

**İSTANBUL TEKNİK ÜNİVERSİTESİ  
ELEKTRİK - ELEKTRONİK FAKÜLTESİ**

**ÜÇ BOYUTLU MEDİKAL GÖRSELLEŞTİRME**

**Bitirme Ödevi**

**Dilek Üstünelan  
040040228**

**Neşe Erden  
040030050**

**Bölüm : Bilgisayar Mühendisliği  
Anabilim Dalı: Bilgisayar Bilimleri**

**Danışman : Yrd.Doç.Dr. Mustafa Ersel Kamaşak**

**Mayıs 2009**

## **Özgünlük Bildirisi**

Bu çalışmada, başka kaynaklardan yapılan tüm alıntılarını, ilgili kaynaklar referans gösterilerek açıkça belirtildiğini,  
Alıntılar dışındaki bölümlerin, özellikle projenin ana konusunu oluşturan teorik çalışmalarını ve yazılım/donanımını bizim tarafımızdan yapıldığını bildiririz.

İstanbul, 18.15.2009

Dilek Üstünelan

Neşe Erden

# ÜÇ BOYUTLU MEDİKAL GÖRSELLEŞTİRME

## ( ÖZET )

Bu çalışmada, 3 boyutlu veri görselleştirme teknik ve algoritmaların öğrenilmesi ve medikal alanda görselleştirme uygulama geliştirilmesi amaçlanmıştır.

Verinin kullanıcıya görsel olarak sunulması, anlaşılabilirliğinin artırılması, verinin üzerinde işlem yapma kolaylığının sağlanması amacıyla görselleştirme uygulamaları geliştirilir. Medikal alanda, 3 boyutlu görselleştirme, tıp eğitiminde, hastalıkların cerrahi müdahaleye gerek duyulmadan teşhisin yapılabilmesinde etkili bir role sahiptir. Özellikle cerrahi alandaki simülasyonları tıbbi gelişmelerin hız kazanmasında önemli bir etkidir.

Proje dahilinde, bilgisayar grafiklerinin temel teknikleri, görselleştirme iş hattı, temel veri tipleri ve bunların temel grafik bileşenlerine dönüştürülmesi, temel ve ileri seviyede görüntüleme algoritmaları incelenmiştir. Yüzey ve hacim görüntüleme teknikleri öğrenilmiştir. Düzgünleştirme, gürültüden arındırma, segmentasyon, gölgelendirme gibi görüntü işleme teknikleri incelenmiştir. Görüntüleme algoritmalarının karmaşıklık ve etkinlik özellikleri araştırılmış; iyileştirme algoritmaları incelenmiştir. Medikal görselleştirmenin yanı sıra, finansal görselleştirme, örtülü modelleme, algoritma görselleştirme teknikleri de araştırılmıştır.

Visualization Toolkit kütüphanesi öğrenilmiş bu kütüphane kullanılarak 3 boyutlu görselleştirme uygulamaları geliştirilmiştir. DICOM tipindeki medikal görüntülerden oluşan veri kümeleri kullanılarak yüzey ve hacim görüntüleme teknikleriyle görselleştirme projesi gerçekleştirilmiştir. Gerçeklenen proje kullanıcıyla etkileşimli olacak şekilde tasarlanmıştır.

Projede görselleştirme hakkında temel bilgiler edinilmiş, temel görüntüleme algoritmaları ile amaçlanan uygulama gerçekleştirilmiştir. İleri düzey görüntü işleme teknikleri ve farklı tiplerde veri kümeleriyle proje geliştirilmeye açıktır.

# 3D MEDICAL VISUALIZATION

## ( SUMMARY )

The main aim of this Project is to learn 3 dimensional data visualization technics and algorithms and to implement an application of medical visualization.

Visualization is the transformation of the data or information into the images. Data visualization is a term which is used for transforming all types of datasets into visual forms. The process of data visualization covers the application of the statistical method and other data analysis technics to the data sets.

Visualization applications are developed in order to represent the data to the user in a visual form, to improve the understandability of the data and to facilitate processing of data. In medical field, via X-rays CT and MRI images are provided, MRI images are combine big magnetic fields with radio waves. When all the images are combined with complex computational formulas the dimensional data is occurred. 3D visualization has an effective role in education of medical science. The surgery simulations have great importance in medical development. Especially, without any surgery operations, great improvements are seen in medical diagnosis; illness is noticed in the beginnings and the charges are decreasing. It is a fact that the lung cancer diagnosis is very improved by 3D visualization. It is believed that the more progress of visualization the less death from cancer.

During the study, the computer graphics primitive technics, the visualization pipeline, basic dataset types, the transform of original data to computer graphics primitives, basic and advanced rendering algorithms are studied. The technics of surface rendering and volume rendering are learned. The image processing technics, such as smoothing, image restoration, enhancement, segmentation and shading examined. The complexity and efficiency properties of the rendering algorithms are analysed. The algorithms of financial visualization, implicit modelling and algorithm visualization are researched besides medical visualization.

The visualization process and computer graphics are dependent to the computer hardware but Visualization Toolkit Library supplies an interface to use graphics capabilities independent from the hardware. Besides, the compiler cMake helps VTK to be used hardware-free.

The main steps of visualization pipeline are transformation and representation. transformation is the process of converting data from its original form into graphics primitives and eventually into computer images. Visualization transforms a computational form into a graphical form. Representation includes both the internal data structures and the graphics primitives.

The steps of 3D medical rendering are:

- Read the input
- Create an isosurface for each classified tissue
- Transform the models in the data into world space
- Render the models

Data reading process is completed via taking the CT(Computed Tomography) and MRI(Magnetic Resonance Imaging) data in DICOM form as dataset and making the data usable using visualization toolkit DICOM data reading functions.

Transformation process is implemented by the transforms between the most important computer graphics coordinate systems, model, world, view and display coordinate systems, using 4x4 transformation matrices. Transformation process allows the representation of data in computer graphics primitives.

Rendering methods can be addressed in two main methods: Image order and Object order. Ray tracing method is an image order process. It works by tracing all possible rays passing through the volume.contrary, the object order methods render all single objects.

One of the main processes of rendering is creating isosurface. Three methods are used to create the isosurface: marching cubes, volume rendering and dividing cubes. Marching cubes is used in the application, because marching cubes is a relatively fast algorithm. This algorithm renders polygons to create the surfaces. It enables both surface and volume rendering of the extracted tissues with the opacity values. Marching cubes class creates an extractor for each tissue. The values of this extractor is mapped to an actor object; so the rendering process is implemented. The code for this rendering process is given in the table below. [14]

```
vtkMarchingCubes *skinExtractor = vtkMarchingCubes::New();
skinExtractor->SetInput(reader->GetOutput());
skinExtractor->SetValue(0,600);

vtkStripper *skinStripper = vtkStripper::New();
skinStripper->SetInput(skinExtractor->GetOutput());

vtkPolyDataMapper *skinMapper = vtkPolyDataMapper::New();
skinMapper->SetInput(skinExtractor->GetOutput());
skinMapper->ScalarVisibilityOff();
vtkActor *skin = vtkActor::New();
skin->SetMapper(skinMapper);
skin->GetProperty()->SetDiffuseColor(0.9200, 0.8100, 0.6555 );
skin->GetProperty()->SetOpacity(0.2);

aRenderer->AddActor(skin);
```

In order to render the 3D data, surface rendering or volume rendering technique should be used.

1. Surface rendering is rendering the isosurfaces of an object. Isosurfaces are determined by the density threshold value for the tissues.using this technique only the surface of the volume is rendered, the interior area is ignored. The surface is opaque.
2. Volume rendering is rendering both the surface and interior area of an object. Determining the opacity value, the tissues can be separately rendered and displayed. The opacity value differs in the range (0,1). 0 represents a total transparent object and 1 represents a total opaque object. In a medical data, these tissues are classified generally

as skin, muscle and bone. By this method, the exterior tissues can be displayed transparent or semi transparent. So the tissues can all be displayed as layers.

Image processing used for removing noise, smoothing, reducing sampling artifacts, image enhancement, segmentation, erosion and dilation. There are image restoration techniques which can filter the noise without destroying the original data. Smoothing process flattens the surfaces preserving edges; so the rendered image is displayed noise-free. Segmentation is the process of classifying the pixels in an image or volume.

With coloring and illumination, the determined tissues of the data, is displayed in determined colors. The colors can be determined by giving the color values in RGB and HSV color system to the VTK coloring functions.

The Visualization Toolkit library is studied and using this library, 3D visualization applications are performed. C++ programming language is used to develop this application. Via using the datasets of DICOM images a visualization application of surface rendering and volume rendering is implemented. The Project is modelled user-interactive.

The Project is studied with the datasets consisting of CT and MRI images in the DICOM(Digital Imaging in Communications and Medicine) form. These datasets are formed of the slices of human body. The size of the datasets differs according to the number of slices and resolution. For example, one of the datasets used in the project representing a thorax consists of 341 slices and is size of 11.3 MB. Another dataset used in project representing feet consists of 250 slices and is size of 125 MB. The difference of size and number of slices between the datasets is one of the main factors which determines the execution time.

The dataset contains dicom files. The dicom files which represents the slices, named with a prefix, same for all and a suffix which is the order of slice. For example, the files in a dataset are named as follows: IM\_00001, IM\_00001, ... , IM\_00250.

In the application part of this project, surface rendering and volume rendering process is implemented. And two datasets: thorax and vix are used to test the application. In the paper, the results of four surface rendering experiments and two volume rendering experiments on these datasets are explained. The execution times are measured. The far difference between the execution times of two datasets are based on the size of the data.

To explore volume data, VTK has different useful techniques apart from isocountering which point the extracted geometry and have scalar value from the original data. By these techniques three orthogonal planes corresponding to axial, sagittal and coronal crosssections are extracted. In the Project, these planes are extracted and displayed; both combined and single. In the application, the axial plane is perpendicular to the backbone, sagittal passes from left to right and coronal passes from front to back. Each plane is rendered with a different color lookup table. These lookup tables contain color value; with the elements of HSV color system: hue, saturation and value. The code for creating a lookup table and the actor object for the axial plane can be seen in the following table.

```

vtkLookupTable *hueLut = vtkLookupTable::New();
hueLut->SetTableRange(0,2000);
hueLut->SetSaturationRange(0.6,0.6);
hueLut->SetHueRange(0,1);
hueLut->SetValueRange(0.8,0.8);
        vtkImageDataGeometryFilter      *axialSection      =
vtkImageDataGeometryFilter::New();
axialSection->SetExtent(0,128,65,65,0,350);
axialSection->SetInput(reader->GetOutput());

vtkPolyDataMapper *axialMapper = vtkPolyDataMapper::New();
axialMapper->SetInput(axialSection->GetOutput());
axialMapper->ScalarVisibilityOn();
axialMapper->SetScalarRange(1450,2450);
axialMapper->SetLookupTable(hueLut);
vtkActor *axial = vtkActor::New();
axial->SetMapper(axialMapper);
aRenderer->AddActor(axial);

```

To make the whole application user-interactive, a user interface is medelled on Microsoft Visual Studio Forms Application. Using this interface, user can choose either surface rendering or volume rendering, determine the dataset for rendering, choose one or more tissues from the bone, muscle and skin tissues of medical data, determine the density treshold, color and opacity values of the chosen tissues.

During this study, basic knowledge of visualization is gained and the planned application is performed with basic rendering algorithms. Project can be developed using advanced image processing technics such as smoothing, filtering, segmentation, illumination, and datasets of different data types. Execution time can be reduced using more efficient algorithms such as space leaping.

With the appropriate enhancements on this project and likewise medical visualization projects, medical science would have an accelerating improvement.

# İÇİNDEKİLER

1. GİRİŞ.....	1
2. PROJENİN TANIMI VE PLANI.....	3
2.1. Projenin Tanımı ve Kapsamı.....	3
2.2. Proje Planı.....	4
2.3. Görev Paylaşımı.....	5
3. KURAMSAL BİLGİLER.....	6
3.1. Görselleştirme.....	6
3.2. Bilgisayar Grafiklerinin Temelleri.....	6
3.3. Veri Görselleştirme İşlemi İş Hattı.....	9
3.3.1. Fonksiyonel Model.....	10
3.3.2. Nesne Modeli.....	11
3.3.3. Görselleştirme İş Hattı.....	11
3.3.4. Programlama Modelleri.....	11
3.3.4. Temel Veri Gösterimleri.....	12
3.4. İleri seviye Görüntüleme Teknikleri.....	12
3.5. Görüntü İşleme.....	13
3.6. 3-Boyutlu Görselleştirme Uygulama Alanları.....	14
3.6.1. Medikal Görüntüleme.....	14
3.6.1.1. Veri Okuma.....	15
3.6.1.2. Isosurface Oluşturma.....	15
3.6.2. Ticari Görselleştirme.....	16
3.6.3. Implicit modelleme.....	16
3.6.4. Algoritma Görselleştirme.....	17
4. ANALİZ VE MODELLEME.....	18
5. TASARIM, GERÇEKLEME VE TEST.....	20
5.1. Çalışma Ortamı.....	20
5.2. Veri Kümesi.....	20
5.3. Veri Kümesinin Okunması.....	21
5.4. Verinin Görüntülenmesi.....	21
5.5. Yüzey Görüntüleme.....	23
5.6. Hacim Görüntüleme.....	24
5.7. Testler.....	26
6. DENEYSEL SONUÇLAR.....	27
6.1. Geometrik Cisimlerin Görüntülenmesi.....	27
6.2. Yüzey Görüntüleme.....	27
6.3. Hacim Görüntüleme.....	31
6.4. Aksiyel, Sagital ve Koronel Eksenlerin Görüntülenmesi.....	33
7. SONUÇ VE ÖNERİLER.....	35
8. KAYNAKLAR.....	36



# GİRİŞ

Bilgisayarlı grafik yöntemlerinin gelişmesi ile birlikte birçok alandaki verilerin görsel olarak ifade edilmesi ve çeşitli alanlardaki görselleştirme uygulamaları yaygınlık kazandı. Veri görselleştirme, tıbbi, finansal, haritalama, endüstri ve eğlence sektöründe etkili bir şekilde kullanılmaktadır. Çok büyük boyut ve karmaşıklıkta verilerin daha anlaşılır ve kullanılabilir şekilde ifade edilebilmesi için görselleştirme tercih edilir bir yöntem olmuştur.

Medikal alanda, X-Ray cihazlarıyla, CT ya da MRI formatında alınan verilerin işlenmesi, görüntülenmesi ve teşhis-tedavi alanında kullanılması için görselleştirme çok önemli bir yer kazanmıştır. Medikal görüntüleme, aynı zamanda, genetik araştırmalar, tıp personel eğitimi, canlı türlerinin incelenmesi alanlarında da etkin bir şekilde kullanılmaktadır.

