

Jeolojik Ölçümlerin OpenGL ile Modellenmesi

Aytaç Tokatlı

**YÜKSEK LİSANS TEZİ**

Elektrik Elektronik Mühendisliği Anabilim Dalı

Mart 2010

Modeling of Geological Measurements with OpenGL

Aytaç Tokatlı

**MASTER OF SCIENCE THESIS**

Department of Electrics and Electronics Engineering

March 2010

Jeolojik Ölçümlerin OpenGL ile Modellenmesi

Aytaç Tokatlı

Eskişehir Osmangazi Üniversitesi  
Fen Bilimleri Enstitüsü  
Lisansüstü Yönetmeliği Uyarınca  
Elektrik Elektronik Mühendisliği Anabilim Dalı  
Telekomünikasyon ve Sinyal İşleme Bilim Dalında  
YÜKSEK LİSANS TEZİ  
Olarak Hazırlanmıştır

Danışman: Yrd. Doç. Dr. Selçuk Canbek

Mart 2010

## ONAY

Elektrik Elektronik Mühendisliđi Anabilim Dalı Yüksek Lisans öğrencisi Aytaç Tokatlı'nın YÜKSEK LİSANS tezi olarak hazırladığı "Jeolojik Ölçümlerin OpenGL ile Modellenmesi" başlıklı bu çalışma, jürimizce lisansüstü yönetmeliđin ilgili maddeleri uyarınca deđerlendirilerek kabul edilmiştir.

**Danışman** : Yrd. Doç. Dr. Selçuk Canbek

**İkinci Danışman** : -

### **Yüksek Lisans Tez Savunma Jürisi:**

**Üye** : Yrd. Doç. Dr. Selçuk Canbek

**Üye** : Prof. Dr. Süleyman Aydın

**Üye** : Doç. Dr. Mustafa Yamaç

**Üye** : Yrd. Doç. Dr. Erol Seke

**Üye** : Yrd. Doç. Dr. Nihat Adar

Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ..... tarih ve ..... sayılı kararıyla onaylanmıştır.

Prof. Dr. Nimetullah BURNAK

Enstitü Müdürü

## ÖZET

Bu tezde, OpenGL ve C Sharp programlama dili kullanılarak üç boyutlu mağara modellemesi gerçekleştirilmiştir. Mağara ölçümü sırasında her istasyonda çekilen fotoğraflar çeşitli işlemlerden geçirildikten sonra programa verilmekte; bu veriler program içerisinde birleştirilerek mağaranın katı modeli oluşturulmaktadır.

Kullanıcı program yardımı ile oluşturulan katı modelin içerisinde veya çevresinde gezinebilmekte, modelin yapısı hakkında fikir sahibi olurken üzerinde uzunluk ve hacim ölçümleri yapabilmektedir.

OpenCave olarak adlandırılan program diğer haritalama programlarından farklı olarak kesit detaylarını fotoğraflardan aldığı için çok daha gerçekçi modeller oluşturabilmekte, kullanıcıya mağara içerisinde geziniyormuş duygusunu verebilmektedir.

Anahtar Kelimeler: Üç boyutlu mağara modelleme, OpenGL

## SUMMARY

In this thesis, 3D cave modeling using OpenGL and C Sharp, which is a part of Microsoft .Net Framework, is implemented. After the prior taken photographs of a cave measurement are migrated with different types of operations, obtained data is given to the software. In the program, taken data is combined for creating solid cave model.

User can look around or look in of a created solid model by using this program. Additionally, while holding a view of the model structure, user can measure lengths and volume.

The program, which is called as OpenCave, can create more realistic models instead of other mapping softwares, because section details are taken from photographs. By this way, user can feel like walking around the inner space of a cave.

Keywords: 3D cave modeling, OpenGL

## TEŞEKKÜR

Bu tez çalışması sırasında anlayışından ve yardımlarından ötürü Osmangazi Üniversitesi Bilgisayar Mühendisliği Bölümü öğretim üyelerinden Yrd. Doç. Dr. Selçuk CANBEK'e teşekkür ederim.

Ayrıca, desteklerini benden hiç esirgemeyen aileme de çok teşekkür ederim.

**İÇİNDEKİLER**

	<b><u>Sayfa</u></b>
<b>ÖZET .....</b>	<b>v</b>
<b>SUMMARY.....</b>	<b>vi</b>
<b>TEŞEKKÜR.....</b>	<b>vii</b>
<b>İÇİNDEKİLER.....</b>	<b>viii</b>
<b>ŞEKİLLER DİZİNİ.....</b>	<b>xiii</b>
<b>ÇİZELGELER DİZİNİ.....</b>	<b>xvi</b>
<b>1. GİRİŞ .....</b>	<b>1</b>
<b>2. MAĞARA HARİTALAMA .....</b>	<b>12</b>
2.1 Ölçüm Malzemeleri.....	12
2.1.1 Sabit kadranlı pusula .....	13
2.1.2 Prizmatik pusula.....	14
2.1.3 Eğimölçer.....	14
2.1.4 Şerit metre ve teleskopik metre.....	15
2.1.5 Poligon kazıkları ve prizma .....	15
2.1.6 Not defteri ve ölçüm çizelgeleri.....	16
2.2 Ölçümler.....	17



**İÇİNDEKİLER (devam)**

	<b><u>Sayfa</u></b>
2.2.1 Hazırlık aşaması.....	17
2.2.2 Gözlem aşaması .....	17
2.2.3 Ölçme/taslak ve kayıt altına almak .....	18
2.3 Mağara Ölçümleri ve Standardı.....	18
2.3.1 Orta hat hassaslık derecelendirmesi .....	18
2.3.2 Ayrıntı belirleme sınıflandırması .....	19
2.3.3 Mağara ölçümünde basit yöntemler .....	20
2.3.3.1 Orta hattın çıkartılması yöntemi.....	20
2.3.3.2 Duvardan duvara yöntemi.....	21
2.3.3.3 Karışık yöntem .....	21
2.3.3.4 Işınsal yöntem .....	22
2.3.4 Mağara ölçümünde ileri yöntemler .....	23
2.4 Mağara Haritası Çizim Yöntemleri.....	23
2.4.1 Elle çizim.....	24
2.4.2 Pusula ile çizim .....	24
2.4.3 Açılışölçerle çizim .....	25
2.4.4 Kesit çizimleri.....	25

**İÇİNDEKİLER (devam)**

	<b><u>Sayfa</u></b>
2.4.5 Boyuna açılmış kesitler .....	26
2.4.6 En kesitler .....	27
2.5 Özetle Geleneksel Harita Çıkartma Yöntemi .....	27
2.6 Fotoğrafla Haritalama Yöntemi .....	31
2.6.1 Paralel izdüşüm dönüşümü .....	34
2.6.2 Rotasyon .....	35
2.6.3 Resim iyileştirme yöntemi .....	35
2.6.4 Gölge filtresi .....	35
<b>3. ÜÇ BOYUTLU MAĞARA MODELİ OLUŞTURMA .....</b>	<b>36</b>
3.1 Grafik Kütüphanelerine Genel Bakış .....	36
3.1.1 Direct3D .....	36
3.1.2 OpenGL .....	37
3.1.3 Performans karşılaştırması .....	38
3.1.4 Tercih edilen grafik kütüphanesi .....	38
3.2 Kesit Fotoğraflarının Birleştirilmesi .....	39
3.3 Nesnelerin Oluşturulması .....	43
3.3.1 Doku kaplama .....	43

## İÇİNDEKİLER (devam)

	<b><u>Sayfa</u></b>
3.3.2 Kamera.....	45
3.4 Verilerin Birleştirilmesi.....	46
3.5 Verilerin Yorumlanması.....	47
3.5.1 Döndürmek için gerekli olan açının hesaplanması .....	48
3.5.2 Ara istasyonlar .....	48
3.5.3 İstasyonların döndürülmesi.....	49
3.5.4 “Cubic spline” eğrileri.....	50
3.6 Kullanıcı Ayarlarındaki Değişikliklerin Tespit Edilmesi.....	54
3.7 Hacim ve Uzunluğun Hesaplanması .....	55
3.8 Çizim .....	57
<b>4. UYGULAMANIN KULLANIMI .....</b>	<b>59</b>
4.1 Serbest Dolaşım Modu.....	59
4.2 Yol Modu .....	60
4.3 Prizma Modu .....	62
4.4 İstasyonlar Arası Dolgu Kaplama.....	63
4.5 Ekran Görüntüsünün Saklanması.....	65
4.6 Yeni Bir Modelin Yüklenmesi.....	65

**İÇİNDEKİLER (devam)**

	<b><u>Sayfa</u></b>
4.7 Konumla İlgili Temel Bilgiler .....	66
4.8 Klavye Kısa Yollarının Deęiřtirilmesi .....	68
<b>5. SONUÇ.....</b>	<b>69</b>
<b>6. KAYNAKLAR DİZİNİ .....</b>	<b>71</b>

## ŞEKİLLER DİZİNİ

<b><u>Sekil</u></b>	<b><u>Sayfa</u></b>
1.1 Winkarst programı prizma içerisinde gösterim.....	4
1.2 Winkarst programı genel gösterim .....	4
1.3 Survex programı genel gösterim .....	5
1.4 Survex programı detaylı gösterim .....	6
1.5 Compass programı detaylı gösterim.....	7
1.6 Compass programı üç boyutlu gösterim .....	8
1.7 Compass programı jeolojik özelliklerle gösterim .....	8
1.8 CaveRender programı ekran görüntüsü .....	9
2.1 Haritacılıkta kullanılan optik ve optik olmayan aletler .....	13
2.2 Sabit kadranlı pusula .....	13
2.3 Prizmatik teodolit pusula .....	14
2.4 Eğimölçer.....	15
2.5 Orta hat yönteminde poligon noktalarının durumu .....	21
2.6 Karışık yöntem .....	22
2.7 Işınsal ölçüm yöntemi.....	22
2.8 Boyuna açılmış kesitte boy uzamaları .....	26
2.9 Bir mağara galerisinde en kesit kavramı.....	27
2.10 İstasyonların isimlendirilmesi .....	28
2.11 İstasyonlar arası izdüşüm ve ölçülen uzaklık.....	29
2.12 İstasyonlar arası yükselti açısı ve yükselti.....	29
2.13 Diğer istasyonların isimlendirilmesi.....	30

## ŞEKİLLER DİZİNİ (devam)

<b><u>Sekil</u></b>	<b><u>Sayfa</u></b>
2.14 Geleneksel harita çıkartma yöntemi .....	31
2.15 Fotoğrafla haritalama yöntemi ile harita çıkartma .....	32
2.16 Işık kaynağı .....	33
2.17 Fotoğrafla haritalama yönteminin uygulanışı .....	33
2.18 Fotoğrafla haritalama yöntemi ile elde edilen görüntü.....	34
3.1 Bir kesite ait noktalar ve birleştiren “cubic spline” .....	39
3.2 İstasyonların listelendiği Excel dosyasına ait içerik.....	40
3.3 Uygulama içerisinde kullanılan modelin açıklaması.....	41
3.4 Cubic spline eğrisi ve istasyonlar ile ilişkisi.....	51
3.5 Uzunluk hesaplamasında oluşan farkın gösterimi.....	56
3.6 Hacmin hesaplanması .....	57
4.1 Serbest dolaşım modu.....	60
4.2 Yol modu .....	61
4.3 Yol modu ve ışığın etkisi .....	61
4.4 İstasyonları kapsayan prizmanın gösterilmesi .....	62
4.5 İstasyonları kapsayan prizmanın çevresinde dolaşım.....	63
4.6 İstasyonları kapsayan prizmanın çevresinde detay bilgileri .....	63
4.7 Dolgunun gösterimi aktifken istasyonlar .....	64
4.8 Dolgunun gösterimi pasifken oluşan eğri .....	64
4.9 Ekran görüntüsü ile ilgili menüler .....	65
4.10 Koordinatların gösterimi.....	66
4.11 Harita üzerinde kullanıcının konumu .....	67

