3D kullanımının, geliştirilen Kup3B yazılımının (bölümlerine göre) kaynak kod satır sayısında sağladığı azalmayı gösteren grafik.

Tüm yazılımlarda verimin bu derece artması beklenemez. Bunu etkileyen faktörler şu şekilde sayılabilir:

- Geliştirilecek yazılımın doğası, verimliliği ve yazılım geliştirme süresini etkileyecektir. Bazı grafik yazılımları doğrudan grafik, modelleme, boyama, çizim gibi işlemleri yaptırarak yazılımlar olabildiği gibi, bazı yazılımlar da diğer alanlarda olup içinde grafiklerin kullanımını içeren yazılımlardır. Yazılımın içeriğine ve amacına göre bu oran değişecektir.
- Yazılımın karmaşıklık düzeyi ve grafiklerdeki gerçekçiliğin düzeyi de önemli bir faktördür. Geliştirilecek yazılımın grafiksel gerçekçilik beklentisi de bu oranı değiştirecektir.
- Yüksek düzeyli 3B API'lerin ileri işlevlerinin kullanımının öğrenilmesi zordur. Yazılım geliştiricilerin bilgi ve deneyim farklılıkları harcanan çaba, yazılım geliştirme süresi ve maliyetlerdeki bu oranı etkileyecektir.

Üç boyutlu nesnelere için fare etkileşim özelliklerinin doğrudan gelmesi bu amaçla yazılması gereken kod satır sayısını büyük ölçüde azalttığı gibi, tüm yazılımlarda belli bir standardı da sağlamakta ve yazılım kalitesini de arttırmaktadır. Temel nesnelere (küp, küre, koni ve silindir gibi) desteklenmesi, birçok basit sahnenin

oluşturulabilmesini sağlamakta ve birçok 3B basit yazılım için yeterli olmaktadır. Üç boyutlu arayüzlerin ayrıca dışarıdan hazır üç boyutlu model yüklenebilmesini sağlama, üç boyutlu geometrik içerik oluşturmayı sağlama ve üç boyutlu metin oluşturma gibi önceki bölümlerde bahsedilen birçok özellikleri de vardır.

Sonuçta yüksek düzeyli 3B API'ler üç boyutlu grafik yazılım geliştirme sürecini hızlandırmakta ve kolaylaştırmaktadır. Bu da, geliştirilen yazılımlardaki kaliteyi arttırmakta, maliyetleri düşürmekte ve yazılımların sürümlerinin gecikmesinden doğabilecek sorunları ortadan kaldırarak yeni fırsatlar oluşmasını sağlamaktadır.

5. KAYNAKLAR

Day, B. 1998. 3D Graphics Programming in Java : Part 1, Java 3D, Javaworld. Aralık 1998.

Day, B. 1999. 3D Graphics Programming in Java : Part 2, Advanced Java 3D, Javaworld Ocak 1999.

Knudsen, J. 1999. Java 2D Graphics, O'Reilly, USA, 339 p.

Sommerville, I. 2000. Software Engineering, 6th Edition, Addison-Wesley, USA, 693 p.

Sowizral, H. A. and Nadeau, D. R. 1999. Introduction to Programming With Java 3D, San Diego Supercomputer Center, University of California at San Diego, SIGGRAPH99 Course Notes, 603 p.

Sun Microsystems, Inc., Java 3D 1.2 API Specification, www.javasoft.com/products/java-media/3D/index.html

Watt, A. 2000. 3D Computer Graphics, Third Edition, Addison-Wesley, 570 p.