Dünyada önemi gittikçe artan medikal görselleştirme çalışmaları, Türkiye’de nispeten daha yavaş ve yetersiz ilerlemektedir. Bu nedenle, bu alandaki araştırmalarda yer almak üzere bitirme projesi olarak 3 boyutlu medikal görselleştirme araştırma ve uygulamaları seçilmiştir.

Bu proje kapsamında, bilgisayarlı grafik teknikleri, veri tipleri, veri kümelerinin işlenmesi, temel veri görüntüleme, görüntü işleme teknikleri, temel görselleştirme algoritmaları araştırılmış ve öğrenilmiştir. Proje, medikal veri görselleştirme konusuna odaklanmış olmakla beraber, finansal veri görselleştirme alanında da araştırmalar yapılmıştır.

Medikal görselleştirme projesinde, canlı vücutlarından yatay kesitler alınarak oluşturulan veri kümeleri kullanılmış, bu kümeler üzerinde yüzeysel ve hacimsel olarak 3 boyutlu görselleştirme uygulamaları gerçekleştirilmiştir.

Bu rapor, aşağıdaki düzenlenmiştir:

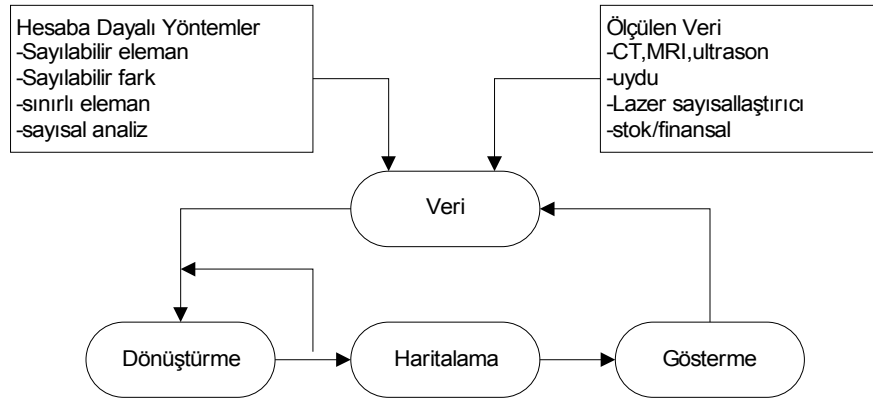
- 2.Bölümde projenin amacı ve kapsamı detaylıca açıklandıktan sonra projenin hangi parçalara ayrıldığı ve iş planı verilmiştir. Grup elemanlarının projedeki iş yükleri açıkça belirtilmiştir.
- 3.Bölümde 3 boyutlu veri görselleştirme işleminin daha iyi anlaşılması için gerekli kuramsal bilgiler verilmiştir. Veri görüntüleme ve görüntü işleme tekniklerine ilişkin detaylar anlatılmıştır.
- 4.bölümde medikal veri görselleştirme işleminin basamakları analiz edilmiş; bu basamakların gerçekleşmesine ilişkin model tanımlanmıştır.
- 5.Bölümde bir önceki bölümde analizi ve modellenmesi yapılan bileşenlerin nasıl gerçekleştiği, kullanılan yazılımlar ve yöntemler açıklanmış, gerekli görülen yerlerde detaylar verilmiştir. Daha sonra önerilen çözümün testi için kullanılan senaryolar ve bu testlerin uygulandığı veri kümeleri hakkında detaylı bilgi verilmiştir.

- 6.Bölümde deneyler sonucu elde edilen veriler, program çıktıları olarak ekran görüntüleri verilmiştir.
- 7.Bölümde ise görselleştirme konusunda yapılan teorik çalışma ve gerçekleştirilen görselleştirme uygulaması değerlendirilmiştir; uygulamanın eksik görülen yönlerini geliştirmeye ilişkin öneriler sunulmuştur.

# PROJENİN TANIMI VE PLANI

## Projenin Tanımı ve Kapsamı

Görselleştirme(visualization), genel anlamıyla, verilerin veya bilgilerin resimlere dönüştürülmesidir. Başka bir deyişle, ham veriler arasındaki ilişkilerin ve bulguların daha hızlı anlaşılabilir bir formda görsel olarak sunulması işlemidir.



Şekil 2.1 : Veri görselleştirme işlemi

Gerçek verilere ulaşmak ve bu verileri işlemek, kullanmak hem pahalı olduğu, hem de kolay olmadığı için bu gerçek veri kümelerinin görselleştirilmiş örnekleri ya da simülasyonları ile çalışmak birçok avantaj sağlar:

Medikal görüntüleme tıp personelinin ya da öğrencilerinin gerçek verilerle araştırma yapmaları her zaman mümkün olmayabilir. Medikal görselleştirme ile gerçekleştirilen bir simülasyon üzerinde çalışmak, pratiği güçlendirebilir.

X-Ray cihazları ile elde edilen verilerin görselleştirilmesi, ameliyat gerektirmeden, 3 boyutlu veri sağlanmasını, teşhis ve tedavinin kolaylaşmasını mümkün kılar.

Psikolojide, sanal gerçeklik kapsamında, fobilerin tedavisi, görselleştirme uygulamaları ile sağlanır.

Türlerin incelenmesi, araştırılması çalışmalarında verilerin daha güvenilir olmasını sağlar, verilerin işlenmesini hızlandırır.

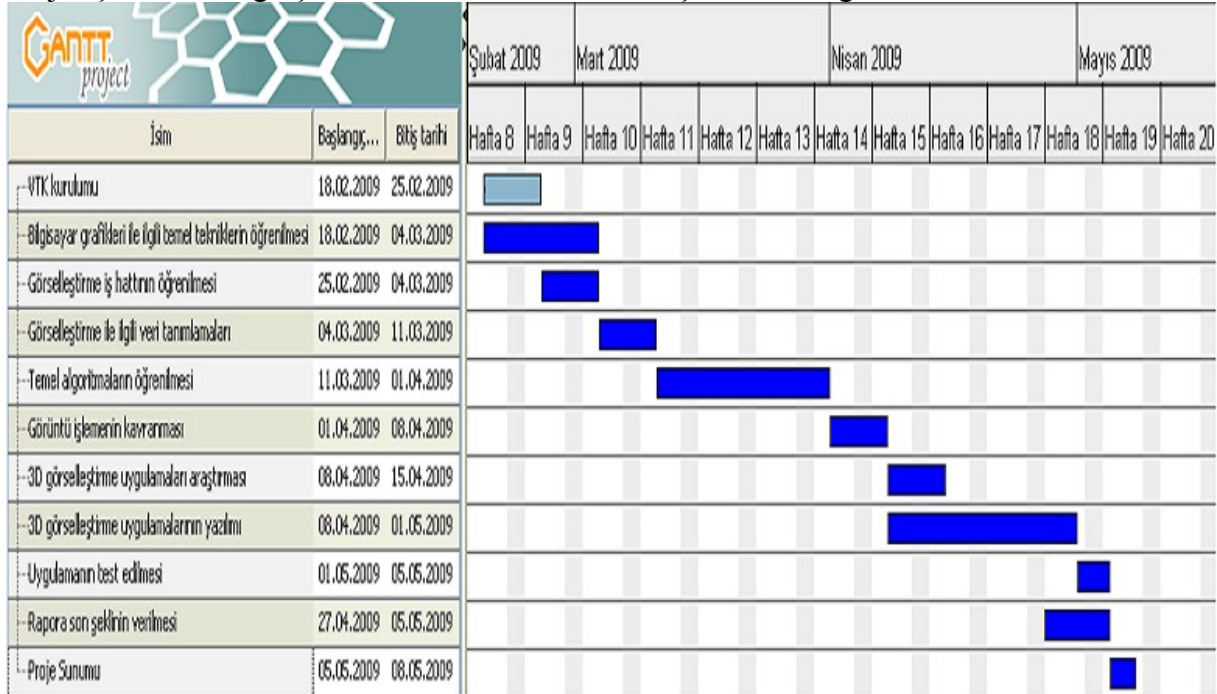
Medikal veri görselleştirme, iki farklı teknikle ele alınmıştır. Yüzeysel görüntüleme, bir nesnenin doku yoğunluğu eş yüzeylerini, noktalar, doğrular, vektörler, üçgenler ve çokgenler yardımıyla görüntüleyen ve nesnenin iç kısmıyla ilgilenmeyen 3 boyutlu görüntüleme tekniğidir. Hacimsel görüntüleme ise, nesnenin iç bölgesini, farklı doku yoğunluklarına sahip kısımları ve katmanlarıyla görüntüleyen 3 boyutlu görüntüleme tekniğidir. Bu teknik, nesnenin hem yüzeyinin, hem de iç bölgesinin, farklı yoğunluk ve opaklık değerleriyle görüntülenebilmesini sağlar. Projede, bu iki tür görselleştirme algoritması da uygulanmıştır.

## 2.2. Proje Planı

Projenin gerçekleşme aşamaları aşağıdaki gibidir:

- Visualization Toolkit kurulumu
- Bilgisayar grafikleri ile ilgili temel tekniklerin öğrenilmesi
- Görselleştirme iş hattının öğrenilmesi
- Görselleştirme ile ilgili veri tiplerinin araştırılması, veri kümelerinin elde edilmesi
- Temel görselleştirme algoritmalarının araştırılması, öğrenilmesi
- Görüntü işlemenin kavranması
- 3 boyutlu görselleştirme algoritmalarının araştırılması
  - Finansal veri görselleştirme uygulamaları
  - Medikal veri görselleştirme algoritmaları
- 3 boyutlu görselleştirme uygulamalarının yazılımı
  - Yüzeysel görüntüleme
  - Hacimsel görüntüleme
- Uygulamanın farklı veri kümeleri ile test edilmesi
- Proje raporuna son şeklinin verilmesi

Proje aşamalarının gerçekleşme süreleri ve tarihleri Şekil 2.2’de görülebilir.



Şekil 2.2 : Proje Planı

## 2.3. Görev Paylaşımı

Görev paylaşımı ile ilgili detaylar Tablo 2.3’de görülebilir.

**Tablo 2.3:** Görev Paylaşımı

Görev Tanımı	Paylaşım		Önem Derecesi (1-5)
	Neşe Erden	Dilek Üstünelan	
	%100	%100	3
Visualization Toolkit kurulumu	%100	%100	3
Bilgisayar grafikleri ile ilgili temel tekniklerin öğrenilmesi	%75	%25	3
Görselleştirme iş hattının öğrenilmesi	%100		3
Görselleştirme ile ilgili veri tiplerinin araştırılması, veri kümelerinin elde edilmesi		%100	4
Temel görselleştirme algoritmalarının araştırılması, öğrenilmesi	%50	%50	3
Görüntü işleminin kavranması	%25	%75	4
3 boyutlu görselleştirme algoritmalarının araştırılması	%50	%50	5
3 boyutlu görselleştirme uygulamalarının yazılımı (yüzeysel görüntüleme)	%75	%25	5
3 boyutlu görselleştirme uygulamalarının yazılımı (hacimsel görüntüleme)	%25	%75	5
Uygulamanın farklı veri kümeleri ile test edilmesi	%50	%50	2
Proje raporuna son şeklinin verilmesi	%50	%50	3

# KURAMSAL BİLGİLER

Bu bölümde 3 boyutlu veri görselleştirme işleminin daha iyi anlaşılması için gerekli kuramsal bilgiler verilecektir.

## Görselleştirme

Görselleştirme, verilerin ya da bilgilerin resimlere dönüştürülmesidir. Görselleştirme, insanın birincil duyu organlarını, görme gücünü olduğu gibi, insan aklının işlem gücünü de kullanır.

Bilimsel görselleştirme(scientific visualization) kullanıcı arayüzlerini, veri gösterim ve işleme algoritmalarını, görsel gösterimleri ve ses, dokunma gibi diğer duyuşal gösterimleri kapsayan bilgisayar bilimi alanına verilen resmi addır.

Veri görselleştirme(data visualization) terimi ise daha genel kapsamda; tüm veri kümelerinin görsel hale getirilmesine ilişkin kullanılan bir terimdir. Bu veriler, finansal veriler, ticari veriler ya da iş verileri olabilir. Veri görselleştirme işlemi, istatistiksel metodların ve diğer veri analizi tekniklerinin veri kümelerine uygulanmasını da kapsar.

Veri görselleştirme teknikleri, medikal alanda, teşhise ilgili önemli bir araç olarak kullanılmaktadır. Bu teknikler arasında, CT(Computed Tomography) ve MRI(Magnetic Resonance Imaging) sayılabilir. Bu yöntemler, hastanın dahili anatomisiyle ilgili bilgi elde etmek için, bir örnekleme ya da veri toplama işlemi kullanır. Bu bilgi, hastadan belli aralıklarla alınmış düzlemsel kesitlerin resimleri formatındadır. Bu kesitler bir araya getirilerek, hacimsel bir görüntü elde edilir; hacimsel görüntüler de işlenerek anatomik yapılar ortaya çıkarılabilir. Modern teknikler kullanılarak, hiçbir cerrahi müdahaleye gerek duyulmadan, bir hastanın beyni, iskelet sistemi ya da dolaşım sistemi bir bütün olarak görüntülenebilir. Bu görüntü işlenerek, dokusal yoğunluklara göre bölmelere ayrılabilir; canlı vücudunun görüntüsünde bu bölmeler genel olarak, kemik, kas ve cilt dokularından oluşur.

Veri görselleştirme algoritmaları, görüntüleme ve bilgisayarlı grafik işlemlerini olduğu gibi, veri işleme ve filtreleme, kullanıcı arayüzü oluşturma, hesaplama teknikleri ve yazılım modelleme ve tasarım işlemlerini de kapsar.

Visualization Toolkit, nesneye dayalı modelleme tekniği kullanılarak modellenmiş bir kütüphanedir. Modellemede, nesne modelleme tekniği(object modelling technique) kullanılmıştır. Bu teknik, nesneye dayalı bir dizayn oluşturmak için 3 model kullanmaktadır: nesne modeli, dinamik model ve fonksiyonel model.

## Bilgisayar Grafiklerinin Temelleri

Görüntülemeye ilişkin temel metodlar, görüntüye dayalı(image-order) ve nesneye dayalı(object-order) görüntüleme metodlarıdır.

Işın izleme(ray tracing) yöntemi, görüntüye dayalı bir işlemdir. Her bir ışının aldığı yolu ve zaman içindeki durumunu izleyerek çalışır. Nesneye dayalı metodlar ise, her bir nesneyi ayrı ayrı görüntüleyerek çalışır.