**ŞEKİLLER DİZİNİ (devam)**

<b><u>Şekil</u></b>	<b><u>Sayfa</u></b>
4.12 Eksenler .....	67
4.13 Klavye kısa yollarının deęiřtirilmesi .....	68

**ÇİZELGELER DİZİNİ**

<b><u>Çizelge</u></b>	<b><u>Sayfa</u></b>
2.1 Öncü kayıt çizelgesi.....	16
2.2 Artçı kayıt çizelgesi.....	16
2.3 Orta hat hassaslık derecelendirmesi .....	19
2.4 Ayrıntı belirleme sınıflandırması .....	19



## BÖLÜM 1

### GİRİŞ

Yeraltında bulunan, en az bir insanın girebileceği kadar genişliğe sahip olan doğal boşluklara “mağara” denir. Bunlar bir kaç metreden, kilometrelerce uzunluk ve yüzlerce metre derinlik veya yüksekliğe ulaşabilirler. Mağaralar oluşum şekline göre: doğal ve yapay mağaralar olarak iki gruba ayrılır. İnsanların kazdığı veya hayvanların oyduğu boşluklar yapay mağaraları oluştururlar. Buna karşılık ana kaya oluşurken veya oluştuktan sonraki fiziksel ve kimyasal olaylarla oluşan mağaralara da “doğal mağara” adı verilir.

İnsanların barınma, korunma, eşya saklama gibi ihtiyaçlarının dışında sportif ya da bilimsel amaçla mağaraları araştırması ve mağaralara girmesine mağaracılık denir. İnsanlık tarihindeki birçok olayda olduğu gibi mağaracılığın başlaması ve gelişmesi de insanın keşfetme isteği ile gerçekleşmiştir. Merak eden insan, doğanın gizlendiği yerde ne olduğunu bulmak için mağaraya girip karanlığı aydınlatılabildiği yere kadar ilerlemiştir. Merak duygusu ve keşif isteğinin peşi sıra amatör bir şekilde başlayan mağaracılık aradan geçen yıllar içinde hem sportif hem de bilimsel anlamda profesyonel olarak yapılr hale gelmiştir. “Speleoloji” olarak adlandırılan mağara bilimi, mağara ve mağara oluşumlarının nasıl ve ne zaman oluştuklarını, mağara ekosistemini ve mağaraları da kapsayan su sistemlerini inceler (Hacettepe Üniversitesi Mağara Araştırma Kulübü, 2004).

Öncelikle, jeoloji ve hidrojeoloji bilim dallarının konu alanı olan mağaralar, araştırmacılara yaşadığımız dünyayla ilgili çok önemli bilgiler sağlar. İlk insanların mağaralarda yaşamış olması ve insanların uzun yıllar boyunca mağaraları çeşitli amaçlarla kullanması, mağaraları arkeolojik açıdan önemli mekânlar haline getirmiştir. Mağaralar kendi ekosistemleri içinde, birçoğuna mağara dışında rastlanmayan geniş bir canlı çeşitliğine de ev sahipliği yapmaktadır. Tüm bu nedenlerden ötürü birçok bilim adamı bilgiye ulaşmak için mağaralara girmektedir. Mağaraya girecek kişinin, mağara

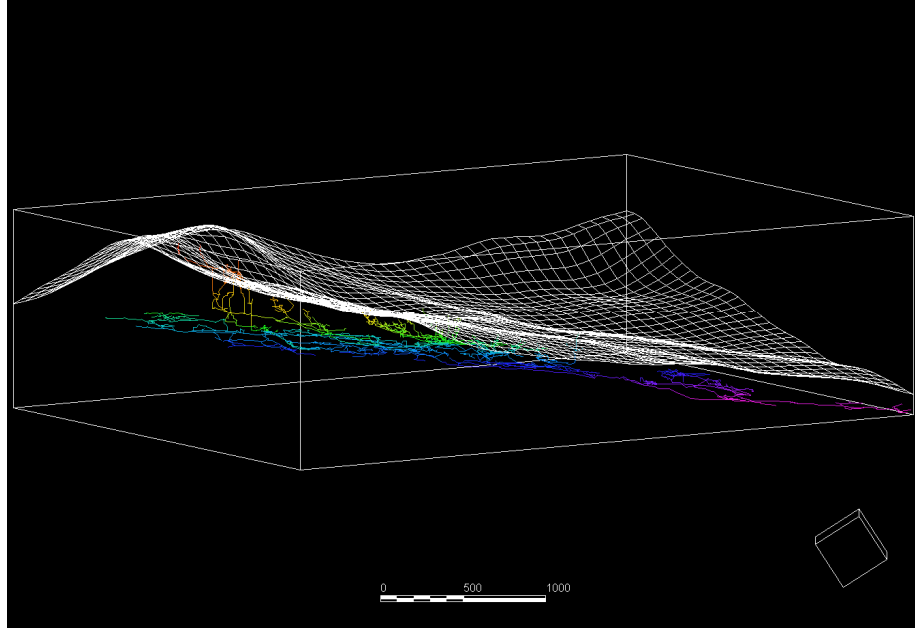
içindeki engelleri aşıp daha ileriye gidebilmesi için teknik, bilgi ve beceriye sahip olması gerekir (Hacettepe Üniversitesi Mağara Araştırma Kulübü, 2004).

Jeolojik haritalar ve bu haritaların türetilmiş ürünleri ekonomik ve sosyal öneme sahiptir. Bu haritalar daha modern ve daha da çok internet üzerinden erişebilir hale geldiğinde kullanışlılığı artmaktadır. Mineral ve su kaynaklarının tespit edilmesi ile birlikte çıkartılabilmesine yardımcı olan haritalar, yeraltı su kalitesini koruma ve değer biçilmesi açısından atık boşaltım tesislerinin konumlarının belirlenmesinde de önemli bir kaynaktır. Ayrıca hassas ekosistemin korunması, tanımlanması ile doğal felaketlere karşı hazırlıklı olunması açısından konum belirlemede temel unsurdur. Jeolojik haritalar, sosyal ihtiyaçlar göz önüne alındığında yol, otoban, tren yolu, su dağıtım sistemi, binaların konumlandırılması, şehir planlama gibi ulusların iç yapılarını düzenleyebilmeleri için temel başvuru kaynağıdır (The Geological Society of America, 2008).

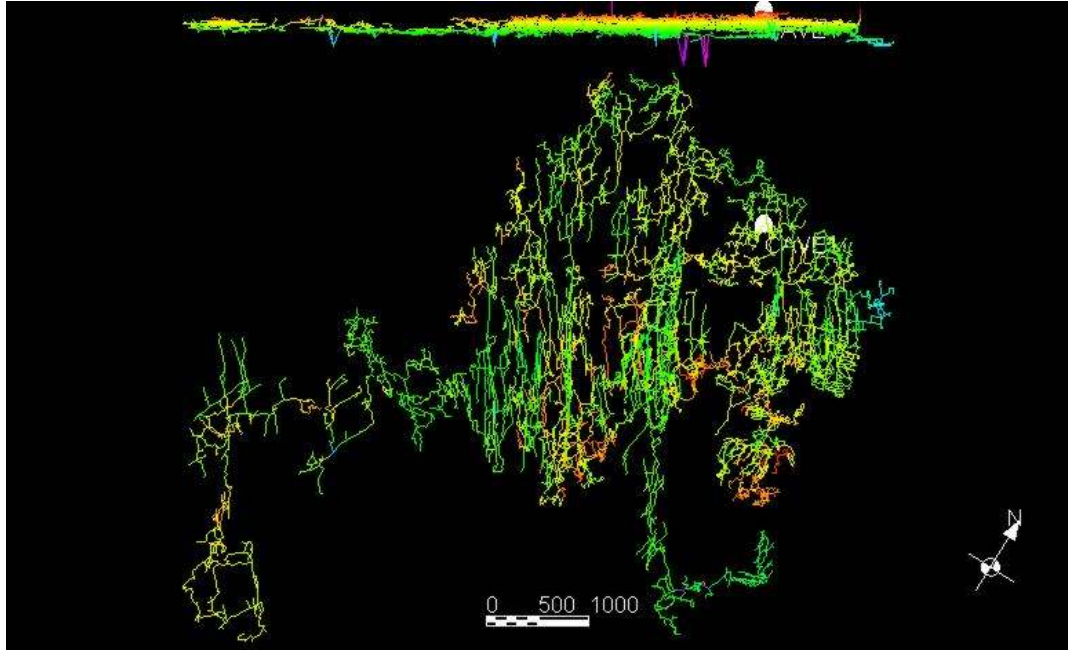
Arkeoloji ve biyoloji gibi mağaralarla ilgilenen diğer bilim dalları tarafından da sıklıkla kullanılan jeolojik haritaların daha anlaşılır olması ve üzerinde ölçümler yapılabilmesi için mağaraların da üç boyutlu modellenmesi gerekmektedir. Günümüzde şehirlerin ve özellikle mimari yapıların üç boyutlu modellenmesi ve içerisinde sanal olarak dolaşma olanağı, planlama, projelendirme, sunum ve görselleştirme işlemleri üç boyutlu olarak bilgisayarlarda yapılmaktadır.

Günümüzde birçok mağara modelleme programı bulunmaktadır. Wookey'in (2001) yaptığı incelemede Survex, Tunnel, Cml, Vectors, Caveplot, Toporobot, Cavesurveystandard, Caps, Cavemap, Cmap, Compass, Karst256, Smaps, Smaps Lite, Surv93, Survey, Caveview, CaveRender, Onstation, Pitter/Plotter, Speleo Graphics, Wincompass, Winkarst ve Wincaps gibi mağara inceleme ve modelleme yazılımları listelenmiştir. Bunların arasında aktif olarak kullanılan Winkarst, Survex, Compass ve CaveRender yazılımları öne çıkmaktadır. Bu yazılımların özellikleri aşağıda detaylı olarak belirtilmektedir.