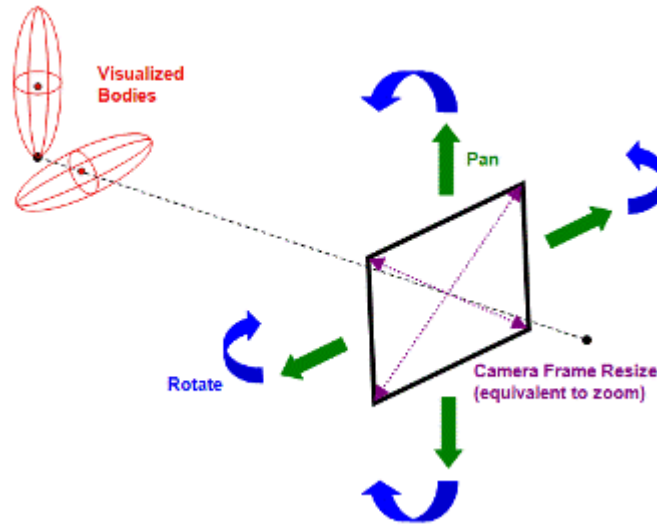
Yüzeysel – hacimsel görüntüleme: Yüzeysel görüntüleme(surface rendering) tekniği ile, nesnelerin yüzeyleri görüntülenir. Hacimsel görüntüleme(volume rendering) tekniği ile ise, nesnelerin hem yüzeyi hem de iç bölgesi görüntülenir.

Görüntülemenin önemli bir bileşeni renklendirmedir. Nesnelerin renkleri, RGB ve HSV renk sistemleri kullanılarak ifade edilir. RGB sistemi, renkleri, içlerindeki kırmızı, yeşil ve mavinin yoğunluklarına göre ifade eder. HSV sistemi ise, renkleri renk(hue), canlılık(saturation) ve ton(value) değerleri ile ifade eder. Renk, baskın dalga boyunu, canlılık, dalga boyunun ne oranda rengin içinde bulunduğunu, ton ise rengin parlaklığını ifade eder. [18]

**Tablo 3.2:** Yaygın renklerin RGB ve HSV değerleri

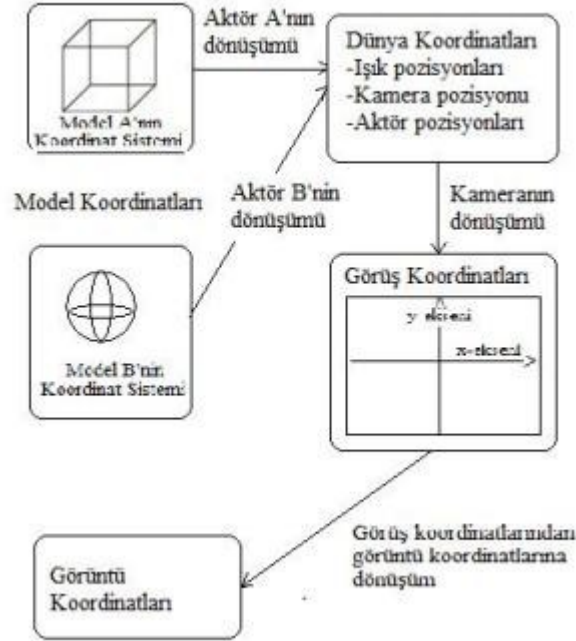
Color	RGB	HSV
Siyah	0,0,0	*,*,0
Beyaz	1,1,1	*,0,1
Kırmızı	1,0,0	0,1,1
Yeşil	0,1,0	1/3,1,1
Mavi	0,0,1	2/3,1,1
Sarı	1,1,0	1/6,1,1
Cyan	0,1,1	1/2,1,1
Magenta	1,0,1	5/6,1,1

Kameranın pozisyonu, yönü, odak noktası, kamera izdüşüm metodu, kamera düzleminin sınırları, 3 boyutlu bir nesnenin 2 boyutlu olarak gösterilmesinde önemli faktörlerdir.



**Şekil 3.2.1:** Kameranın görüş doğrultuları

Bilgisayarlı grafiklerde kullanılan dört yaygın koordinat sistemi vardır: model, dünya, görüş, görüntü.



Şekil 3.2.2: Model, dünya, görüş ve görüntü koordinat sistemleri

Model koordinat sistemi, her modele ait, lokal bir Kartezyen sistemdir. Dünya koordinat sistemi, aktörlerin, kameraların ve ışıkların pozisyonlarının ve yönelimlerinin belirlendiği sistemdir. Görüş koordinat sistemi, kameranın gördüğü görüntüyü belirtir. (-1,1) aralığında değerler alan x ve y değer çiftlerinden ve z (derinlik) koordinatından oluşur. X ve y değerleri, görüntü düzlemindeki konumu, z değeri ise kameraya uzaklığı belirtir. Kameranın özellikleri, dünya koordinatlarından görüş koordinatlarına dönüşümü gerçekleştirmekte kullanılan 4x4 bir dönüşüm matrisiyle ifade edilir. Görüntü koordinat sisteminin x,y koordinatları, görüntü düzlemindeki pixellerin konumlarını belirtir.

Koordinatlar arasında dönüşümü sağlamak için 4x4 bir dönüşüm matrisi kullanılır:

$$T = \begin{bmatrix} \text{scaleX} & 0 & 0 & t_x \\ 0 & \text{scaleY} & 0 & t_y \\ 0 & 0 & \text{scaleZ} & t_z \\ 0 & 0 & 0 & t_w \end{bmatrix}$$

Şekil 3.2.3: Dönüşüm matrisi

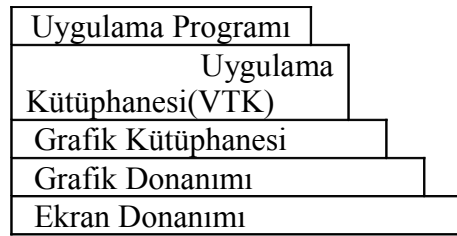


Görüntüyü döndürme işlemi, Visualization Toolkit'te bulunan RotateXYZ() metoduyla yapılabilir. Orijininden geçen herhangi bir vektör etrafında döndürme işlemi yapılabilmesi için 4x4 bir rotasyon matrisi kullanılır.

$$\mathbf{T}_R = \begin{bmatrix} \cos \theta_{x'x} & \cos \theta_{x'y} & \cos \theta_{x'z} & 0 \\ \cos \theta_{y'x} & \cos \theta_{y'y} & \cos \theta_{y'z} & 0 \\ \cos \theta_{z'x} & \cos \theta_{z'y} & \cos \theta_{z'z} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Şekil 3.2.4: Rotasyon matrisi

Bilgisayar grafikleri ve görüntüleme işlemleri, donanıma tamamen bağımlıdır. Daha üst seviye programlar, oluşturdukları uygulama kütüphaneleriyle bilgisayarın grafik yeteneklerinin kullanımı için bir arayüz sağlar. Visualization Toolkit de bu kütüphanelerden biridir. Görselleştirme işleminin çok daha az sayıda komutla gerçekleşmesine olanak tanır. Bilgisayar grafiklerinde görüntü, piksellerin bir araya gelmesi ile elde edilir. Ancak, bu yüksek düzey programlar ile birlikte, grafiklerin oluşturulmasında, nokta, doğru, üçgen ya da çokgenler kullanılır.



Şekil 3.2.5: Grafik Arayüz Hiyerarşisi

## Veri Görselleştirme İşlemi İş Hattı

Görselleştirme işleminde, veriyi grafik bileşenlere dönüştürmeyi daha iyi anlamak için bir akış modeli düşünülürse, but model iki aşamaya ayrılabilir:

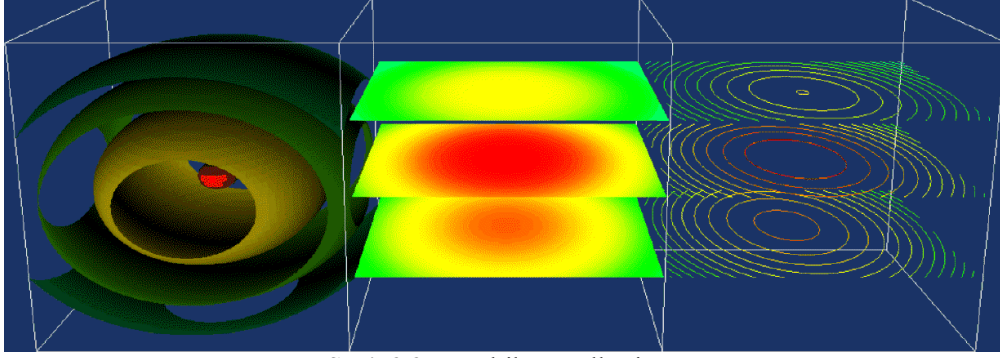
- Dönüştürme
- Sunuş

Dönüştürme, veriyi orjinal formatından grafik bileşenlere –bilgisayar görüntülerine- çevirme işlemidir. Sunuş ise veriyi görüntüye çevirmek için gerekli veri yapılarını ve grafik bileşenleri kullanarak veriyi sunma işlemidir. Nesne tabanlı bir bakış açısıyla, dönüştürme, fonksiyonel modeldeki bir işlemken sunuş, nesne modelindeki nesnedir. Bu yüzden görselleştirme modeli, fonksiyonel model ve nesne modeli ile karakterize edilebilir.

Bu modellerin açıklanmasından önce veri görselleştirme işleminin bir örnekle anlatılması kavramayı kolaylaştıracaktır.

$$F(x, y, z) = a_0x^2 + a_1y^2 + a_2z^2 + a_3xy + a_4yz + a_5xz + a_6x + a_7y + a_8z + a_9$$

Yukarıda matematiksel olarak verilen bir kuadrik fonksiyonunun  $-1 \leq x, y, z \leq 1$  aralığındaki görselleştirilmiş hali aşağıdaki gibi olur:

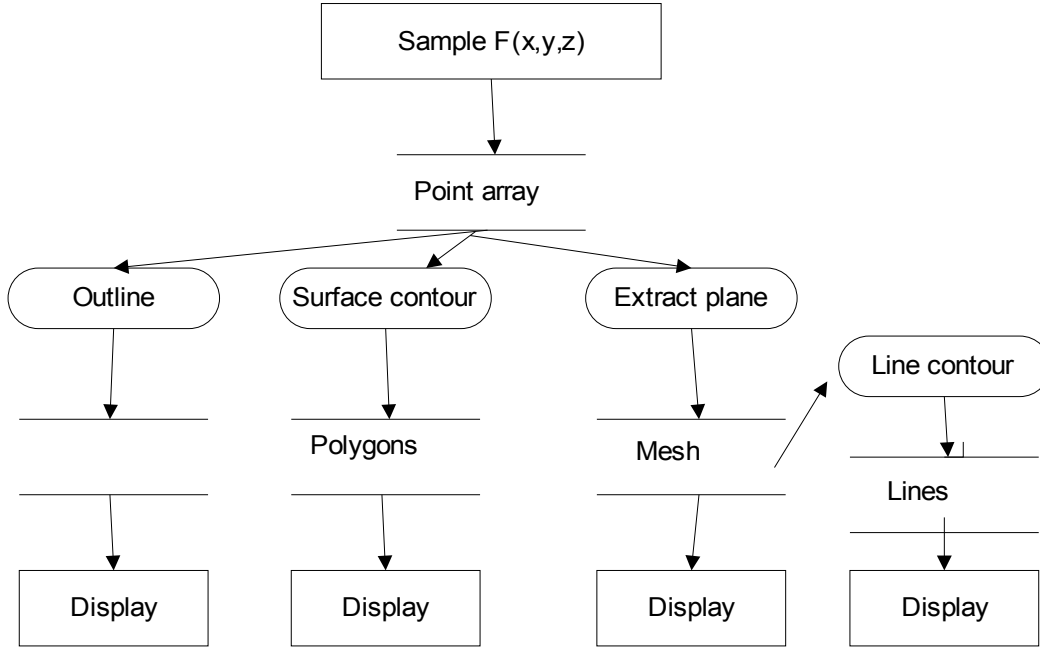


Şekil 3.3: Kuadrik görselleştirme

Bu görüntünün elde edilebilmesi için 3 farklı görselleştirme tekniği kullanılır. İlk olarak,  $F(x, y, z) = c$  ( $c$ , keyfi sabit) fonksiyonuna dayanarak 3 boyutlu yüzey oluşturulur. Sonra, bir fonksiyon değeri tarafından renklenmiş ve veriyi kesen 3 farklı düzlem gösterilir. Son olarak, bu 3 düzlem, sabit değerli çizgilerle biçimlendirilerek gösterilir. [16]

## Fonksiyonel Model

Fonksiyonel model, sistem boyunca veri akışının nasıl olduğunu gösterir. Görselleştirme işlemi için fonksiyonel model aşağıdaki gibidir:



Şekil 3.3.1: Fonksiyonel Model

Şekilde, oval bloklar veri üzerinde yapılan işlemleri, dikdörtgen bloklar veri yığınlarını(objeler) , oklar veri akış yönünü göstermektedir.

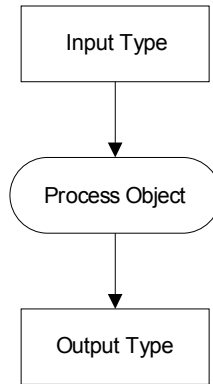
## Nesne Modeli:

Fonksiyonel modelde görselleştirme sırasında verinin akışı açıklanırken nesne modelinde veri üzerinde işlem yapacak modul açıklanır. Nesne modelinde veri yığınları ve işlemler için ayrı nesnelere kullanılır. Bu modelin avantajı, görselleştirme algoritmalarından oluşan işlemlerin veri yapılarına tam giriş iznine sahip olmasıdır. [11]

## Görselleştirme iş hattı

İş hattı, veriyi sunan nesnelere(data objects), veri üzerinde işlem yapan nesnelere(process objects) ve veri akış yönü göstergelerinden oluşur.

- Veri nesnelere(data objects): Veri nesnelere bilgiyi temsil ederler. Aynı zamanda, bilgiyi yaratmak, silmek ve bilgiye girişi sağlayan methodları sağlarlar.
- İşlem nesnelere(process objects): İşlem nesnelere giren veri üzerinde, çıkış verisini oluşturmak için işlem yaparlar. İşlem nesnelere, ya girdi olan veriden yeni veri üretir ya da girdi veriyi yeni bir forma dönüştürür.