Winkarst, Resurgent Software tarafından hala geliştirilmesine devam edilen yazılımlardan birisidir. Yazılıma <http://www.resurgentsoftware.com/WinKarst.html> adresinden ulaşılabilir. Winkarst programı entegre olarak çalışan mağara araştırma editörüne sahiptir. Araştırmacı mağaranın özelliklerini herhangi bir yere kaydetmeden değiştirebilmektedir. Ana kayıtları bozmadan hata düzeltme özelliklerine sahiptir. PNG formatındaki 100 resme kadar işlem yapabilmektedir. İki istasyon arasında yumuşak geçişler yapılabilir. Sabit bir doğru temel alınarak çizim yapılması istenildiğinde en az iki istasyonun sabitlenmesi istenilmektedir. Ardından eğer üçüncü bir istasyon da sabitlenirse, oluşturulan çizimi doğruyu ve istasyonları temel olarak değiştirmektedir. Resimler, sisteme arka planı saydam olacak şekilde yüklenmektedir. Birden fazla dosya formatını desteklemektedir. SMAPS, SEF, Compass, mak, dat ve plt formatlarını okuma özelliğine sahiptir. Ayrıca, Microsoft Access gibi programlarda hazırlanmış GPS formatlarını da mağara modeli üzerine uygulayabilmektedir. Topografik veya bölgesel haritaları tarayıp mağara sisteminin arka planı olarak gösterebilmekte, doku malzemesi olarak GeoTiff formatındaki resimleri kullanabilmektedir. WayPoint+ veya Garmin formatlarında GPS verilerini kullanabilmektedir. İstasyon bilgilerini, dBaseIV formatında diğer veritabanlarından alarak saklayabilmekte, eğer bir mağaranın coğrafik referans noktası biliniyorsa otomatik manyetik eğitimi hesaplaması yapabilmektedir. Mağara haritalarını BMP, PNG, DXF, TXT, Garmin TRK ve WPT formatlarında, TXT ve DBF hesap çizelgesi veya ERSI şekil dosyası olarak dışarı verebilmektedir. BMP ve PNG formatlarında herhangi bir piksel boyutlarında dışarı veri verebilmektedir. Dışarı verilen dosyalar CorelDraw, AutoCAD, MS Word, DesignCAD, WayPoint+, TOPO!GPS, Bryce, MS Excel ve MS Access içerisinde kullanılabilir. Dikey kesit dağılım diyagramı oluşturabilmekte ve bunun 256 katmana kadar histogram yapısını çıkartabilmektedir. Mağara parametrelerini ve istatistikleri gösterebilmektedir. Bu parametre ve istatistikler: mağara uzunluğu, uç değerler, istasyon sayısı, spiral sayısı, spiraller üzerindeki hatalar, spiraller üzerindeki beklenen hatalar, spiral uzunluğu, istasyon bağlantı noktalarının sayısı, girişe en yakın istasyonun uzaklığıdır. GPS araçları ile birlikte kullanabilmek için coğrafik izdüşümler oluşturabilmektedir. Yazılım ile ilgili örnek ekran görüntüleri Şekil 1.1 ve Şekil 1.2 üzerinden incelenebilir.

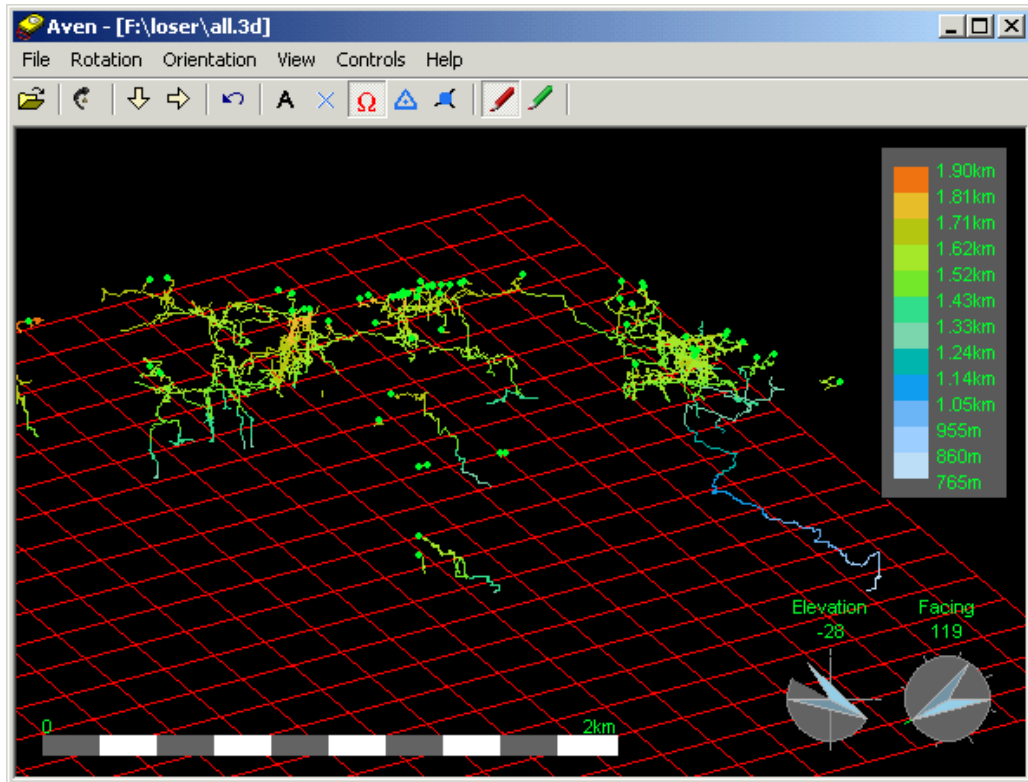


Şekil 1.1 Winkarst programı prizma içerisinde gösterim

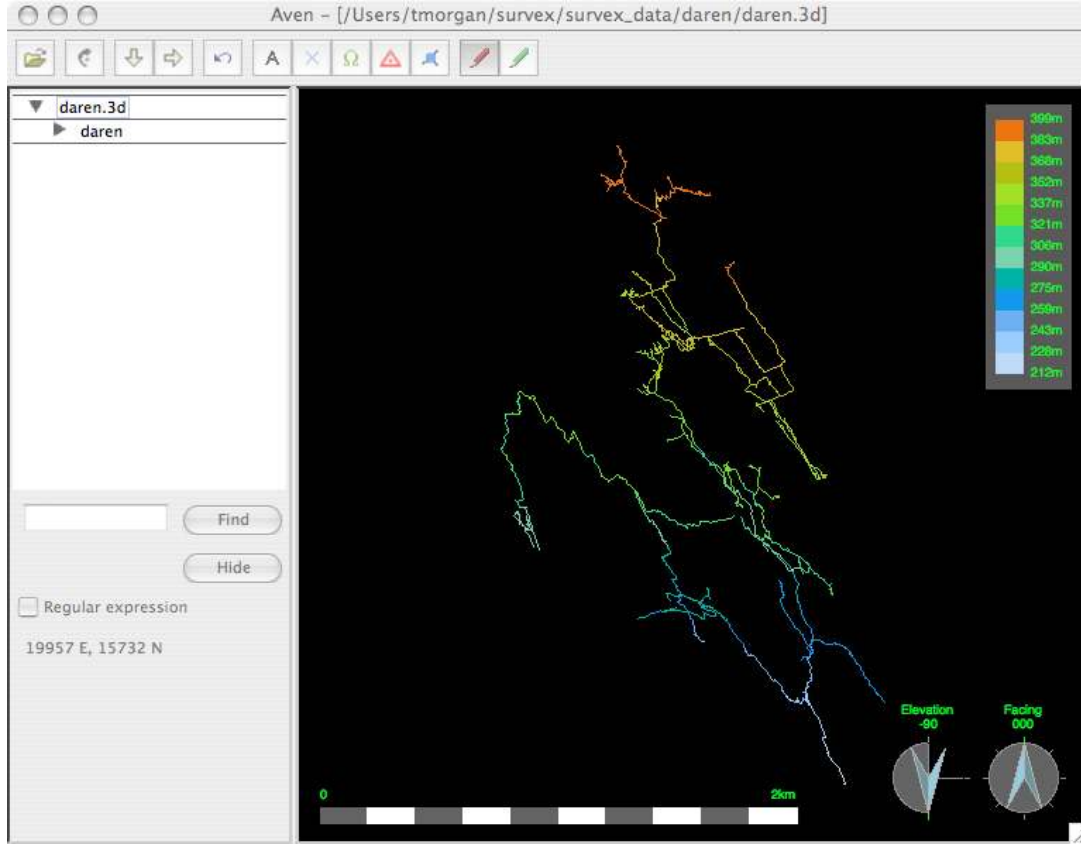


Şekil 1.2 Winkarst programı genel gösterim

Survex açık kaynak kodlu olarak geliştirilen ve <http://survex.com/> adresinden ulaşılabilen Microsoft Windows, Mac OS X, ve Unix gibi bir çok işletim sisteminde çalışabilen bir yazılımdır. Yazılım geliştirilirken araştırmacıyı engelleyecek herhangi bir limitten kaçınıldığı belirtilmektedir. Birçok farklı projeden modelleme veya modellemeleri bir proje altında toplayabileceği ve modellemenin oluşturacağı veri üzerinde herhangi bir sıralama olmasına gerek olmadığı belirtilmektedir. Eksik veriler üzerinden, geri gözleme sistemi ile veya derinlik verileri olan su altı sistemlerini modelleyebilmekte, verileri küçük parçalara ayırarak işlemektedir. Manyetik değişiklikleri kullanarak kalibrasyon ayarı yapabilmekte, kullanıcının tanımlayacağı metrik sistemler ile bilinen metrik sistemleri modellemelere yansıtabilmektedir. Yazılım ile ilgili örnek ekran görüntüleri Şekil 1.3 ve Şekil 1.4 üzerinden incelenebilir.



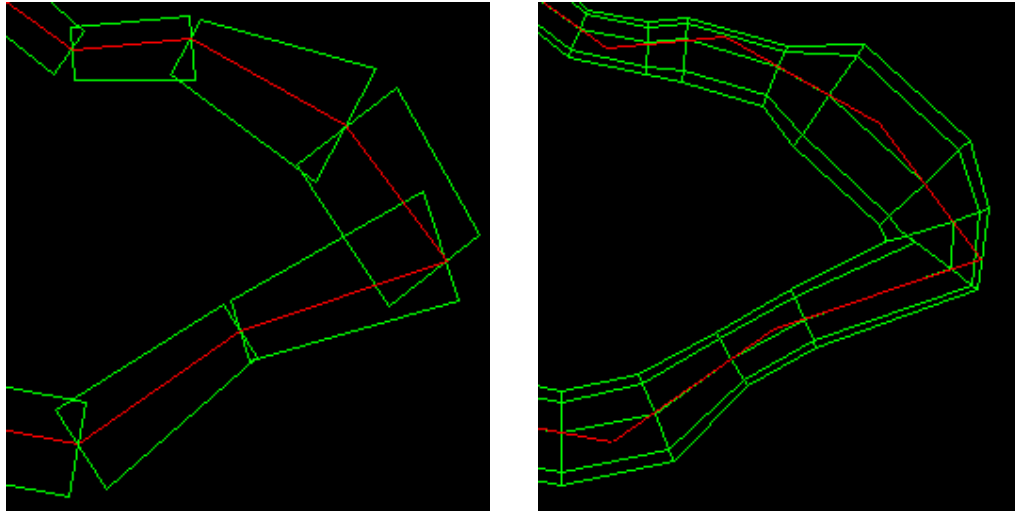
Şekil 1.3 Survex programı genel gösterim