Şekil 3.3.3: Tek-tip sistem

## Programlama modelleri

Görselleştirme sistemleri, insan etkileşimli doğal tasarımlar oldukları için kullanılabilirlikleri kolay olmalıdır. Bunun yanı sıra yeni veriye adaptasyona hazır ve hızlı veri açınmalarına izin verecek şekilde esnek olmalıdır. Bu özellikleri sağlayacak çeşitli programlama modelleri geliştirilmiştir. Görselleştirme uygulamalarının en yüksek seviyesi uygulamalardır. Görselleştirme uygulamaları, uygulama alanlarına göre özelleşmiş ve kullanıcı arayüzlü olarak atarlanmıştır. Uygulamalar kullanımı kolay fakat esnekliği düşüktür. Buna karşın programlama kütüphaneleri, kütüphane tabanlı veri yapıları üstünde işlem yapan prosedürlerin bütünüdür. Genelde bu kütüphaneler C ve

FORTRAN gibi programlama dilleriyle yazılmışlardır. Bu özellik diğer programlama araç ve teknikleriyle birleştirilebilecek şekilde esneklik sağlar.

Alternatif görsel programlama modelleri olarak, scene graphs ve spreadsheet model de kullanılabilir. Scene graphs, nesnelere, düğümleri sıralı bir şekilde temsil eden çevrimsiz bir ağaç yapısındadır. Spreadsheet modelinde daha kompleks operasyonları eklemek, çıkarmak ve yürütmek için tek bir programlama dili kullanır. Hesaplanan görsel çıktı hücrelerde gösterilir.

## Temel Veri Gösterimleri

Veriyi görselleştirmek için verinin yapısı hakkında bilgi sahibi olmak gerekmektedir. Bu bilinç, yararlı veri modelleri ve güçlü görselleştirme sistemleri yaratmak için yardımcı olur. Veri düzenli(regular) veya düzensiz(irregular) yapıda bulunabilir. Düzenli veri, veri noktaları arasında bir kalıtım ilişkisine sahiptir. Bu yapıdaki veriyi oluşturan bir noktalar kümesinde bütün noktalar koordinatları tutulmaktansa başlangıç yeri, noktalar arasındaki uzaklık ve nokta sayısının tutulması yeterlidir. Düzensiz tipteki verilerde bilgi, hızlı değişimlerde yoğun, değişimlerin az olduğu durumlarda seyrek gösterilir.

## İleri Seviye Görüntüleme Teknikleri

Hacimsel görüntü elde edilmesi için farklı teknikler kullanılabilir. Dokulara verilen şeffaflık dereceleri ya da alfa değerleri sayesinde, sadece opak yüzeyler değil; saydam yüzeyler de görüntülenebilir. Aynı zamanda, yüzeyler saydamlaştırılarak, iç bölgeleri de görüntülenebilir. Bu teknik, hacimsel görüntülemeye olanak tanır. Alfa değeri, (0,1) aralığında değişir. 0 değeri tamamen saydam bir nesneyi, 1 değeri ise tamamen opak bir nesneyi belirtir. Hacimsel görüntüleme tekniğinde, şeffaflık değeri kadar verinin sınıflandırılması da önemlidir. Örneğin medikal veri kümelerinde, veri, dokuların yoğunluklarına göre sınıflandırılır ve tabakalara ayrılır; bu sınıflar, kemik, kas ve cilt gibi dokuları belirtebilir. Her bir tabakaya bir renk ve şeffaflık değeri verilir. Ayrıca, hacimsel ışıklandırma, gölgelendirmeyi destekleyecek şekilde tanımlanmalıdır. [1]

Hacimsel görüntüleme tekniklerinden biri, görüntü tabanlı tekniktir. Bu teknikte temel fikir, kamera parametrelerine göre ortama bir ışın göndererek, bu ışının ulaştığı her pikselin renk değerinin, üzerine düşen ışık yoğunluğuna göre belirlenmesidir. Bir ışın boyunca veriyi izlemek için, cisim, sabit aralıklarla örneklenebilir ya da ışının geçtiği her bir hacim elemanı(voxel) incelenerek cisimden geçen ışınların ayrıık ifadeleri elde edilebilir. [17],[1]

Bir diğer hacimsel görüntüleme tekniği ise, nesne tabanlı tekniktir. Bu teknik, hacim elemanlarının veri kümesindeki konumlarına ve kamera parametrelerine göre, cisimden alınan örnekleri işler. Verilen opaklık değerine bağlı olarak, hacim elemanları önden arkaya ya da arkadan öne doğru belirli bir sırayla işlenir. [12]

Bir veri kümesindeki, görüntülenecek ilgili nesnelere sınıflandırılması, hacimsel olarak görselleştirilmiş bir görüntü elde etmede çok önemli bir adımdır. Örneğin, bir CT veri kümesindeki örnek bir verinin sınıflandırılması, dokulara yoğunluğu belirten bir eşik değeri verilerek sağlanır. Bir hacimsel elemanın yoğunluk değeri, örneğin kemik dokusu için belirlenen

eşik değerinden büyükse, bu doku, kemik olarak sınıflandırılır. Bu işlem, tüm olası doku yoğunluk değerleri için uygulanırsa, bir transfer fonksiyonu olarak ikili adım fonksiyonu(binary step function) elde edilir. Transfer fonksiyonunun görevi, bir hacimsel elemandaki bilgiyi, renk, opaklık, doku gibi farklı değerlerle eşleştirmektir. Böylece, medikal veri kümelerinin hacimsel görüntülemesiyle elde edilen görüntüdeki dokular arası geçişler, kemik, kas, cilt gibi dokuların farklılaşması, bu dokulardan istenilenlerin görüntülenmesi ya da farklı şeffaflık dereceleri verilerek tüm iç organların görüntülenmesi sağlanır.

Hacimsel görüntüleme işlemi, 3 boyutlu veriden, 2 boyutlu bir görüntü elde etme; aynı zamanda 3 boyutlu yapıyı görselleştirme işlemi olduğundan, bunun sağlanabilmesi için gölgelendirme ve ışıklandırma tekniklerinden yararlanır. Işıklandırma tekniklerini uygun şekilde uygulamak için, bir piksele ulaşan ışık şiddeti  $I'$ 'yi ışığın izlediği yolun integrali ile tanımlayan bir iletim teorisi ışıklandırma modeli kullanılabilir. [12]

Hacimsel bir veri kümesini görüntülemek, sayısal hesaplamaları yoğun bir işlemdir. Görüntülenecek hacimsel verinin her üç boyuttaki büyüklüğü  $n$  olarak ve görüntüleme işlemi boyunca her birim hacimsel elemanın işleneceği kabul edilirse, hacimsel görüntüleme işleminin karmaşıklığı,  $O(n^3)$ 'tür. Ancak, veri kümesindeki boşlukları belirten bölgelerin ya da ilgi konusu olmayan verinin elenerek, yalnızca ilgi konusu olan verinin görüntülenmesi yoluyla işlem süresi kısaltılabilir. Boşluk atlama(space leaping) olarak adlandırılan etkinlik artırma teknikleri, verinin son oluşturulacak görüntüye katkısı olmayacak olan bölgelerini işlemeyerek görüntüleme algoritmalarını hızlandırma teknikleridir. Bu amaçla kullanılan bir teknik, hiyerarşik olarak, verinin tüm önemli bölgelerini tutan bir sekizli ağaç yapısı kullanır. Ağaç yapısının kök düğümü, tüm hacimsel veriyi tutar. Kökün 8 çocuğu vardır; bu çocuk düğümlerin her biri, hacmin  $1/8$ 'ini tutar. Bu bölme işlemi, düğümler cismin homojen bir bölgesini ifade edene kadar sürer. Görüntüleme işlemi sırasında, yalnızca ağacın boş olmayan yaprak düğümleri dikkate alınır. [15]

Çiftli(stereo) görüntüleme tekniği, 3 boyutlu nesnelere görüntülemeye en etkili tekniklerden biridir. 3 boyutlu nesnelere iki göz de kullanılarak bakıldığı ve bu iki gözün de gördüğü resimler açısız olarak birbirinden farklı olduğu için, beyin bu iki görüntüyü işleyerek, nesnelere görüş alanındaki uzaklıklarını belirler. Görüntülemeye bu türden bir işlem için, kamera parametrelerine ve iki göz arasındaki uzaklık bilgisine ihtiyaç duyulur. İki gözden elde edilen görüntüler, kullanılan tekniğe göre zaman paylaşımolu olarak ya da aynı anda görüntünün bulunduğu ekranı paylaşarak gösterilebilir. Her iki yöntemle de, hem sağ hem de sol gözden görülen görüntü, gösterilmiş olur.

## Görüntü İşleme

Veri görselleştirmede, görüntü işleme, takip eden adımda yapılacak olan yorumlama ve işlemi geliştirmek amacıyla, görüntünün içeriğini değiştirmek için kullanılır. Görüntü işleme teknikleri kullanılarak, görüntü gürültüden arındırılabilir ve otomatik ya da yarı otomatik bölütleme yapılabilir.

Ağ düzgünleştirme(mesh smoothing) tekniği, bir veri setinin noktasal koordinatlarını düzgünleştirilen bir görüntü işleme tekniğidir. Bu tekniğin amacı, görüntülenene verinin şeklini, görünüşünü düzgünleştirmektir. Bir veri kümesi, yapıdan(topoloji ve geometri) ve veri

özelliklerinden oluşur. Düzgünleştirme işlemi ile, veri kümesinin topolojisinde hiçbir değişiklik meydana gelmez; yalnızca geometrisi değişir. Bu teknik ile, yüzey gürültüsü yok edilir, yüzeylerin görünüşü iyileştirilir. Bunun için kullanılan temel ve tkin bir teknik, Laplace düzgünleştirme tekniğidir. Bu teknik, yüzey eğriliğini azaltacak ve yüzeyi düzleştirecektir.

Yapay dokuların istatistiksel özelliklerine dair sahip olunan bilgi, filtrelerin veriye en az etki ederek gürültüyü seçmesine ve yok etmesine olanak tanır. Örneğin, gürültü, frekans spektrumunda eşit bir dağılım gösterirken, yaygın görüntülerin gücünün büyük bir kısmı düşük frekanslarda yoğunlaşmıştır. Bu durumda, alçak geçiren filtreler, gürültüyü büyük oranda giderirken orijinal görüntünün büyük kısmını bozulmamış şekilde bırakır. Alçak geçiren filtreler, bir Gasuss çekirdeği ile konvolüsyon şeklinde uygulanabilir.

Basit düzgünleştirme yardımıyla gürültüyü giderme ile ilgili bir problem, sınırların bulanık olmasıdır. Yüksek frekanslar, görüntülerin yalnızca küçük bir parçasını oluşturmasına rağmen, insanın görme sistemi, kenarların ve sınırların uzamsal formunda, yüksek frekanslara karşı çok ince bir duyarlılığa sahiptir. Bir görüntüdeki düşük frekansların büyük kısmı, görme sisteminde, retinayı terk etmeden bile elenmektedir. Kenarları koruyan bir düzgünleştirme yaklaşımı, yöne bağımlı yayındır. Bu filtre, görüntünün görece olarak daha düz olan bölgelerini düzgünleştirir; sert geçişlerde yayını yapmaz. Yayını, gürültü istenen düzeyde azaltılana kadar tekrarlanır. İki olası yayını kriteri vardır: eğim büyüklüğü, belirlenen değerin altındaysa ya da iki piksel arasındaki fark belirlenen bir sabitten küçükse yayını yapar.

Medyan bir filtre, kenarları korurken aynı zamanda düzgünleştirir. Bu filtre, her pikseli, o pikselin komşularının yoğunluk değeri ile belirlenen medyan değeri ile değiştirir. Medyan filtreler, oluşma olasılığı düşük olan yüksek genlikli gürültüler üzerinde etkindir. Giderilen gürültünün miktarını ve boyutunu kontrol etmek için iki yol vardır: komşu piksellerin genişliği değişebilir ya da filtre birden fazla kez uygulanabilir

### **3-Boyutlu Görselleştirme Uygulama Alanları**

Proje boyunca görselleştirme tekniklerinin tasarım ve gerçekleştirilmesi öğrenilirken bu kavramların tıbbi görüntüleme, ticari görselleştirme, modelleme ve algoritma görselleştirme gibi alanlardaki uygulamaları geliştirilmiştir.

Görselleştirme tasarımı işlemi, her seferinde benzerlik göstermektedir. İlk önce, veri okunur ve Visualization Toolkit veri tiplerinden bir tanesine dönüştürülür. İkinci adımda, uygulama boyunca ilgili veri için görselleştirmenin seçimi yapılmalıdır. Son adımda, kullanıcının veriyi anlayabilmesi için fiziksel ve soyut bileşenler bir araya getirilerek bir görselleştirme yaratılır.

### **Medikal Görüntüleme**

Radyoloji, insan anatomi görüntüleriyle ilgilenen bir bilim dalıdır. Bu görüntüler, X-ray, CT, MRI ve ultrason gibi çeşitli tıbbi görüntüleme aygıtından elde edilir. Genelde radyologlar iki boyutlu görüntülerle çalışırlar fakat bazı durumlarda üç boyutlu model kullanımı kaçınılmaz olmaktadır. Özellikler cerrahi operasyonların planlanması ve geliştirilmesinde üç boyutlu

modeller büyük avantaj sağlamaktadır. Bu durumda, CT verileri öne çıkar. Bir CT görüntüsü, grinin tonlarını – hava için siyah, yumuşak doku için gri, kemik için beyaz- içerir.



Şekil 3.6.1: İnsan kafatası için bir CT kesiti

Bir organdan yatay olarak alınan CT kesitleri (dataset- veri kümesi) tıbbi araştırmalara ortam oluşturmak, cerrahi eğitim ortamları hazırlamak gibi sebeplerle Visualization Toolkit teknikleri ile görsel bilgilere dönüştürülürler. Bu tekniklerden Isocontouring tekniği ile deri ve kemik yüzeyi çıkartılabilir. ( 600 yoğunluklu doku hava/deri, 1150 yoğunluklu doku yumuşak alan/kemik sınırlarını belirler.)

Medikal görüntü dilimlerinin boyutlarının, başlangıç noktasının ve veri boşluklarının bilinmesi yeterlidir. Bu bilgilerine sahip olan veriyi, görselleştirmek için genel olarak kullanılan adımlar 3-boyutlu görselleştirme teknikleri adımlarıdır.

1. Veriyi oku
2. Herbir anatomik özellik için bir isosurface oluştur
3. Modelleri hasta uzayından dünya uzayına dönüştür
4. Modelleri görüntüle

### 3.6.1.1. Veri Okuma

Medikal veri daha önce de anlatıldığı gibi değişik formatlarda olabilir. Bir örnek ve son ekten oluşan dosya isimlendirilmiş dilimler, VTK kütüphanesinde ilgili veriyi okuyabilen nesnelere okunur (vtkPNMreader, vtkDICOMImageReader..). VTK kütüphanesinde desteklenmeyen bir veri formatı okunmak istenirse kullanıcı kendi nesnesini yazabilir. Yazılacak nesnenin bir kurucu fonksiyonu, Execute() vePrintself() fonksiyonlarının olması zorunludur. Bunların yanısıra, kullanıcı ihtiyac duyduğu methodları nesneye ekler.

### 3.6.1.2. Isosurface oluşturma

Isosurface görselleştirme için volume rendering, marching cubes ve dividing cubes tekniklerinden biri seçilebilir. Bu tekniklerden volume rendering diğerlerine kıyasla daha çok vakit almaktadır.

Donanım tabanlı çokgen görüntüleme veya çıkartılan yüzey içinde hareket işlemi yapılacaksa marching cubes tekniği tercih edilir. Dividing cubes ise yazılım tabanlı görüntüleme için daha uygundur. [14]

Medikal cisimler için marching cubes birçok üçgen üretir. Bu yüzden düşük çözünürlüklü veri kümelerinin kullanılması bu tekniğin verimliliğini artırır. Proje çerçevesinde vtkMarchingCubes, varsayılan filtre olarak kullanılmıştır.

Visualization Toolkit, isocountouring dışında vtkStructuredPointsGeometryfilter gibi teknikler de sunmaktadır. Bu filtre, radyologların aşına olduğu aksiyel, sagittal ve koronel eksenlere göre dikey düzlemlerin çıkartılmasında kullanılır.

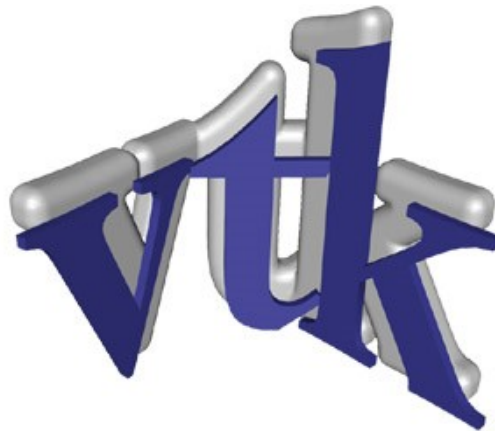
## Ticari Görselleştirme

3 boyutlu veri görselleştirme tekniğinin ticari alanda uygulanması medikal alandakine göre nispeten daha yenidir. Son yıllarda artan bilgi hacmi ve 3 boyutlu grafik ve tekniklerin daha interaktif hale gelmesi 3 boyutlu görselleştirmeyi daha önemli kılmıştır. Bu teknik ile günümüzün karmaşık verileri daha anlaşılır bir hale getirilirken işleme süreleri de oldukça kısalmıştır. Görselleştirme işlemi yapılırken elde bulunan sayısal(finsansal) veriler vtk veri dosyasına çevrilir. Uygun hale getirilen veri medikal görselleştirmeye benzer yöntemlerle (VTK kütüphane fonksiyonlarıyla) okunur ve görüntülenir.

## Implicit Modelleme

Visualization Toolkit, geometrik modelleme yeteneğine de sahiptir. Bunlardan en önemlilerinden biri implicit modellemedir. Özellikle logo tasarımlarında bu modelleme kullanılır.

Bir logo tasarlanmak istendiğinde harfler, vtkTransformpolyDataFilter ile birbirlerine göre göreceli pozisyonları ayarlanır. Sonra, vtkAppendpolyData ile dönüştürülmüş harflerden gelen bütün çokgenler bir çokgen kümesi içinde birleştirilir. Bu işlemin ardından vtkImplicitModeller ile bir hacimsel veri kümesi oluşturulur. Oluşturulan bu veri kümesinden vtkContourfilter ile logo şekli çıkartılır. [13]

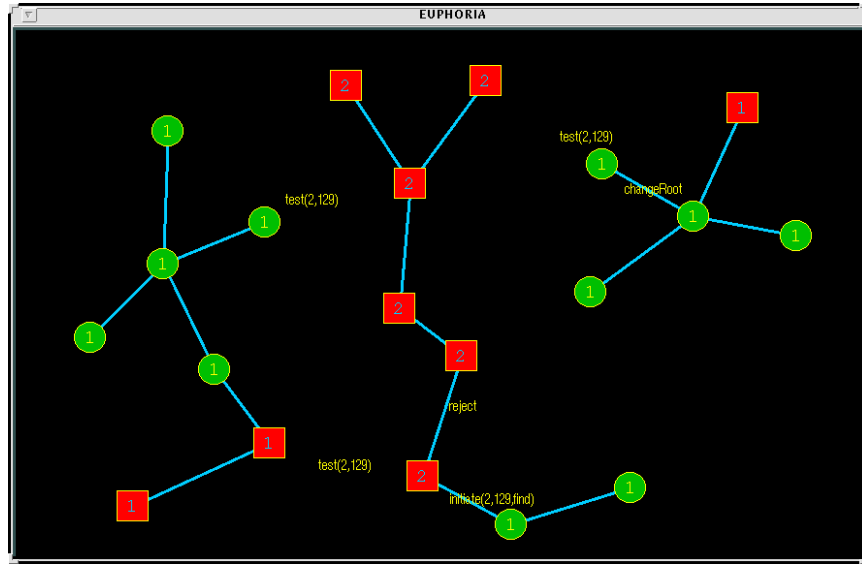


Şekil 3.6.1.4: vtkImplicit ile oluşturulan bir logo



## Algoritma görselleştirme

Yukarıda anlatılan alanların yanısıra görselleştirme, algoritma ve veri yapılarının görsellenmesi için de kullanılmaktadır. Bu yaklaşım ile sunulan dosya, dizini ve dosya-dizin arasındaki ilişki 3 boyutlu modellenir. Benzer bir yaklaşım yığın, kuyruk, bağlantılı liste, ağaç ve diğer veri yapıları içinde görsellenebilir.



Şekil 3.6.1.5: Visualization of an algorithm

# ANALİZ VE MODELLEME

Kuramsal olarak açıklanan veri görselleştirme işlemi, temelde, dört temel görevden oluşur:

- Verinin okunması
- Verinin orijinal formatından bilgisayar grafiklerine dönüştürülmesi
- Verinin görüntülenmesi
- Görüntünün, kullanıcıdan alınan parametrelerle işlenerek, yüzeysel ve hacimsel görselleştirme yöntemleriyle tekrar şekillendirilmesi

CT ve MRI verilerinin DICOM veya PNG formatında veri kümeleri olarak alınıp, visualization toolkit kütüphanesindeki özel veri okuma fonksiyonları kullanılarak verinin programda işlenebilir hale getirilmesi ile veri okuma işlemi tamamlanır.

Veri dönüştürme(transformation) işlemi, 4x4 boyutlarındaki transformasyon matrisleri yardımıyla, bilgisayar grafiklerinin dört önemli koordinat sistemi arasındaki dönüşümlerle gerçekleştirilir.

Verinin görüntülenmesi işlemi, okunan ve dönüştürülen verilerin bilgisayar ekranında gösterilmesidir. Görüntüleme teknikleri, 2 boyutlu boyama programlarından 3 boyutlu karmaşık tekniklere kadar çeşitlilik gösterir. Görüntü tabanlı ve nesne tabanlı olmak üzere iki kategoriye ayrılacak görüntüleme teknikleri kullanılarak, verilerin görüntüsü, bilgisayar ekranında oluşturulur.

3 boyutlu verilerin görüntülenmesi için, yüzeysel veya hacimsel görüntüleme tekniklerinden biri kullanılmalıdır. Yüzeysel görüntüleme tekniğinde, cisimlerin iç kısımları ile ilgilenilmez; sadece yüzeyi görüntülenir. Bu teknikte, aynı yoğunluk değerlerine sahip dokular birleştirilerek yüzeyler oluşturulur. Hacimsel görüntüleme tekniğinde ise, cisimlerin hem yüzeyleri hem de iç kısımları görüntülenir. Bu teknikte, dokulara farklı opaklık değerleri verilerek, iç kısımlarının da görülebilmesi sağlanır.

Elde edilen görüntünün işlenmesi aşaması, verinin daha anlaşılır kılınması, daha iyi ifade edilmesi ya da görüntünün gürültülerden arındırılarak netleştirilmesi, istenilen forma sokulması, daha güzel görünmesi gibi amaçların gerçekleştirilmesi için önemli bir aşamadır. Veri görselleştirme işleminin, verinin daha anlaşılır ve görülebilir forma sokulması amacıyla yapıldığı göz önüne alınırsa, görüntü işleme, veri görselleştirmenin vazgeçilmez bir parçasıdır.

Görüntü işleme aşamasında, renklendirme, ışıklandırma, düzgülendirme, bölütleme gibi işlemlerle görüntü istenilen forma sokulur. Renklendirme ile, görüntünün istenilen dokuları, istenilen renklerde gösterilir. VTK kütüphanesindeki renklendirme fonksiyonlarına RGB ve HSV renk sistemlerine ait değerler parametre olarak verilerek istenilen renkler elde edilir. Işıklandırma işlemi için yerel bir ışık kaynağı ya da sonsuz bir ışık kaynağından gelen ışınların etkisi görüntülenene cisme yansıtılır. Kameranın yönelimi ve ışığın geldiği doğrultu belirlenerek görüntülenene cisim gölgelendirilir. Düzgülendirme işlemi ile, görüntü, gürültülerden arındırılır, netleştirilir ve yüzeylerin daha düzgün görünmesi sağlanır. Düzgülendirme için, Gaussian ve medyan düzgülendirme yöntemleri mevcuttur. Bunun için de uygun filtreler ve örnekleme

yöntemleri kullanılır. Bölütleme, bir resmin veya cismin pixellerinin sınıflandırılması işlemidir. Bu işlem, dokuların ayrıştırılmasında kullanılır. Örneğin, medikal bir hacimsel veride, kemik, kas ve cilt dokularının ayrıştırılarak görüntülenmesi işlemi, bölütleme yardımıyla gerçekleşir. Bu tekniklerin uygulanması, VTK kütüphanesindeki fonksiyonlar kullanılarak gerçekleştirilmiştir.

Görüntüleme ve görüntü işleme aşamasında, kullanıcı ile etkileşim sağlanmıştır. Bunun için bir kullanıcı arayüzü tasarlanmıştır; görüntülenecek veri kümesi, yüzeysel ya da hacimsel görüntüleme tercihi, görüntülenmek istenen dokuların özellikleri ve renk özellikleri kullanıcıdan alınan değerler ile belirlenmiştir.

# TASARIM, GERÇEKLEME VE TEST

## Çalışma ortamı

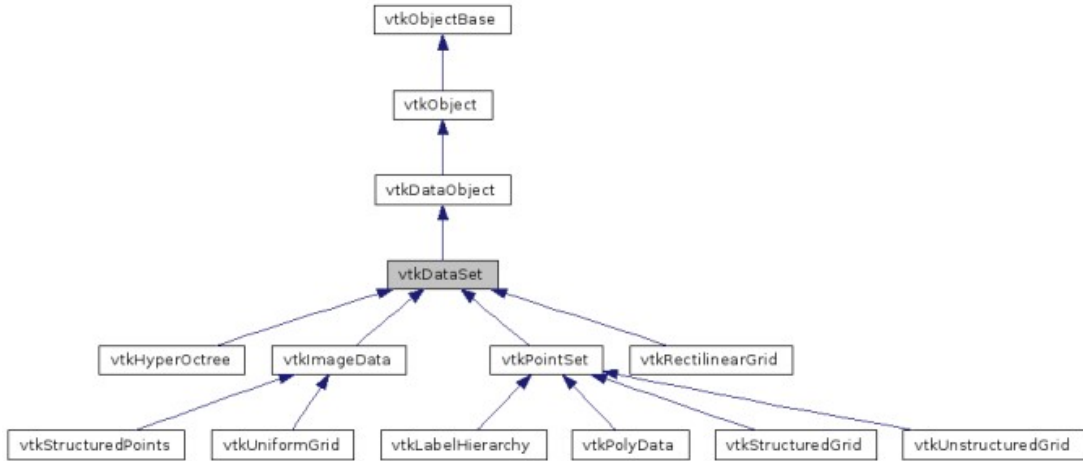
Proje geliştirme platformu olarak, Microsoft Visual Studio 2008 kullanılmıştır. Görselleştirme için uygun sınıf ve fonksiyonlar sunan ve açık kaynaklı bir yazılım olan Visualization Toolkit 5.2.1. kütüphanesi kullanılmıştır. Bu kütüphanenin donanımdan bağımsız olarak kullanılması ve grafik özelliklerinin her donanımda uygun gösterilmesi için yine açık kaynaklı Cmake 2.6 derleyicisi kullanılmıştır. Visualization Toolkit Tcl, Python, Java, C++ dilleri ile birlikte kullanılabilir; ancak grup elemanları C++ dilinde daha tecrübeli olduğu için, programlama dili olarak, C++ dili kullanılmıştır.

## Veri kümesi

Projede, dicom(Digital Imaging in Communications and Medicine) formatındaki CT ve MRI görüntülerinden elde edilmiş veri kümeleri ile çalışılmıştır. Bu veri kümeleri, insan vücudunun parçalarından belirli milimetrik aralıklarla CT ile alınan kesitlerden oluşur. Bu veri kümelerinin boyutları, kesitleme aralığına, alınan kesit sayısına ve görüntü kalitesine göre çeşitlilik göstermektedir. Örneğin, bir göğüs kafesinin görüntüsünü oluşturan veri kümesi, 341 dicom dosyasından oluşup, 11.3 MB boyutundayken; bir çift ayak görüntüsünü oluşturan veri kümesi, 250 dicom dosyasından oluşup, 125 MB boyutunda olabilmektedir. Veri kümeleri arasındaki bu boyut ve dosya sayısı farkları, kullanılan görüntüleme algoritması ile birlikte, görüntüleme süresini belirleyen temel faktörlerdendir.

Veri kümesi, her biri ayrı dicom dosyası olarak saklanan dilimlerden oluşur. Bu dilimleri ifade eden dicom dosyaları, hepsi için aynı olan bir önek ve dilimin sırasını belirten bir sonek ile isimlendirilir. Örnek olarak bir veri kümesindeki dosyalar, sırasıyla; IM\_00001, IM\_00002, ... , IM\_00250 gibi isimler alır.

VTK'nın okuyabildiği veri kümeleri, vtkDataSet sınıfı ile ifade edilir.