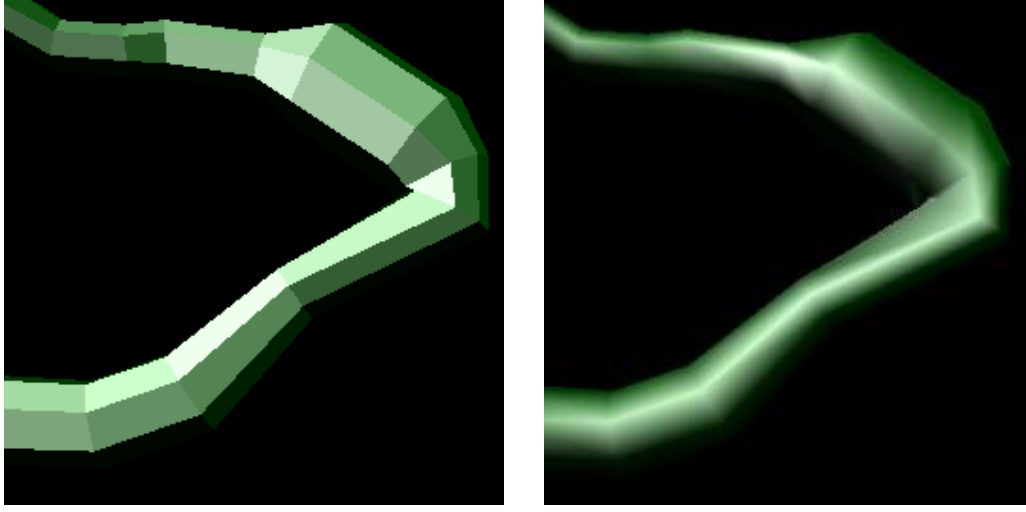
Şekil 1.4 Survox programı detaylı gösterim

Compass 1979 yılında Larry Fish tarafından geliştirilmeye başlanmıştır. Öncelikle Mainframe üzerinde çalışan sürümler zamanla IBM üzerinde çalışacak hale getirilmiş, en son olarak da Microsoft Windows üzerinde çalışan sürümleri hazırlanmıştır. Compass ile ilgili detaylı bilgiye <http://fountainware.com/compass/> adresinden ulaşılabilir. Yazılım 32 Bit DEM dosya okuyucusu içermektedir. Çok büyük modellemelerde araştırmacının gördüğü alandan belirli bir oranda büyük bir parçayı hafızada tutmaktadır. Sistemin zorlanmaması için geri kalan alanı saklamamakta, ihtiyaç olduğunda yüklemektedir. Yani modellemeyi parçalara ayırarak çalışmaktadır. Proje olarak modellemeleri saklamayı ve birleştirmeyi desteklemekte, editörü ile gösterilen model üzerinde istenilen alanın yapısının değiştirilebilmesine olanak sağlamaktadır. Rose Diagramı ve derinlik histogramı oluşturabilmekte, yönetme

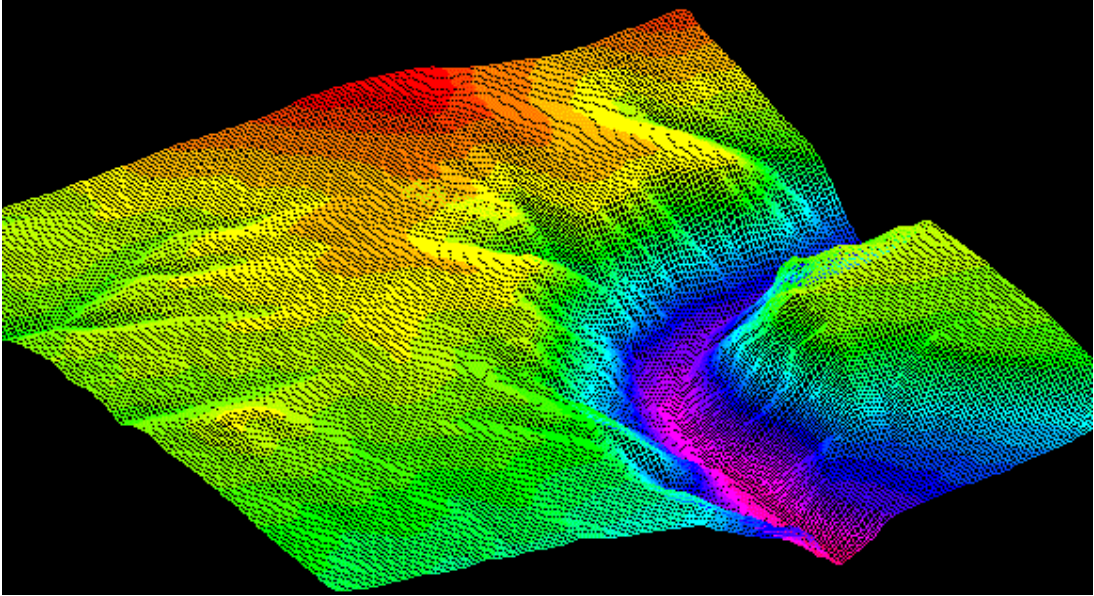
kolu (joystick) kullanımına olanak tanımaktadır. AutoCAD kullanıcılarından 3D çizim modeline sahip olmayanlar için DXF formatında verileri dışarı verebilmekte ve AutoCAD 12, 13 ve 14 sürümleri için yön oklarını ve orantı barını kullandırmaktadır. 32000 metre uzunluğunda bir tünel gösterilirken döşeme kaplama ve gölgelendirme yöntemlerini kullanarak saniyede 20 görüntü ve 2.000.000 poligon gösterebilmektedir. Model içerisinde kullanıcı gezinirken yaptığı tüm adımları film olarak kaydedebilmekte, diğer tüm bileşenleri değişmeden dikey olarak büyütme işlemi yapabilmektedir. DirectX üzerine kurulan yapısı son kullanıcının çok yüksek veya çok düşük çözünürlükte çalışabileceği düşünülerek tasarlanmıştır. DEM formatındaki dosyalar içerisinde birbirlerini takip eden farklı dosyalar varsa bunları birleştirebilmektedir. Topografik haritalar hazırlayabilmekte ve Corel Draw formatında modellemeyi dışarı verebilmektedir. Yazılım ile ilgili örnek ekran görüntüleri Şekil 1.5, 1.6 ve Şekil 1.7 üzerinden incelenebilir.



Şekil 1.5 Compass programı detaylı gösterim



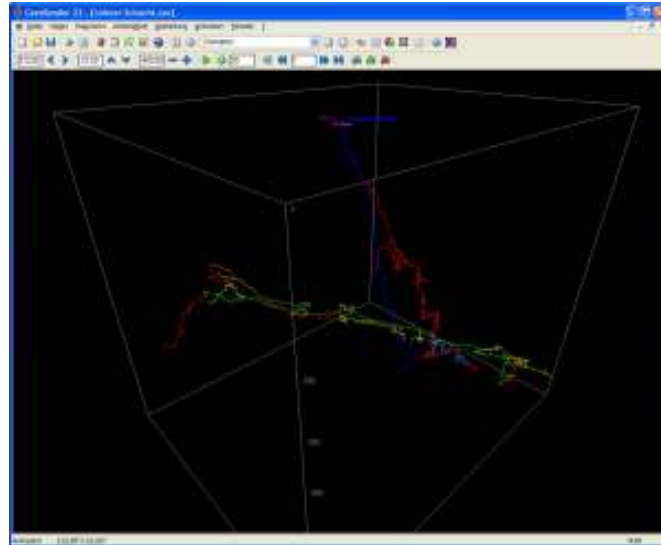
Şekil 1.6 Compass programı üç boyutlu gösterim



Şekil 1.7 Compass programı jeolojik özelliklerle gösterim



CaveRender Jochen Hartig tarafından geliştirilmektedir. <http://caverender.de/caverend/caverend.htm> adresinden ulařılabilen yazılım mađara modellerinin bařka ortamlardan alınmasını ve üzerinde deđişiklik yapılması ile topografya özelliklerinin kullanılmasını desteklemektedir. Excel formatında veriler üzerinden modelleme işlemini gerçekleřtiren yazılım iki istasyon arasındaki mesafenin ölçümü, istasyonların listelenmesi ve istasyon bilgileri arasında arama yapılabilmesi gibi temel özellikleri içermektedir. Ayrıca, özel koordinat sistemi kullanılarak istasyona ait veriler üzerinde deđişiklikler yapılabilir. Hata yakalama özelliđi ile birlikte noktalar arasındaki uzaklıkların ölçümü, toplam uzunluk ve derinlik hesaplaması yapabilen yazılım arařtırmacıya paralel ve perspektif bakıř esnekliđi sağlamaktadır. Yazılım verileri düz metin formatında dosyaya yazıp fare aracılıđı ile kaydırma, döndürme, yakınlařtırma işlemleri yapabilmektedir. Ayrıca Java sınıfları ile internet ortamında sunumların yapılabilmesine olanak sağlamaktadır. Yazılım, VRML adı verilen “Virtual Reality Modeling Language” formatında dosyalama işlemi yapabilip ve temel bilgiler verildikten sonra otomatik veya manuel renklendirme ile oluşturduđu modeli bitmap formatında dıřarıya verebilmektedir. Yazılım ile ilgili örnek ekran görüntüsü Şekil 1.8 üzerinden incelenebilir.



Şekil 1.8 CaveRender programı ekran görüntüsü

Winkarst; tümleşik çalışan mağara araştırma editörüyle, Survex; farklı modelleri tek bir yapı altında toplamasıyla, Compass; gösterilecek modeli parçalara ayırıp işlemci üzerindeki yükü azaltmasıyla ve CaveRender, topografya özelliklerinin kullanımını sağlamasıyla öne çıkmaktadır. Birbirlerinden farklı özellikler içeren bu yazılımlar araştırmacıyı istatistiksel verilerden kurtarıp daha detaylı inceleme yapılabilmesine olanak sağlamaktadır. Ancak, söz konusu yazılımlar incelendiğinde çalışma prensiplerinin geleneksel harita çıkartma yöntemi ile elde edilen verilerin üzerine inşa edildiği görülmektedir. Geleneksel harita çıkartma yönteminde mağara incelemesi en az iki kişiden oluşan bir ekip tarafından yapılmaktadır. Araştırma esnasında bu ekip iki gruba ayrılır. Birinci grup öncü, ikinci grup ise artçı olarak adlandırılır. İncelenecek mağaranın haritasının çıkartılması ekip tarafından belirlenen, istasyon adı verilen konumlarda yapılır (Eskişehir Mağara Araştırma Derneği Eğitim Notları, 2005). Geleneksel yöntemlerde, her istasyonun kesit yüksekliği ve genişliği tahmin edilir veya elle ölçülür. Mağaranın yaklaşık üç boyutlu haritası bu kesitlerin birleştirilmesi ile oluşturulur. Daha doğru bir harita elde edebilmek için istasyon sayısı arttırılmalıdır. Ancak, bu işlem istasyon konumunda duran araştırmacı tarafından yapıldığı için, detayların birçoğu el ile çizilmiş şematik çizimlerde kaybolur (Canbek ve Adar, 2005).

İstasyon kesitlerinin araştırmacı tarafından elle çizilmesi yerine fotoğrafları çekilebilir. Fotoğrafla haritalama (photo-survey) denilen bu yöntem ile çekilen her bir kesit fotoğrafı resim işleme teknikleri ile düzenlenir. Bu yöntemde 180 derecelik yarı küreyi aydınlatan ışık kaynağı kullanılarak fotoğraf çekilir. Işık kaynağı, kesit detayı elde edilecek istasyona yerleştirilir. Fotoğraf makinesi ise istasyonun arkasında uygun bir yere yerleştirilir ve ışık kaynağından ne kadar uzaklıkta olduğu kaydedilir. Çekilen her fotoğraf bazı görüntü işleme tekniklerinden geçirilmektedir. Öncelikle fotoğrafa paralel izdüşüm dönüşümü uygulanır. Böylece ortam geometrisinden kaynaklanan perspektif bozulmaları düzeltilir ve görüntü üzerinde her noktada ölçüm yapılabilir. Ardından, kesit düzleminin tam karşısından fotoğraf çekilemediği durumlarda, kesit görüntüsünün doğru elde edilmesini sağlayacak döndürme işlemi yapılır. Görüntü üzerindeki gereksiz detaylar ile gürültü temizlenir ve gölge filtreleme tekniği ile her kesitin sınır eğrisi elde edilir. Kesit eğrileri belirli aralıklarla örneklenerek kesit sınır noktaları elde edilir.

Bu çalışmada fotoğrafla haritalama yöntemi ile elde edilen kesit eğrileri üç boyutlu düzlemde konumlandırılmış ve mağara modeli oluşturulmuştur. Konumlandırılan her istasyon ile bir sonraki istasyon arasındaki geçişlerin sertliğini azaltabilmek için interpolasyon yöntemi ile ara istasyonlar elde edilmiştir. Kullanıcı isteğine göre aktif hale getirilen bu özellik, kesitlerin birleştirilmesi sırasında hesaplamalara eklenmiştir. Böylece, diğer programlara kıyasla çok daha gerçekçi görünen mağara modelleri elde edilmiştir.

Bu çalışmada geliştirilen ve OpenCave olarak adlandırılan yazılım, temel olarak Wookey'in (2001) yaptığı araştırmada değinilen yazılımların genel özelliklerini içermekle birlikte mağaranın daha gerçekçi modellenmesini sağlamakta, ayrıca kullanıcıya mağaraya gitmeden önce, detaylı olarak iç yapısını inceleme ve dolaşma fırsatı sunmaktadır. Mağaranın modeli üzerinden elde edilen uzunluk, hacim, hata oranı ve bir önceki konuma göre eğimin hesaplanması ile daha detaylı incelemeler yapılabilmektedir.

Bu çalışmanın ikinci bölümünde geleneksel harita çıkartma yöntemi, fotoğrafla haritalama yöntemi, iki yöntem arasındaki farklar ve kullanılan model incelenecektir. Üçüncü bölümde, modelin nasıl uygulandığı anlatılacaktır. Dördüncü bölümde, ikinci bölümde anlatılan model doğrultusunda oluşturulan yazılımın nasıl kullanılabileceğine değinilecektir. Beşinci bölümde ise sonuçlar vurgulanacaktır.

## BÖLÜM 2

### MAĞARA HARİTALAMA

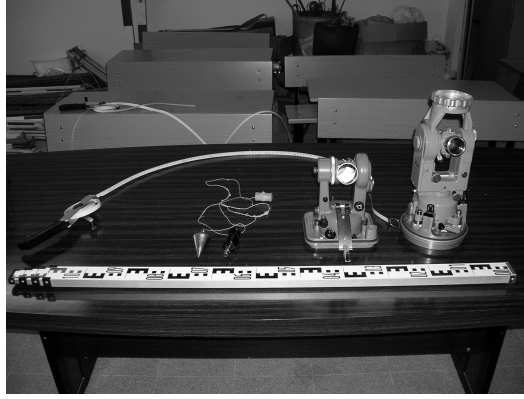
Günümüzde, özellikle bilgisayarların gelişimi ile birlikte ortaya mağara haritası çizebilen özel yazılım programları çıkmıştır. Bunların en önemli özelliği ise üç boyutlu harita çizebilmeleridir. Üç boyutlu haritaların en önemli özelliği, mağaranın gelişimini her açıdan görebilmeye olanak sağlamasıdır. Bilgisayar ortamında yeryüzü şekilleri ile birlikte kullanıldığında gelişim çok net görülebilmektedir (Türkiye Mağaracılar Birliği, 2007).

Mağaralar, çok farklı yapıda jeolojik özellikler içerdikleri için zor erişilebilen bir yapıdadır. Bir mağaranın haritasının çıkartılması ancak özel eğitimlerden geçmiş deneyimli mağaracılar tarafından yapılabilmektedir. Bir mağaranın haritasının çıkartılabilmesi için mağarayı araştıran ekip iki istasyonun birbirine olan bağlı yapısından ötürü pusula, eğimölçer, uzaklık ölçer gibi özel teçhizatlarla donatılmış olmalıdır. Bu tür cihazların taşınması esnasında oluşabilecek sıkıntılar nedeni ile bir araştırmacının araştırma yapabilmesini kolaylaştırmak için modelleme ortaya çıkmıştır.

#### 2.1 Ölçüm Malzemeleri

Tüm harita çeşitleri haritalama amacına uygun olarak tasarlanmış optik ve optik olmayan ölçme aletleri ile gerçekleştirilir. Öncelikle, tüm haritalama tekniklerinde kullanılan aletler, ardından mağara haritalamasında kullanılan araç ve gereçlere değinmek aşamaları anlamayı kolaylaştıracaktır (Türkiye Mağaracılar Birliği, 2007).

Kullanılan optik aletler nivo, teodolit, prizma pusula ve ölçüm aletlerinden oluşurken (Şekil 2.1) optik olmayan aletler ise şerit metre, teleskopik metre, miralar, çeküller, ipler, çizelgeler, not defterleri ve poligon kazıklarından oluşmaktadır.



Şekil 2.1 Haritacılıkta kullanılan optik ve optik olmayan aletler

### 2.1.1 Sabit kadranlı pusula

Pusula yerin manyetik alanı ile çalışarak yön gösteren optik alet olarak tanımlanabilir. Optik aletlerin içinde çalışma prensibi en basit olan cihazdır. İki temel tipi vardır. Bunlardan ilki standart oklu pusula, ikincisi ise prizmatik ve optik mercekli pusuladır. Oklu pusulalarda 360'a bölünmüş sabit bir kadran bulunur (Şekil 2.2). Manyetik alan etkisi ile okun kırmızı ucu kuzeye döner. Bu tip pusulalar harita - pusula ikilisi için uygun olmasına rağmen, mağara haritalama için uygun değildirler (Türkiye Mağaracılar Birliği, 2007).



Şekil 2.2 Sabit kadranlı pusula

### 2.1.2 Prizmatik pusula

Mağara haritalarında kullanılan açı ölçümleri bu tip pusula ile yapılmaktadır. Sıvı dolu kapalı bir hacim içinde serbest dönme yapan hareketli bir kadran vardır (Şekil 2.3). Bu kadran 360'a veya 400'e bölünmüş olabilir. 360'a bölünmüş olanlar derece, 400'e bölünmüş olanlar grad cinsinden ölçüm yaparlar. Grad cinsinden okuma yapanlar hassas ölçümler içindir (Türkiye Mağaracılar Birliği, 2007).



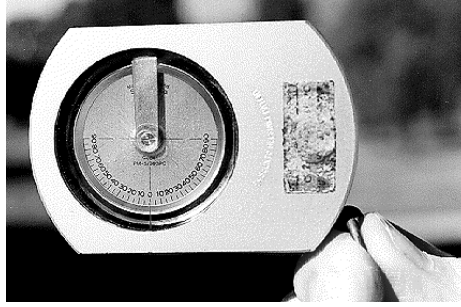
Şekil 2.3 Prizmatik teodolit pusula

Prizmatik pusulaların diğer pusulalara göre en büyük avantajı milimetre hassasiyetinde nokta okumaları yapabilmeleridir. Mağaracılıkta tercih edilmelerinin bir diğer sebebi de haritaların kâğıda aktarımları sırasında da kullanılabilmesidir (Türkiye Mağaracılar Birliği, 2007).

### 2.1.3 Eğimölçer

Prizmatik pusulalarda olduğu gibi kapalı bir hacim içinde ve sıvı içerisinde serbest hareket eden bir kadrandan oluşur (Şekil 2.4). Optik olmayan tipleri de vardır. Kadran dışından ve mercekten bakılınca aynı değerlerin okunabilmesi için tasarlanmışlardır. Kadranda bir tam dairenin yarısı alınmış ve bu yarım daire iki eşit parçaya bölünmüştür. Dışarıdan bakış sırasında bu çeyrek dairelerdeki değerler okunur.

Her çeyrek daire pozitif ve negatif değerlere bölünmüştür (0, +90 ve 0, -90 derece). Mercekten bakıldığında ise yüzdelik değeri görülmektedir. Kadranın kasa içindeki çeperi yaklaşık 3mm'dir. Bu 3mm'nin bir tarafı derece bir tarafı yüzde cinsinden değer verir (Türkiye Mağaracılar Birliği, 2007).



Şekil 2.4 Eğimölçer

#### 2.1.4 Şerit metre ve teleskopik metre

Şerit metre uzunluk, genişlik ve yükseklik ölçümlerinde kullanılırlar. Çelikten veya plastikten yapılmışlardır. Plastik olanlar mağara içi ölçümler için daha uygundur. 20, 25, 30 veya 50 metre uzunluğunda olurlar.

Teleskopik metre içi içe geçmiş şekildedirler. Her bir parçası bir metre uzunluğundadır. Beş metreye kadar yükselirler. Alüminyumdan yapılmışlardır. Yedi metreye kadar olan tavan yüksekliklerinin ölçümü için idealdirler.

#### 2.1.5 Poligon kazıkları ve prizma

Poligon kazıkları hassas ölçümlerde poligon noktalarının tespit edilmesinde kullanılırlar. Ahşap, alüminyum, demir veya çelikten olabilirler. Prizma ise salonlarda

hassas ölçümlerde kullanılır. Dik inmeye ve dik çıkmaya yarayan optik aletlerdir (Türkiye Mağaracılar Birliği, 2007).

### 2.1.6 Not defterleri ve ölçüm çizelgeleri

Not defterleri mağara içinde yapılan gözlemleri not etmek için kullanılırlar. A4 kâğıdının yarısı kadar olanlar idealdir. Mukavva altlıklı ve plastik kaplı olmaları ömürlerini uzatır.

Ölçüm çizelgeleri A4 boyutunu geçmemelidirler. Sert tabanlı ve plastik kaplı olmalıdırlar. İdeal boyutu A4 kâğıdının yarısı kadar olanlarıdır. Anlaşılabilir olması için önceden düzenlenmelidirler. Öncü ve artçı grup çizelgeleri olarak iki parçaya ayrılır (Çizelge 2.1 ve Çizelge 2.2).

Çizelge 2.1 Öncü kayıt çizelgesi

İst.	En (m)	Yükseklik (m)	Özellikler	Kesit Şekli
1				
2				
3				
4				

Çizelge 2.2 Artçı kayıt çizelgesi

İst.	Mesafe (m)	Yükselti (derece)	Yön (derece)	Özellikler	Kesit Şekli
0					
1					
2					
3					
4					



## 2.2 Ölçümler

Ölçüm süreci arazide meydana gelir. Arazi çalışmaları ise hazırlık, gözlem, ölçme, taslak ve kayıt altına alma süreçlerini kapsar. Bu süreç ise belli bir disiplin altında gerçekleşir.