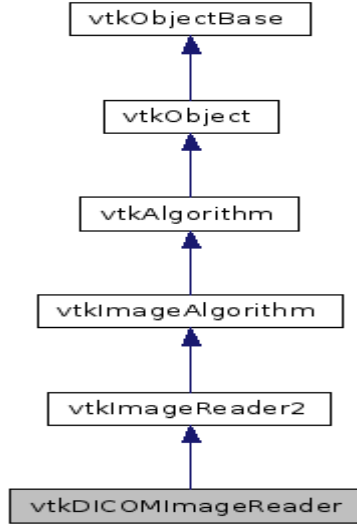


Şekil 5.2: vtkDataSet için sınıf diyagramı

## Veri Kümesinin Okunması

Dicom formatındaki veri kümesi, VTK kütüphanesinin `vtkDICOMImageReader.h` kütüphanesi kullanılarak okunur. VTK kütüphanesinin, png, bmp ya da tiff gibi farklı formatlardaki resim dosyalarını okuyan fonksiyonları mevcut olmakla birlikte; CT ve MRI ile elde edilen tıbbi verilerin genel olarak dicom formatında bulunması sebebiyle, projede dicom veri setlerinin kullanılması tercih edilmiştir. `vtkDICOMImageReader`, hem tek bir dosyayı hem de birçok dosyadan oluşan veri kümesini okuma yeteneğine sahiptir. Parametre olarak dosyanın ya da veri kümesinin bulunduğu klasörün dizinini alır.

`vtkDICOMImageReader` kütüphanesine alternatif olarak, `vtkVolume16Reader` kütüphanesi ile 16 bitlik resim dosyaları okunabilir; bu kütüphane kullanılarak dicom formatında verileri okuyabilen bir sınıf tasarlanabilir. Bu kütüphane, temel bir okuma kütüphanesidir. Ancak böyle bir sınıf oluşturulmasına ihtiyaç duyulmamıştır.

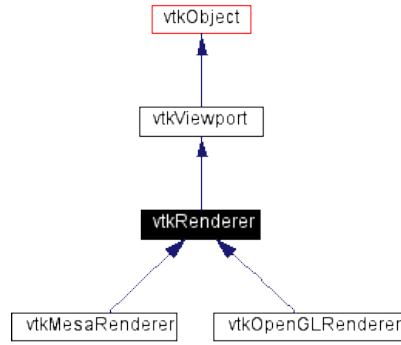


Şekil 5.3: `vtkDICOMImageReader` için sınıf diyagramı

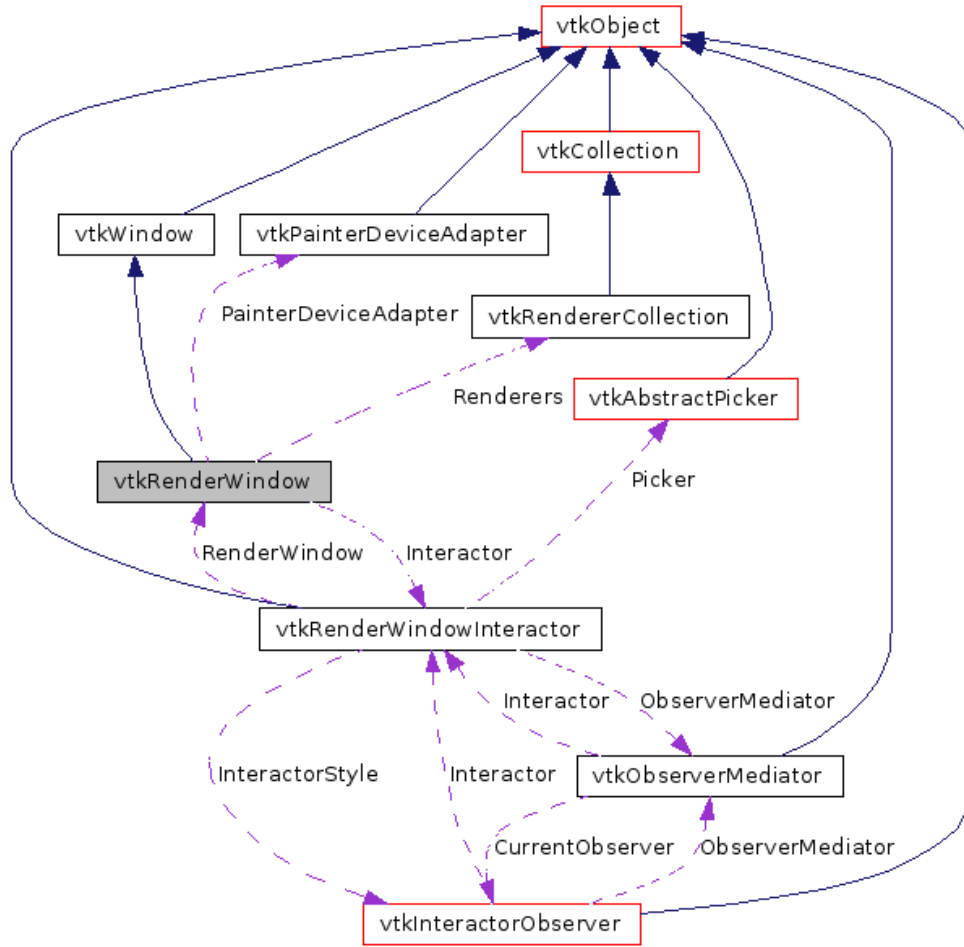
## Verinin görüntülenmesi

Okunan verinin görüntülenmesi için `vtkRenderer` ve `vtkRenderWindow` kütüphaneleri kullanılır.

`vtkRenderer` kütüphanesinin oluşturduğu `render` nesnesi, nesneleri görüntüleme işlemini kontrol eder. Görüntüleme işlemi, bir şekli, ışığı ve kamera görüşünü bir resme dönüştürme işlemidir. `vtkRenderer`, aynı zamanda, dünya koordinatları, görüş koordinatları(bilgisayar grafiklerinin koordinat sistemi) ve görüntü koordinatları(görüntüleyen cihazın ekran koordinatları) arasındaki koordinat dönüşümünü sağlar. Bunların yanı sıra, çift yönlü ışıklandırma gibi belirli gelişmiş görüntüleme özelliklerini de kontrol eder.



Şekil 5.4.1: vtkRenderer için sınıf diyagramı



Şekil 5.4.2: vtkRenderWindow için işbirliği diyagramı

vtkRenderWindow kütüphanesi, renderer nesneleri için, görüntüyü içinde oluşturabilecekleri bir pencere yaratır. vtkRenderWindow, görüntü penceresinin özelliklerini belirleyen soyut bir nesnedir. Bu nesneye ait metodlar, pencere boyutunu belirler, görüntüleme işlemini senkronize eder, stereo görüntüleme(iki gözden elde edilen görüntülerin gösterilmesi) işlemine olanak tanır.

vtkRenderWindowInteractor, platformdan bağımsız olarak, görüntü penceresiyle kullanıcı etkileşimini sağlayan sınıftır. Mouse, klavye ve zaman olayları ile görüntü penceresi arasında etkileşim mekanizması sağlar.

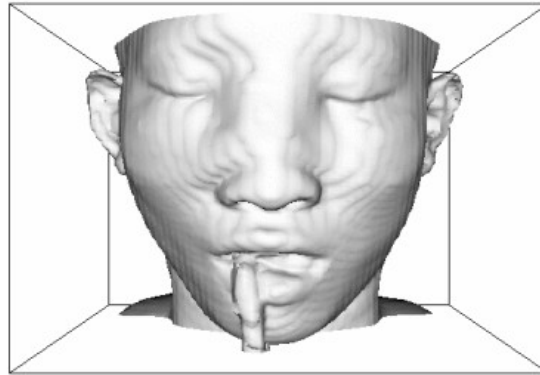
vtkImageViewer sınıfı, 2 boyutlu görüntülerin gösterimi için uygun bir sınıftır. Bu sınıfın oluşturduğu viewer nesnesi, okunan verinin görüntülenmesinde kullanılır.

vtkLight, 3 boyutlu görüntüleme için sanal bir ışık oluşturur. Işığı konumlandırmak, yönlendirmek, açıp kapatmak, parlaklığını ve rengini ayarlamak için metodlar sağlar. vtkCamera sınıfı, 3 boyutlu görüntüleme için sanal bir kameradır. Görüş ve odak noktasını konumlandırmak ve yönlendirmek için metodlar sunar.

vtkActor sınıfı, görüntülenen ekrandaki bir nesneyi temsil eder. Aktör konumlarına ve yönlerine ilişkin fonksiyonları vtkProp sınıfından türetilir. vtkProperty sınıfı, geometrik bir nesnenin yüzey özelliklerini temsil eder. Bu özelliklerin başında, renklendirme, ışığı yansıtma oranı, opaklık ve gölgelendirme gelir. vtkMapper, veriyi temel grafik bileşenlere çevirmek için bir arayüz oluşturan soyut bir sınıftır. Bu eşleme, bir lookup tablosu oluşturmak ve bu tablo yardımıyla veriyi eşlemek üzere bir değer aralığı belirlemek yoluyla kontrol edilebilir.

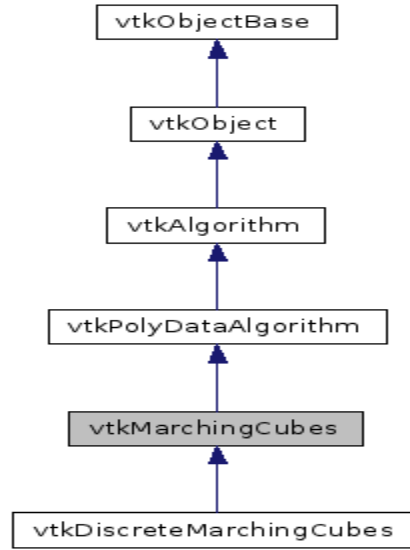
## Yüzey Görüntüleme

Görüntülenmek istenen veri kümesi, farklı yoğunluk değerlerine sahip dokulardan oluşur. Bu dokulardan görüntülenmek istenen dokuya ait yoğunluk değeri belirlenerek vtkMarchingCubes sınıfından oluşturulan extractor nesnesine parametre olarak verilir. Böylece, vtkPolyDataMapper sınıfından oluşturulan mapper nesnesi yardımıyla doku oluşturulur. Buradan elde edilen çıktı, vtkActor sınıfından oluşturulan nesnesine parametre olarak verilerek görüntülenecek nesne oluşturulmuş olur. Bu nesneye istenen renk özelliği atanabilir. Elde edilen görüntü, yüzey görüntüleme yöntemiyle elde edilmiş olduğu için, yalnızca yüzey görüntüsü elde edilir; iç bölge görüntülenmez. Görüntülenecek cismin yalnızca dış yüzeyi görüntülenmek zorunda değildir; örneğin medikal verilerde cilt dokusu görüntülenebileceği gibi, uygun yoğunluk değerleri belirlenerek kemik dokusu ya da damar dokusu da görüntülenebilir.



Şekil 5.5.1:Yüzey görüntüleme ile elde edilmiş baş resmi

vtkMarchingCubes sınıfı, bir cisimden eşyüzeyler(isosurface) oluşturur. Girdi olarak hacimsel bir veriyi ve yüzey için belirlenen yoğunluk değerini alır; bundan eş yüzeyler üretir.



Şekil 5.5.2: vtkMarchingCubes için sınıf diyagramı

## Hacim Görüntüleme

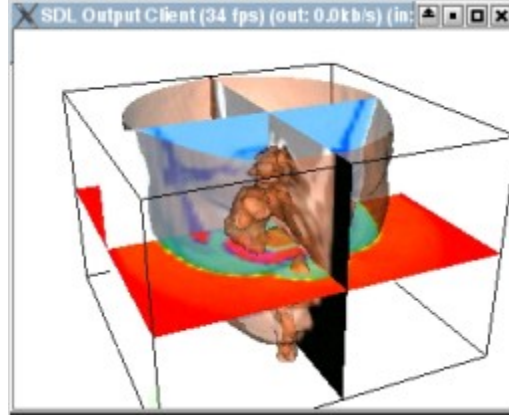
Görüntülenmek istenen cisim, sınıflandırılır ve tabakalara ayrılır. Her tabakanın ayrı doku yoğunlukdeğeri vardır. Belirlenen bu yoğunluk değerlerine göre, katmanlar halinde dokular oluşturulur. Her doku için, vtkMarching Cubes sınıfından extractor nesnelere oluşturulur. Oluşan doku, vtkActor sınıfından nesneye atanarak, renk özelliklerinin yanı sıra opaklık değeri de belirlenerek görüntülenir. Opaklık değeri, (0,1) aralığında değer alır; 0 değeri tam saydamı, 1 değeri tam opağı belirtir. Bu farklı opaklık değerleri sayesinde dış tabakalar saydamlaştırılarak cisimlerin iç kısımlarının da görüntülenmesi sağlanır. Böylece hacimsel görüntü elde edilir.



Şekil 5.6.1:Hacim görüntüleme ile elde edilmiş ayak resmi

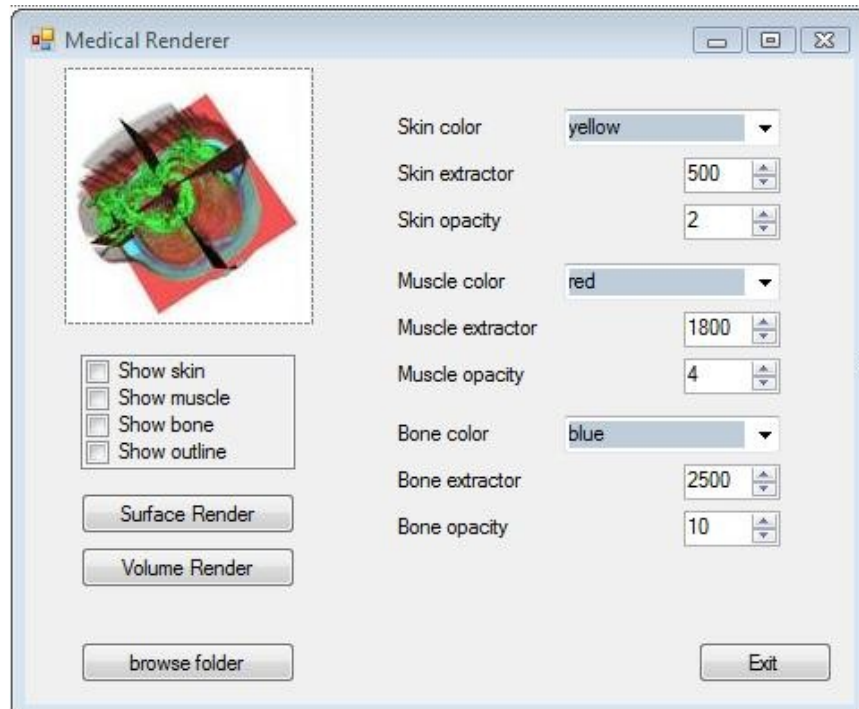


vtkImageDataGeometryFilter sınıfının sunduğu teknikler ile, aksiyel, sagittal, koronel eksenlere göre dikey düzlemlerin çıkartılır; hacimsel olarak elde edilmiş görüntünün bu düzlemler üzerindeki izdüşümleri elde edilir. Her eksen için birer lookup tablosu yaratılır ve eksenleri belirten her bir düzlem için vtkActor nesnesi oluşturulur; bu düzlemlere ait özellikler atanır.