### 2.2.1 Hazırlık aşaması

Bir mağarada ölçüme başlanmadan önce ölçüm sırasında kullanılacak tüm araç ve gereçlerin kusursuz olarak işler durumda olması oldukça önemlidir. Bir mağaracı için en önemli faktörlerden birisi de zamanın iyi kullanılmasıdır. İşler durumda olmayan her alet zaman ve enerji kaybına neden olacaktır. Pusula, eğimölçer, nivo ve teodolit gibi aletler yüzeyde kontrol edilmeli ve varsa ölçüm hata payları not edilmelidir. Ölçüm yapılacak mağara gps veya pusula yardımı ile topografik harita üzerinde belirlenmelidir.

### 2.2.2 Gözlem aşaması

Hazırlık aşamasında kullanılan topografik haritanın arazide hâkim bir tepeye çıkılarak sağlanması yapılmalı varsa gerçek topografya ile harita arasındaki farklar tespit edilmelidir. Öğrenilen jeolojik bilgiler elden geldiği kadar arazide tespit edilmeye çalışılmalıdır. Gerekliyse örnekler alınmalıdır. Mağaranın konumu incelenmeli, güncel veya eski dönemlerdeki konumu belirlenmeye çalışılmalıdır. Yüzeyde bu çalışmalar yapıldıktan sonra kalabalık olmayan bir ekiple mağaraya bir ön giriş yapılmalıdır. Bu giriş sırasında mağaranın fiziksel şekli incelenmeli, kabaca poligon noktaları belirlenip hafızada tutulmalıdır. Gözlem sonucu elde edilen bilgilerle daha sonra anlatılacak olan ölçüm tekniklerinden hangisinin daha avantajlı olacağına karar verilmelidir. Yapılan gözlem sonrasında haritalama ekibinin sayısı da ortaya çıkmalıdır. Kalabalık bir ekip karmaşaya, az bir ekip ise zaman kaybına yol açacaktır. İdeal ölçüm ekibi en az üç kişiden oluşmalıdır. Bunlar en az açılış değerlerini okuyan ve

not eden kiři ile uzunlukları ölçen iki kiřiden oluřmalıdır (Türkiye Mağaracılar Birlięi, 2007).

### **2.2.3 Ölçme/taslak ve kayıt altına almak**

Bu aşamada ölçüm başlamıştır. Sistem birbiri ardınca seçilen poligon noktaları arasındaki açısal, metrik değerlerin okunması şeklinde işler. Gözlem aşamasında belirlenen ölçüm teknięi uygulamaya konulur. Ölçüm sırasında mağaranın taslaęı çıkartılır, seçilen poligonlar bu taslak üzerinde işaretlenir ve gerekli kayıtlar tutulur.

## **2.3 Mağara Ölçümleri ve Standardı**

İngiliz Mağara Araştırma Birlięi (British Cave Research Association, BCRA) mağara ölçümlerindeki farklılıkları belirli standartlar dâhilinde derecelendirebilmek için 1976 yılında aşağıda belirtilen yöntemi kabul etmiştir. Günümüzde birçok ülkede yaygın olarak kullanılan ve yurdumuzda da TMB (Türkiye Mağaracılar Birlięi) tarafından kabul edilen bu ölçüm standardı iki temel kıstasa dayanır. Bunlar; orta hat hassaslık derecelendirmesi ve ayrıntı belirleme sınıflandırmasıdır (Türkiye Mağaracılar Birlięi, 2007).

### **2.3.1 Orta hat hassaslık derecelendirmesi**

Bir mağaranın ölçümü esnasında, mağara içindeki galeri ve boşlukların mümkün olduęu oranda ortasından geçen bir hat boyunca ve gereken her noktada yeni bir istasyon kurularak uzunluk, pusula ve eğim ölçümü alınır. “Hassaslık” değerleri, her istasyon ölçümünde, gerçek değerlerden ne kadar sapıldığını gösterir (Çizelge 2.3).

Çizelge 2.3 Orta hat hassaslık derecelendirmesi

Derece 1	Hiçbir ölçüme dayanmayan kroki veya taslaklar
Derece 2	Gerekli durumlarda birinci ve üçüncü dereceler arasında kalan bir kroki için kullanılabilir.
Derece 3	Yatay ve düşey açılarda +/- 2,5°, uzunluklarda ve istasyon noktasında +/- 50 cm'den daha az ölçüm hatası
Derece 4	Beşinci derecenin şartlarını sağlayamayan, ancak üçüncü dereceden daha hassas ölçümler için kullanılabilir.
Derece 5	Yatay ve düşey açılarda +/- 1°, uzunluklarda ve istasyon noktasında +/- 10 cm'den daha az ölçüm hatası
Derece 6	Yatay ve düşey açılarda +/- 0,5°, uzunluklarda ve istasyon noktasında +/- 2,5 cm'den daha az ölçüm hatası
Derece x	Pusula ve klino yerine teodolit kullanılarak yapılan ölçümler

### 2.3.2 Ayrıntı belirleme sınıflandırması

Yukarıda bahsedilen, istasyonlar arası ölçüm sırasında kaydedilen, mağarada mevcut oluşum, biçimlenim, galeri genişliği, yüksekliği gibi detayların kaydedilme biçimidir (Çizelge 2.4).

Çizelge 2.4 Ayrıntı belirleme sınıflandırması

Sınıf A	Bütün ayrıntılar hafızaya dayalı
Sınıf B	Galeri ayrıntıları ölçüm yapılmaksızın mağara içinde kaydedilmiş
Sınıf C	Galeri ayrıntılarının ölçümleri sadece istasyonlarda yapılmış
Sınıf D	Bütün ayrıntıların ölçümleri istasyonların yanı sıra, galerinin şeklinin, genişliğinin, yönünün değiştiği durumlarda, iki istasyon arasında da yapılmış

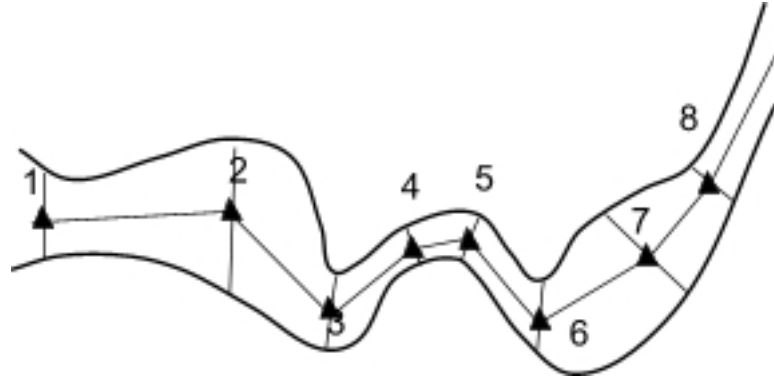
### **2.3.3 Mağara ölçümünde basit yöntemler**

Ön etütleri yapılan mağaralarda çeşitli yöntemlerle ölçümler alınır. Bunlar basit ve ileri yöntemlerdir. Klasik ölçümde 4 çeşit yöntem kullanılır. Bunlar; orta hattın çıkartılması, duvardan duvara ölçüm yöntemi, karışık yöntem ve ışımsal yöntemdir.

#### **2.3.3.1 Orta hattın çıkartılması yöntemi**

Mağara girişinin hemen dışında ve hemen önünde olmak kaydıyla bir P0 (sıfır) poligonu tespit edilir. Bu nokta bağlama noktasıdır. Birden fazla başlangıç noktası tespit edilebilirse bu hesapların karışma ihtimalini arttıracaktır. P0 (başlangıç noktası) poligonunda duran kişi mağara girişine doğru olan açığı okur. Burada, bu kişinin dikkat edeceği en önemli husus pusulanın yere tam paralel olmasıdır. P0 poligonundan P1 poligonuna kuzeyle olan açı okunur ve not edilir. P0'dan P1'e uzunluk ölçümü yapılmaz, fakat P1 poligonunda genişlik, alt ve üst uzunluklar ölçülür. Bu sadece giriş ağzının ortaya çıkması ve ölçümlerin coğrafi haritalarda kullanımının kolaylaştırılması içindir. Daha sonra P0 poligonundaki kişi P1 poligonuna gelir, diğer ekip üyeleri seçilen P2 poligonuna gider. P1 poligonu ile P2 poligonu arasında uzunluk, genişlik, eğim ve doğrultu açısı tespit edilip çizelgeye kayıt edilir. Ölçümü yöneten kişi aynı zamanda krokiyi çizer ve bu kroki üzerinde poligon noktalarını işaretler. Bu işlemler seçilen her poligon noktasında tekrar edilir (Şekil 2.5).

Seçilmiş her poligon noktasının gidiş istikametine göre sağ tarafta kalan kısmı "P1 sağ", sol tarafta kalan kısmı "P1 sol" değeridir. Şerit metrenin tutulduğu noktanın altı "P1 alt", üstünde kalan kısmı "P1 üst"tür. Bunlar tercihe göre değişebilir. Metre eğer zeminle sıfır durumda ise P1 alt sıfır değerini alır ve metrenin üstünde kalan kısım üst değerdir (Türkiye Mağaracılar Birliği, 2007).



Şekil 2.5 Orta hat yönteminde poligon noktalarının durumu

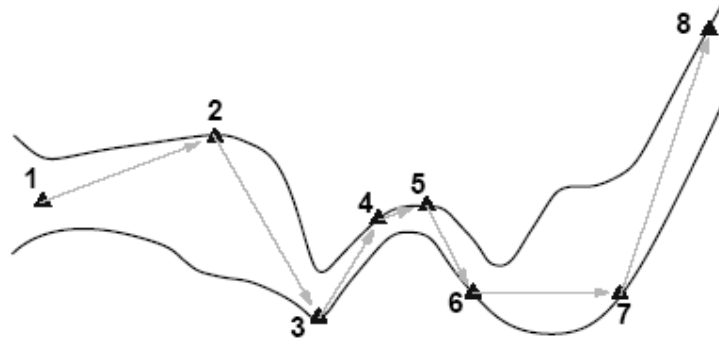
### **2.3.3.2 Duvardan duvara yöntemi**

Bu yöntemde de bir başlangıç noktası belirlenir. Orta hat yönteminde olduğu gibi P0'dan P1'e doğrultu açısı ölçülür. P1 poligonundan itibaren poligon noktaları pasajın ortasından değil, zig - zaglar çizecek şekilde karşılık duvar diplerinde belirlenir. Fakat bu yöntemin çeşitli dezavantajlar vardır. Bunlar zaman zaman duvar diplerindeki derin çatlaklar, su dolu rezervuarlarda hareket zorluğu ve tavan yüksekliğinin ölçülmesi şeklinde haritacının karşısına çıkar (Türkiye Mağaracılar Birliği, 2007).