Şekil 5.6.2: Üç düzlemde birleşik görüntü ve saydam deri

Bu uygulamaların, kullanıcı ile interaktif bir yapıya kavuşturulması amacıyla, Visual Studio Windows Forms Application üzerinde, bir kullanıcı arayüzü oluşturulmuştur. Bu arayüzü kullanarak, kullanıcı, yüzey ya da hacim görüntüleme seçeneğini seçebilir, görüntülenecek veri kümesini belirleyebilir, medikal veriye ait kemik, kas ve cilt doku seçeneklerinden görüntülemek istediklerini seçebilir, bu dokulara ait yoğunluk, renk ve opaklık değerlerini belirleyebilir.



Şekil 5.6.3: Kullanıcı arayüzü

## Testler

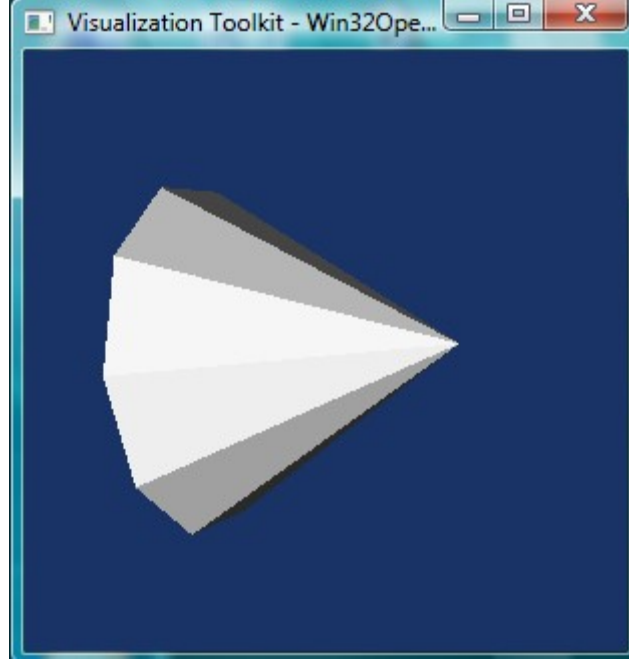
Gerçeklenen uygulamaları test etmek için, iki farklı veri kümesi kullanılmıştır. Bunlardan biri, bir göğüs kafesinin görüntüsünü oluşturan, 341 dicom dosyasından oluşup, 11.3 MB boyutunda olan thorax adlı veri kümesidir. Diğeri ise, bir çift ayak görüntüsünü oluşturan, 250 dicom dosyasından oluşup, 125 MB boyutunda olan vix adlı veri kümesidir.

Bu veri kümeleri üzerinde görüntülenecek dokular seçilerek farklı renk, doku, opaklık değerleri verilmiş, yüzey,hacim görüntüleme ve aksiyel, koronel,sagital eksenlere göre izdüşüm görüntüleme işlemleri ayrı ayrı uygulanmış, elde edilen görüntüler incelenmiştir. Veri kümelerine göre bu yöntemlerin çalışma hızı gözlemlenmiştir.

# DENEYSEL SONUÇLAR

## Geometrik Cisimlerin Görüntülenmesi

Proje geliştirme aşamasından önce bir geometri görüntüleme uygulaması yapılmıştır. Bu amaçla çeşitli geometrik cisimler oluşturulmuş; görüntü penceresi ile mouse ve klavye etkileşimi sağlanmıştır. Yükseklik için 3.0, yarıçap için 1.0 ve çözünürlük için 10 değerleri verilerek oluşturulan koni görüntüsü Şekil 6.1’de verilmiştir.



Şekil 6.1: Koni

## Yüzey Görüntüleme

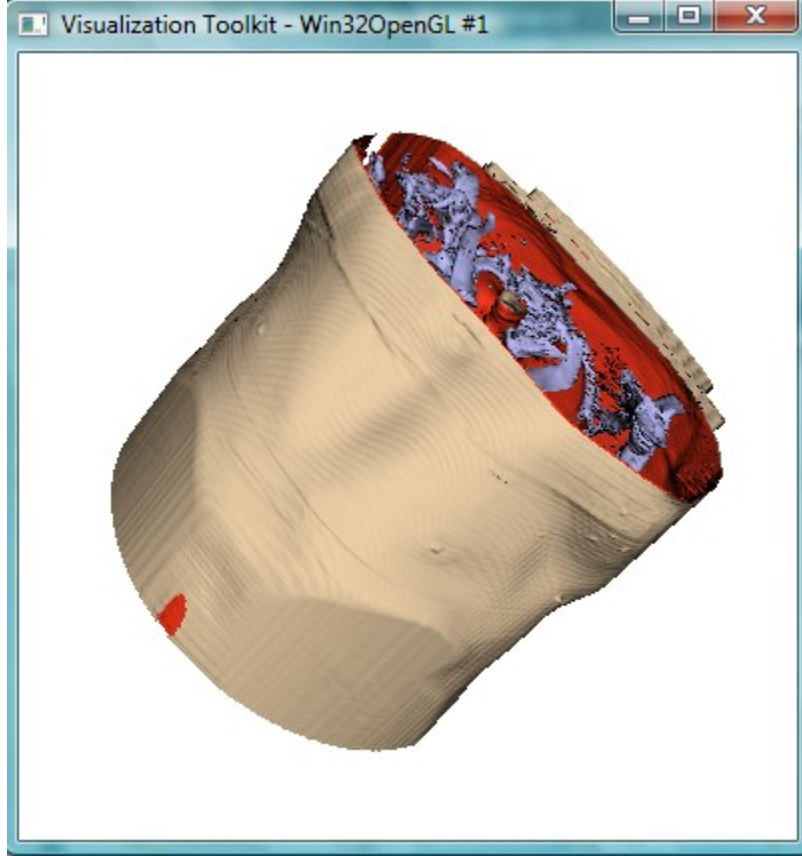
Yüzey görüntüleme uygulaması ile ilgili thorax ve vix veri kümeleri ile deneyler yapılmıştır. Yapılan deneylerden dördünün sonucu aşağıda verilmiştir.

### Deney 1

thorax veri kümesi üzerinde yüzey görüntüleme yapılmıştır. Deri, kas ve kemik dokularına aşağıda belirtilen yoğunluk ve renk değerleri verilmiştir. Elde edilen yüzey görüntüsü, Şekil 6.2.1’deki gibidir. Verinin görüntüsünün elde edilmesi için geçen süre 18 sn’dir.

**Tablo 6.2.1:** Deneyl program kod parçası

```
skinExtractor->SetValue(0,1400);  
skin->GetProperty()->SetDiffuseColor(0.9200, 0.8100, 0.6555 );  
muscleExtractor->SetValue(3,1900);  
muscle->GetProperty()->SetDiffuseColor(0.9999, 0.2222, 0.1111 );  
boneExtractor->SetValue(5,2350);  
bone->GetProperty()->SetDiffuseColor(0.7555,0.7555,1);
```



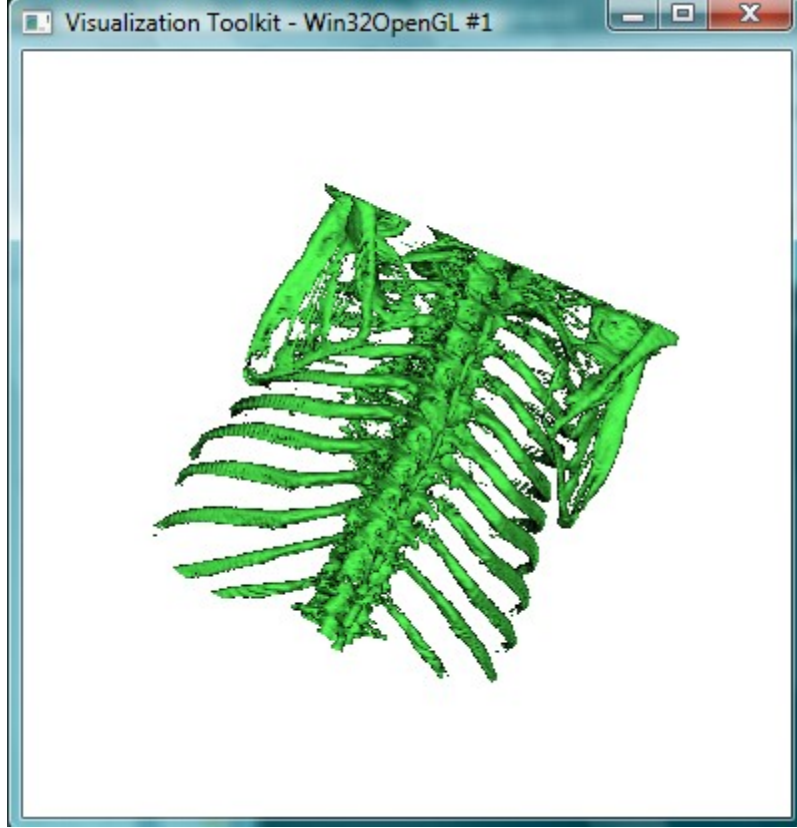
**Şekil 6.2.1:** thorax yüzey görüntüleme

## Deney 2

thorax veri kümesine ait kemik dokusunun yüzeyi görüntülenmiştir. Kemik dokusu için aşağıda belirtilen yoğunluk ve renk değerleri verilmiştir. Elde edilen yüzey görüntüsü, Şekil 6.2.2'deki gibidir. Verinin görüntüsünün elde edilmesi için geçen süre 15 sn'dir.

**Tablo 6.2.2:** Deneyl program kod parçası

```
boneExtractor->SetValue(5,2300);  
bone->GetProperty()->SetDiffuseColor(0.3333,1,0.3333);
```



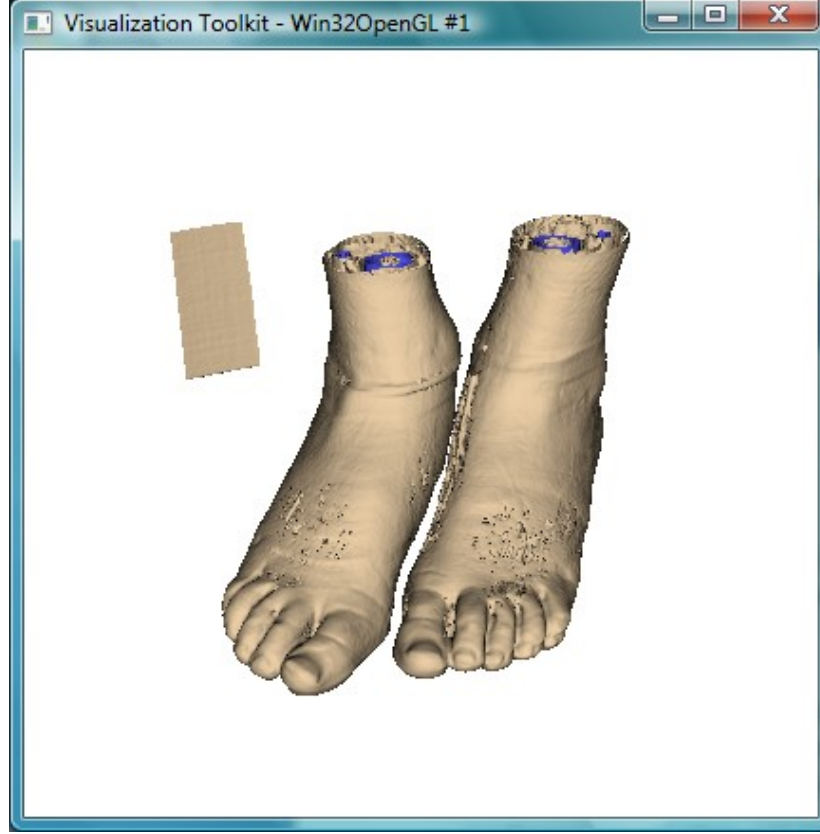
Şekil 6.2.2: thorax kemik yüzeyi

### Deney 3

Yüzey görüntüleme tekniği ile vix veri kümesinin deri ve kemik yüzeyleri görüntülenmiştir. Deri ve kemik dokularına aşağıda belirtilen yoğunluk ve renk değerleri verilmiştir. Elde edilen yüzey görüntüsü, Şekil 6.2.3'teki gibidir. Verinin görüntüsünün elde edilmesi için geçen süre 2 dk 40 sn'dir.

Tablo 6.2.3: Deney3 program kod parçası

```
skinExtractor->SetValue(5,2300);  
skin->GetProperty()->SetDiffuseColor(0.9200, 0.8100, 0.6555 );  
boneExtractor->SetValue(0,800);  
bone->GetProperty()->SetDiffuseColor(0.3333,0.3333,1);
```



Şekil 6.2.3: vix yüzey görüntüleme

#### Deney 4

vix veri kümesine ait kemik dokusunun yüzeyi görüntülenmiştir. Kemik dokusu için aşağıda belirtilen yoğunluk ve renk değerleri verilmiştir. Elde edilen yüzey görüntüsü, Şekil 6.2.4'teki gibidir. Verinin görüntüsünün elde edilmesi için geçen süre 2 dk 12 sn'dir.

Tablo 6.2.4: Deney4 program kod parçası

```
boneExtractor->SetValue(0,300);  
bone->GetProperty()->SetDiffuseColor(0.3333,0.3333,1);
```



Şekil 6.2.4: vix kemik yüzeyi

## Hacim görüntüleme

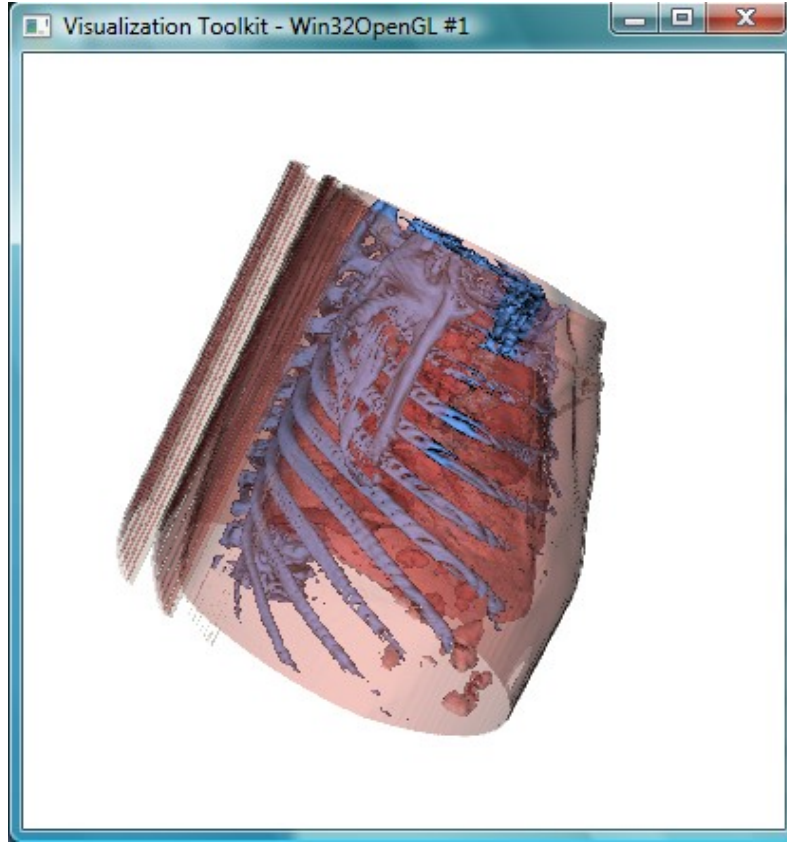
Hacim görüntüleme uygulaması ile ilgili thorax ve vix veri kümeleri ile deneyler yapılmıştır. Yapılan deneylerden ikisinin sonucu aşağıda verilmiştir.