### **2.3.3.3 Karışık yöntem**

Başlangıç noktası yine sabittir. En kullanışlı yöntemdir. Ölçümü yöneten kişi poligonların seçiminde tamamen serbesttir. Orta hat ve duvardan duvara yönteminin birlikte kullanımı ile gerçekleşir. Bu yöntemde de işlemler aynıdır. Farklı olan tek tarafı Şekil 2.6'dan da anlaşılacağı üzere poligon noktalarının belli bir sırayı takip etmemesidir. Örneğin P2 noktası orta nokta iken P3 noktası duvar dibindedir. Bu durumda açı değerlerini okuyan kişi P2 – P3 doğrultusunu okuyacaktır. Burada P3 sol değer sıfır olurken P3 sol değer sol tarafta kalan mesafe olacaktır. Bu yöntem haritacıya

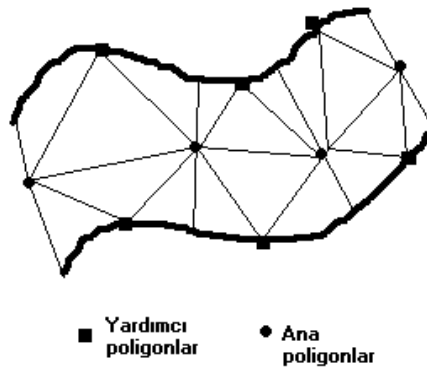
büyük hareket kabiliyeti ve insiyatif kullanmayı sağlar (Türkiye Mağaracılar Birliği, 2007).



Şekil 2.6 Karışık yöntem

#### **2.3.3.4 Işınsal yöntem**

Oldukça karmaşık ve zaman alıcı bir yöntemdir. Her poligon noktasında tespit edilen her yardımcı noktaya doğru açı değerleri okunur ve uzunluklar ölçülür (Şekil 2.7). Bu yöntem sadece salon ölçümlerinde kullanılmalıdır. Pasajlarda önemli ölçüde zaman ve enerji kaybına neden olur (Türkiye Mağaracılar Birliği, 2007).



Şekil 2.7 Işınsal ölçüm yöntemi

Mağara ölçümlerinde poligon seçimi son derece önemlidir, tecrübe ve pratiklik gerektirir. Bu ise zamanla ve pratikle kendiliğinden oluşur. Poligon noktaları istenilen her yere atılabilir. Ancak, mağara içinde önemli fiziksel değişimlerin olduğu noktalarda muhakkak poligon atılmalıdır. Bu değişimler yükselti farkları, iniş - çıkışlar, havuzlar, göller, önemli damla taş oluşumları (büyük sarkıtlar, dikitler, sütunlar, travertenler...), tehlike oluşturan yerler, su çıkışları ve batışları (sifonlar), yan cepler, tavan yüksekliğinin düştüğü yerler, jeolojik birimler, yapısal farklar (faylar, breşler...), kamp yerleri, zemin ve eğimde değişimler, büyük bloklar, göçmeler, arkeolojik ve antropolojik birikmeler ve örnek noktalarıdır (Türkiye Mağaracılar Birliği, 2007).

#### **2.3.4 Mağara ölçümünde ileri yöntemler**

Turizm, mimari, inşaat amaçlı olarak kullanılacak mağaralarda pusula-eğimölçer ve metre ile yapılan ölçümler verimli olmamaktadır. Çünkü uygulama yapılacak mağaralarda matematiksel işlemler gerçekleştirileceğinden dolayı haritanın son derece hassas olması gerekmektedir. İşte bu amaçla kullanılacak mağaralarda takeometrik ölçümler yapılır. Bu ölçümler ise takeometri ve nivo adı verilen optik cihazlarla gerçekleşir. Bu cihazlarla yapılan ölçümlere takeometrik ölçümler denilir. Oldukça zaman alıcı ve hassas çalışmayı gerektirirler. Takeometrik ölçüm, kısaca noktaların sabit bir poligon noktasına göre kot farklarının ve uzaklığının bulunmasına denir. Işımsal ölçüm yöntemine benzerlik gösterir (Türkiye Mağaracılar Birliği, 2007).

#### **2.4 Mağara Haritası Çizim Yöntemleri**

Bir mağarayı haritalamanın arazi ve büro çalışmaları olmak üzere iki aşaması bulunmaktadır. Önceki bölümlerde anlatılanlar arazi çalışmalarının kısa bir özetiştir. Büro çalışmaları ise arazide alınan değerlerin teknik kurallara bağlı kalınarak kâğıda aktarılmasıdır. Bir mağara haritasını doğru çizebilmek teknik resim kavramlarını ve el, bilgisayar veya semiyolojik yöntemlerle çizim tekniklerinden birini veya bir kaçını bilmeyi gerektirir (Türkiye Mağaracılar Birliği, 2007).

### 2.4.1 Elle çizim

Elle çizim yönteminde açı değerlerini kâğıda aktarmak için iki seçenek bulunmaktadır. Bunlardan ilki pusulayı kullanmak (ki oldukça pratik olmakla beraber hata payı fazladır), ikincisi ise açıölçerlerle açı değerlerini kâğıda aktarmaktır. Diğer malzemeler her iki seçenekte de aynıdır. İlk taslaklar tam boy bir eskiz kâğıdına yapılabilir. Bunda amaç kâğıt üzerinde ne tarafa uzanacağı belli olmayan mağaranın tamamını tek bir paftaya sığdırabilmektir. Kâğıdın tercih edilen başlama noktası ortası olmalıdır. Böylece sağa, sola, aşağıya veya yukarıya uzanım olması durumunda her zaman yeterli alan olacaktır. Başlangıç noktasını yani P0 poligonunu kâğıda işaretledikten sonra ölçeğin belirlenmesi gerekir. Eğer payda küçük yapılırsa harita büyük, payda büyük yapılırsa harita küçük olacaktır. Bunu belirlemenin en kolay yolu ilk iki poligon arasındaki yatay mesafeyi temel alarak bir ölçek belirlemektir (Türkiye Mağaracılar Birliği, 2007).

### 2.4.2 Pusula ile çizim

Ölçek belirlendikten sonra sıra çizelgedeki değerleri kâğıda aktarmaya gelir. Bunun için pusula kâğıt üzerinde belirlenen noktaya koyulur. Pusula üzerinde sabit bulunan kırmızı çizgi ile çizelgedeki açı değeri üst üste gelinceye kadar pusula eksenini etrafında döndürülür. Kırmızı çizgi ile say değeri çakıştığında pusulanın kenarı veya gönye ile pusula kenarına paralel bir çizgi çizilir. Artık yerindeki P0 ile P1 poligonu arasındaki doğrultu kâğıt üzerine aktarılmıştır. Sıra P0 poligonu ile P1 poligonu arasındaki uzunluğun belirlendiği ölçeğe göre çizilmiş olan paralel çizgi üzerinde işaretlemeye gelmiştir. Bu işlem doğrultu açı değerleri bitinceye kadar devam ettirilir. Eğer eğim ölçümleri çok yüksek ise (20 dereceden negatif veya pozitif yönde) fazla ise ilgili trigonometrik bağıntılarla veya izdüşüm yöntemi ile boy kısalması hesaplanır. Fakat ölçek büyük paydalı ise 70 – 90 derece arasındaki eğimler hariç bu kısalma veya uzama hiçbir zaman çizime yansiyacak kadar büyük olmayacaktır. Bu tip hesaplamalar yapmak istenilmiyorsa metre ölçüm sırasında her zaman zemine dik değil paralel tutulmaya çalışılmalıdır. Gerçek uzunluğu böylece ölçülür. İşlem sonunda ortaya



birbiri ucunda eklenmiş çeşitli doğrultularda bir hat çıkacaktır ki buna orta hat denir. Orta hat bir mağara haritasının bel kemiğidir. Fakat her zaman orta hat ortada kalacak diye de bir şey yoktur. Bu durum kullanılan ölçüm yöntemine bağlıdır (Türkiye Mağaracılar Birliği, 2007).

### 2.4.3 Açılçerle çizim

Bunun için 360 dereceye veya 400 grada bölünmüş yuvarlak açılçerlerin bulunması gereklidir. Diğer araçlar aynıdır. P0 poligonuna açılçerlerin merkezi yerleştirilir. Kayıt edilmiş olan ilk açı değeri açılçer üzerinde tespit edilir. Bu doğrultu P0 poligonu ile P1 poligonu arasındaki doğrultu açısıdır. P1 poligonun tespit edildikten sonra P1 sağ ve sol değeri cetvelle ölçekli bir şekilde işaretlenir. Daha sonra P1'den P2'ye olan açı değeri kâğıt üzerine işaretlenir. P1 poligonu ile P2 poligonu iki poligon arasındaki uzunluk değerine bağlı kalınarak birleştirilir. P2 poligonundaki sağ ve sol değerler işaretlenir. İşlem bu şekilde bir biri ardınca devam eder (Türkiye Mağaracılar Birliği, 2007).

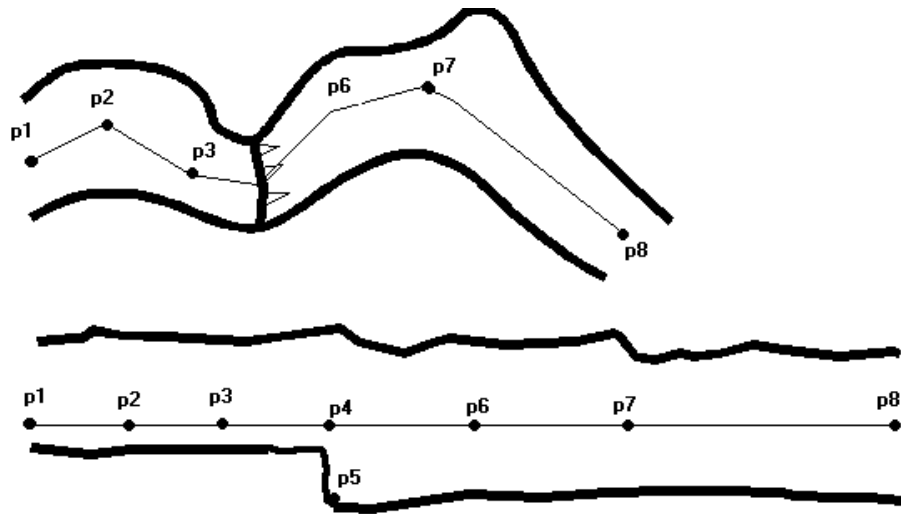
### 2.4.4 Kesit çizimleri

Kesit çizimleri temelde plandan kesit alımlarında izdüşürme yöntemine dayanmaktadır. Bu teknikte koordinat sisteminde 1. bölge sabit alan olarak kabul edilir ve plan bu alana yerleştirilir. Birinci bölgeye yerleştirilen plan aşağıdaki gibi olacaktır. Yatay düzleme paralel olarak çizilmiş iki boyutlu şekillerden iz düşürerek yatay düzleme dik konumdaki kesit görüşleri elde edilebilir.

İzdüşürülmüş kesitler her zaman tercih edilmelidirler. Çünkü bu kesitler gerçek görüntüler olup haritayı kullanan kişinin yanlış anlaşılma olasılığını en aza indirirler. İzdüşürüm metodu kullanılırsa her poligon arasındaki boy kısaltmalarının tek tek hesaplanması gereği ortadan kalkacaktır (Türkiye Mağaracılar Birliği, 2007).