### Deney 1

Hacim görüntüleme işlemi, thorax veri kümesi üzerinde gerçekleştirilmiştir. Deri ve kemik dokularına aşağıda belirtilen yoğunluk, renk ve opaklık değerleri verilmiştir. Elde edilen yüzey görüntüsü, Şekil 6.3.1'deki gibidir. Verinin görüntüsünün elde edilmesi için geçen süre 20 sn'dir.

Tablo 6.3.1: Deney1 program kod parçası

```
skinExtractor->SetValue(0,1200);  
skin->GetProperty()->SetDiffuseColor(0.9200, 0.8100, 0.6555);  
skin->GetProperty()->SetOpacity(0.2);  
muscleExtractor->SetValue(3,1600);  
muscle->GetProperty()->SetDiffuseColor(0.8888, 0.3333, 0.3333);  
muscle->GetProperty()->SetOpacity(0.3);  
boneExtractor->SetValue(5,2300);  
bone->GetProperty()->SetDiffuseColor(0.4444,0.6666,1);  
bone->GetProperty()->SetOpacity(1);
```



Şekil 6.3.1: thorax hacim görüntüleme

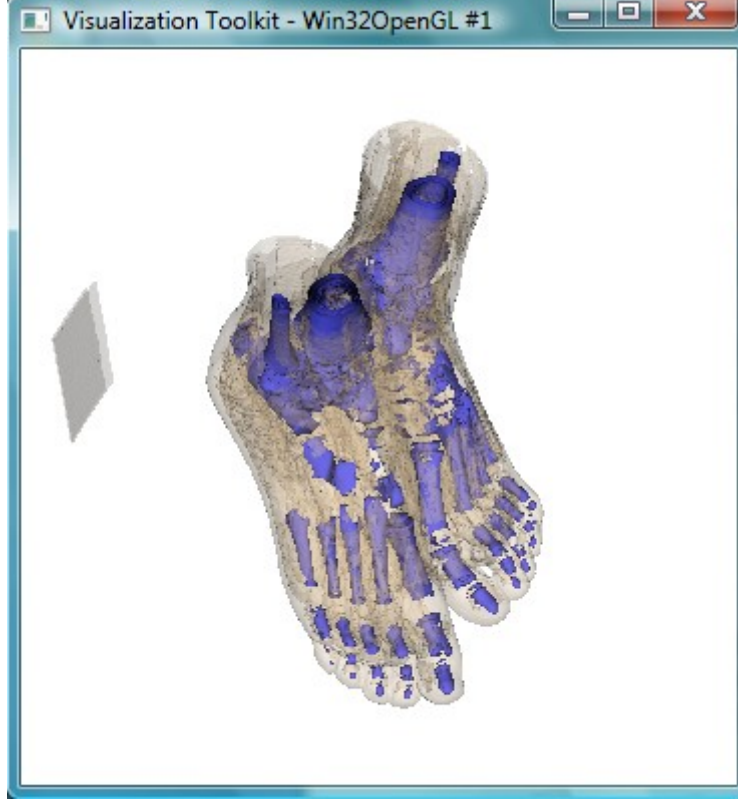
## Deney 2

Hacim görüntüleme işlemi, vix veri kümesi üzerinde gerçekleştirilmiştir. Deri, kas ve kemik dokularına aşağıda belirtilen yoğunluk, renk ve opaklık değerleri verilmiştir. Elde edilen yüzey görüntüsü, Şekil 6.3.2'deki gibidir. Verinin görüntüsünün elde edilmesi için geçen süre 2 dk 29 sn'dir.

Tablo 6.3.2: Deney2 program kod parçası

```
skinExtractor->SetValue(5,2300);  
skin->GetProperty()->SetDiffuseColor(0.9200, 0.8100, 0.6555 );  
skin->GetProperty()->SetOpacity(0.2);  
boneExtractor->SetValue(0,400);  
bone->GetProperty()->SetDiffuseColor(0.3333,0.3333,1);  
bone->GetProperty()->SetOpacity(1);
```





Şekil 6.3.2: vix hacim görüntüleme

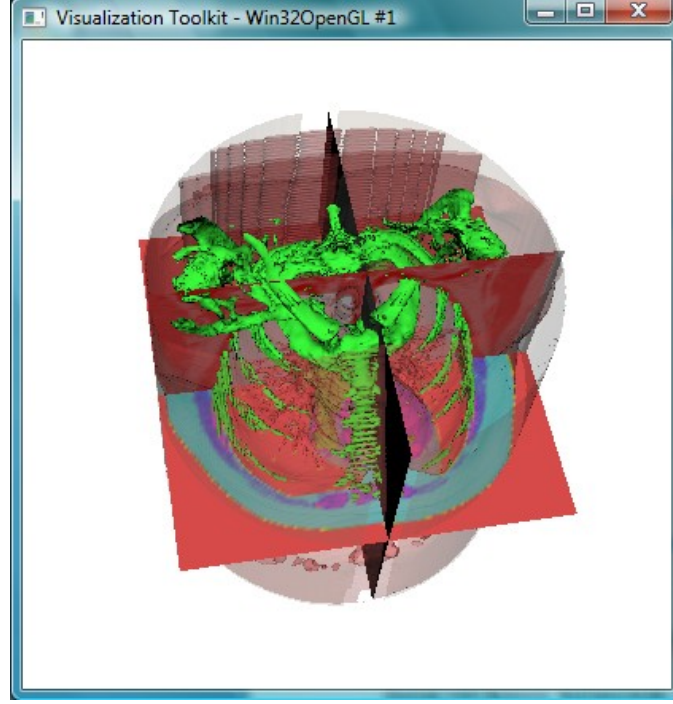
## Aksiyel, Sagital ve Koronel Eksenlerin Görüntülenmesi

thorax veri kümesi kullanılarak, hacimsel görüntü üzerinde üç boyutu temsil eden aksiyel, sagital ve koronel eksenler görüntülenmiştir. Bunun için her eksene ait lookup tabloları yaratılmıştır. Görüntünün elde edilmesi için geçen süre, 21 sn'dir.

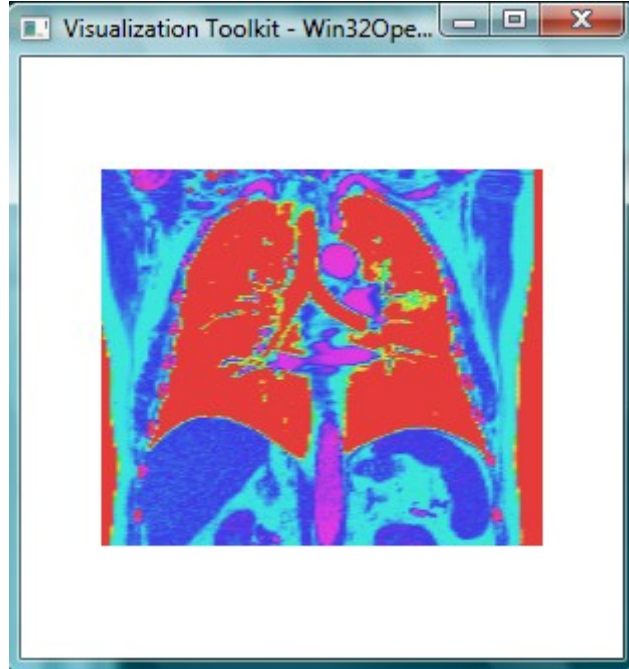
Bu eksenlerin üçü birlikte, Şekil 6.4.1'de görüldüğü gibi hacimsel görüntüyü kesecek şekilde görüntülenebileceği gibi, ayrı ayrı da görüntülenebilir. Şekil 6.4.2'de aksiyel eksen ve üzerinde oluşan cismin izdüşüm görüntüsü görülmektedir. Bu görüntünün elde edilmedi için, lookup tablosu aşağıdaki gibi oluşturulmuş ve aksiyel eksene ait aktöre atanmıştır.

Tablo 6.4: lookup tablosu

```
SetTableRange(0, 2000)  
SetSaturationRange(0.6, 0.6)  
SetHueRange(0, 1)  
SetValueRange(0.8, 0.8)
```



Şekil 6.4.1: Aksiyel, sagittal, koronel eksenlerin görüntülenmesi



Şekil 6.4.2: Aksiyel eksen

# SONUÇ VE ÖNERİLER

Proje, 3 boyutlu görselleştirme algoritmalarının öğrenilmesi ve medikal görüntüleme alanında uygulama geliştirme amacıyla seçilmiştir. Bu amaç, temel oranda gerçekleştirilmiştir. Çok daha karmaşık ve ileri düzeyde olan birçok görselleştirme ve görüntü işleme algoritması bulunmaktadır. Fakat kısıtlı zaman, eldeki verilerin yetersizliği nedeniyle bu algoritmaların tümü üzerinde çalışılamamıştır. Temel algoritmalar çalışılmış, 3 boyutlu görselleştirme alanında temel bilgiler edinilmiş, uygulamalar gerçekleştirilmiştir.

Projede eldeki veri kümelerinin kısıtlı olması nedeniyle, çok çeşitli veri kümeleri üzerinde deneyler yapılamamıştır. Projenin geliştirilmesi için, çeşitli veri kümeleri üzerinde uygulamalar gerçekleştirilmelidir. Projede kullanılan veri tipi yalnızca DICOM verilerdir. Bu veriler, CT ve MRI'lerden elde edilmiştir. Aynı algoritmalar, farklı tiplerde verileri de görüntüleyebilecek şekilde modifiye edilerek proje geliştirilebilir ve kullanım alanları genişletilebilir.

Projede yalnızca temel görselleştirme teknikleri kullanılmıştır. Temel yüzey ve hacim görüntüleme algoritmaları bu projeye uygulamaya konmuştur. Fakat, proje, geliştirmeye çok açık bir uygulamadır. Gelişkin görüntü işleme teknikleri ile elde edilen görüntü, gürültüden arındırılabilir, düzgünleştirilebilir, segmanlara ayrılabilir; segmanlar içinde farklı algoritmalarla görüntüler elde edilebilir, ışıklandırma efektleri ile görüntünün 3 boyutlu görünürlüğü iyileştirilebilir.

Deney sonuçlarında da görüldüğü gibi, görüntüleme algoritmalarının çalışma süreleri, veri kümesindeki dilim sayısına ve veri kümesinin boyutuna göre farklılaşmaktadır. Projede kullanılan 3 boyutlu veri görselleştirme algoritmalarının karmaşıklığı  $O(n^3)$ 'tür. Boşluk atlama olarak adlandırılan iyileştirme algoritmalarıyla bu karmaşıklık azaltılabilir. Ancak bu algoritmalar da veriye bağımlı olduğundan; boşluklu yapıya sahip olmayan verilerde etkili bir iyileştirme sağlanamaz.

Bu proje ve benzeri medikal görselleştirme projeleri, önerilen teknikler kullanılarak yeterince geliştirilir, iyileştirmeler yapılırsa kullanım alanları yaygınlaşacak ve bu tür projeler, tıbbi gelişmelerin ivme kazanmasında rol oynayacaktır.

# KAYNAKLAR

- [1] W. Schroeder, K. Martin, B. Lorensen, *The Visualization Toolkit An Object-Oriented Approach To 3D Graphics, 4th Edition*, Kitware, Inc. Publishers.
- [2] Visualization Toolkit, 2009, <http://www.vtk.org/>
- [3] Cmake 2.6, 2009, <http://www.cmake.org/>
- [4] vtkDataSet Class Reference, 2009,  
<http://www.vtk.org/doc/nightly/html/classvtkDataSet.html>
- [5] vtkDICOMImageReader Class Reference, 2009,  
<http://www.vtk.org/doc/nightly/html/classvtkDICOMImageReader.html>
- [6] vtkRenderer Class Reference, 2009,  
<http://www.vtk.org/doc/release/4.2/html/classvtkRenderer.html>
- [7] vtkRenderWindow Class Reference, 2009,  
<http://www.vtk.org/doc/nightly/html/classvtkRenderWindow.html>
- [8] vtkMarchingCubes Class Reference, 2009,  
<http://www.vtk.org/doc/nightly/html/classvtkMarchingCubes.html>
- [9] Microsoft Visual Studio 2008
- [10] Dataset: vix, 2009, <http://pubimage.hcuge.ch:8080/>
- [11] W. Schroeder, K. Martin, W. Lorensen, *The Design and Implementation Of An Object-Oriented Toolkit For 3D Graphics And Visualization*, GE Corporate Research & Development.
- [12] M. Levoy, *Volume Rendering: Display of Surfaces from Volume Data*, University of North Carolina, May 1988.
- [13] File Formats for VTK, *The VTK User's Guide*, 2009, [www.kitware.com](http://www.kitware.com)
- [14] W. Lorensen, H. Cline, *Marching Cubes: A High Resolution 3D Surface Construction Algorithm*.
- [15] P. Lacroute, *Fast Volume Rendering Using a Shear-Wrap Factorization of the Viewing Transformation*, September 1995.
- [16] M. Baitsch, D. Hartmann, *A Framework for the Interactive Visualization of Engineering Models*, 17th International Conference on the Application of Computer Science and Mathematics in Architecture and Civil Engineering, Weimar, Germany, 12–14 July 2006.
- [17] J. Choi, B. Shin, Y. Shin, K. Cleary, *Efficient Volumetric Ray Casting for Isosurface Rendering*, USA.
- [18] J. Mulder, F. Groen, J. Wijk, *Pixel Masks for Screen-Door Transparency*, Software Engineering (SEN), September, 1998.