### 2.4.5 Boyuna açılmış kesitler

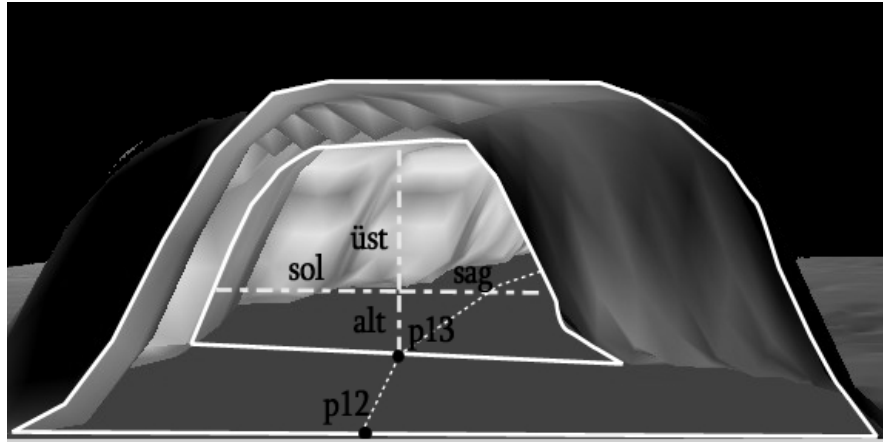
Boyuna açılmış kesitler gerçek durumu yansıtmazlar. Bu kesitte mesafeler her zaman olduğundan daha fazla görünür. Bunda en büyük sebep poligonlar arasındaki açısal farklardan kaynaklanan boy kısaltmalarının ortadan kalkmasıdır. Açık kesit çiziminde birbiri ardınca alınan poligonların açısal değerleri yok sayılarak poligonlar arası mesafeler birbiri ardınca eklenir. Bu durumda kesit doğrultusu tek bir doğrusal eksen üzerinde oluşur. Hiçbir zaman boy kısaltmaları meydana gelmez. Uzun bir çizim oluşur. Tek faydası kot farklarının daha kolay anlaşılmasıdır. Bu kesit şekli mecbur kalınmadığı sürece kullanılmamalıdır. Uluslar arası teknik resim kurallarına aykırıdır. Şekil 2.8’de gösterilen poligonlar arası boy uzamalarına bakıldığında, mağara hemen hemen 1/3 oranında uzamıştır. Bu yöntemde her poligon arasındaki eğimden kaynaklanan boy uzamaları trigonometrik bağıntılarla tek tek hesaplanmak zorundadır (Türkiye Mağaracılar Birliği, 2007).



Şekil 2.8 Boyuna açılmış kesitte boy uzamaları

### 2.4.6 En kesitler

Mağara oluşumları bilinen geometrik şekillerle açıklanamayacak yerlerdir. Bu sebepten dolayı kesit ve planlar çoğu zaman bir mağarayı ifade etmeye tek başlarına yeterli olmazlar. İfade zorluğunun ve önemli ayrıntıların bulunduğu yerlerde en kesitler çizilirler. En kesitler mağara galeri veya salon genişliğinin uzanım yönüne dik olarak kesildiği düşünüldüğünde ortaya çıkan görüntülerdir. Planda noktalı kesik veya kesik çizgilerle gösterilirler (Şekil 2.9).

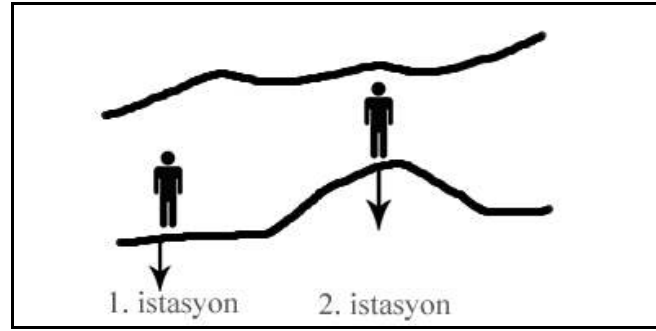


Şekil 2.9 Bir mağara galerisinde en kesit kavramı

## 2.5 Özetle Geleneksel Harita Çıkartma Yöntemi

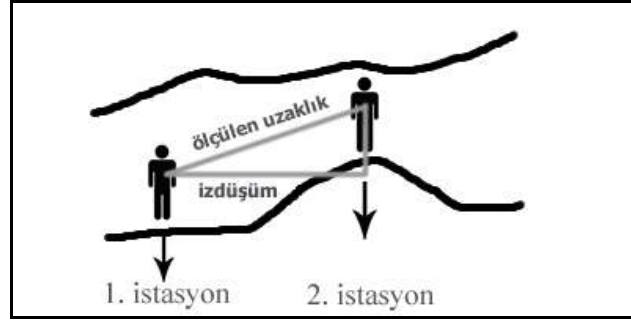
Eskişehir Mağara Araştırma Derneği Eğitim Notları'na (2005) göre en az iki kişiden oluşan bir ekibin oluşturulması bir mağara haritasının çıkartılmasında kullanılacak özel araçlar ve alınacak ölçümlerin düzgün kaydedilmesi açısından önemlidir. Harita oluşturulması için kullanılacak verilerde eksik, hatalı ölçüm veya kayıtlar haritanın tümünde yanlışlığa neden olmaktadır. En az iki kişiden oluşan bu grubun geleneksel harita çıkartma yöntemi için aralarındaki görev paylaşımına bağlı olarak yapmaları gereken işler şunlardır:

- Mağarada, kesin ölçülebilir istasyonlar belirlenir. Mağaranın girişi sıfırinci istasyon olmak üzere, ulaşılabilir ve ölçülebilir makul uzaklıkta birinci istasyon olabilecek bir yer belirlenir. Artçı grup sıfırinci istasyonda, öncü grup birinci istasyonda durur. Şekil 2.10 incelendiğinde öncü grubun ikinci istasyonda ve artçı grubun birinci istasyonda bulunduğu görülmektedir.



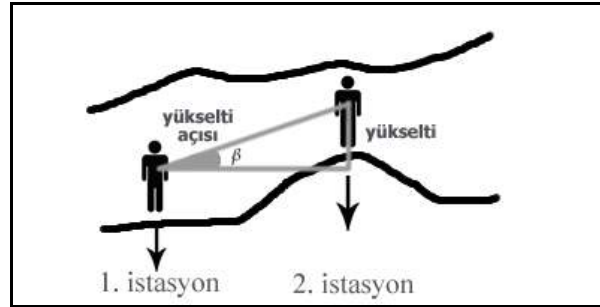
Şekil 2.10 İstasyonların isimlendirilmesi

- Öncü grup tarafından birinci istasyonun eni, yüksekliği, kesit biçimi ve gerekli görülen diğer özellikler kaydedilir.
- Artçı grup tarafından sıfırinci ve birinci istasyonların arasındaki mesafe ölçülür ve iki istasyon arasında kalan mağara bölümünün belirgin özellikleri kaydedilir. Şekil 2.11 incelendiğinde ölçülen uzaklığın artçı ve öncü gruba bağlı olduğu görülmektedir. Burada öncü grup ikinci istasyonda bulunurken birinci istasyondaki artçı gruba göre göz hizası baz alınarak ölçüm yapılmaktadır. Yani bir istasyondan sonraki istasyona nişan olarak ölçüm yapıldığı için verilerin modellenmesi sırasında artçı istasyon ölçülen uzaklık doğrusuna dik olacak şekilde yatay eksen üzerinde döndürülmelidir.



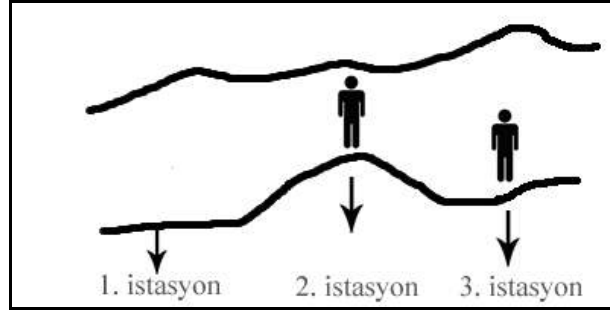
Şekil 2.11 İstasyonlar arası izdüşüm ve ölçülen uzaklık

- Şekil 2.12’te görüldüğü gibi sıfırcı istasyondan bakıldığında, birinci istasyonun yükselti açısı ölçülerek kaydedilir.



Şekil 2.12 İstasyonlar arası yükselti açısı ve yükselti

- Artçı grup tarafından birinci istasyonun, sıfırcı istasyona göre yönü belirlenerek kaydedilir.
- Sıfırcı istasyondaki artçı grup birinci istasyona gider, birinci istasyondaki öncü grup bir sonraki ikinci istasyonu belirleyerek orada durur. İkinci adımla başlayan işlemler bu sefer birinci ve ikinci istasyonlar için yapılır. Şekil 2.13’te üçüncü adımda öncü ve artçı grupların bulunmaları gereken istasyonlar belirtilmektedir.



Şekil 2.13 Diğler istasyonlarının isimlendirilmesi

Mağara içi çalışmalarda belirlenen ölçüm ve kayıtlar, mağaranın rakamsal ölçüleri yanında, diğler özelliklerini de ayrıntılı biçimde belirleyecek şekilde yapılmaktadır. Yapılan kayıtlar, başka bir kişi tarafından değerlendirilebileceğinden belirli bir sistematığe göre belirlenen ölçü birimleri kullanılarak, yanlış anlaşılmayacak, açık ve net bir biçimde kaydedilmesi gerekmektedir. Bu nedenle, genellikle çalışma öncesi hazırlanmış kayıt çizelgeleri kullanılmaktadır. Kayıt çizelgesi, öncü ve artçı gruplar için iki farklı çizelgeden oluşur. Çizelge 2.10 ve Çizelge 2.11’de öncü ve artçı grup için hazırlanmış kayıt çizelgeleri bulunmaktadır (Eskişehir Mağara Araştırma Derneği Eğitim Notları, 2005).

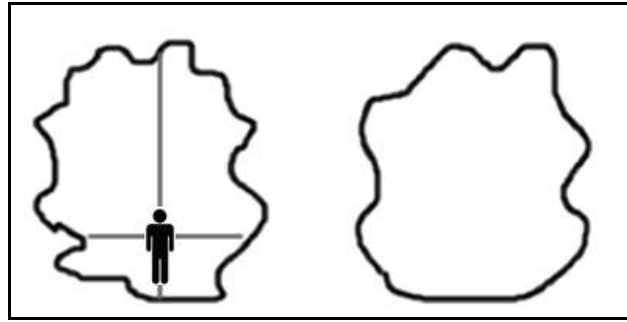
Çizelge 2.1’de yer alan öncü kayıt çizelgesinde, istasyon numarası bir ile başlar. En, yükseklik metre cinsinden girilir. Kesitin özellikleri olarak jeolojik, biyolojik ve arkeolojik özellikleri kaydedilir. Her bir istasyon için dikine kesit şekli şematik olarak çizilir (Eskişehir Mağara Araştırma Derneği Eğitim Notları, 2005).

Çizelge 2.2’de yer alan artçı kayıt çizelgesinde istasyon numarası sıfır ile başlar. Mesafe istasyonun bir sonraki istasyonla arasındaki uzaklıktır ve ölçü birimi metre olacak biçimde kaydedilir. Yükselti, bir sonraki istasyonun, yatay düzleme göre eğim açısıdır ve ölçü birimi derece olacak biçimde kaydedilir. Yön, bir sonraki istasyonun kuzeye göre açısıdır ve ölçü birimi derece olacak biçimde kaydedilir. Kesitin özellikleri olarak jeolojik, biyolojik ve arkeolojik özellikleri kaydedilir. Her bir istasyon için

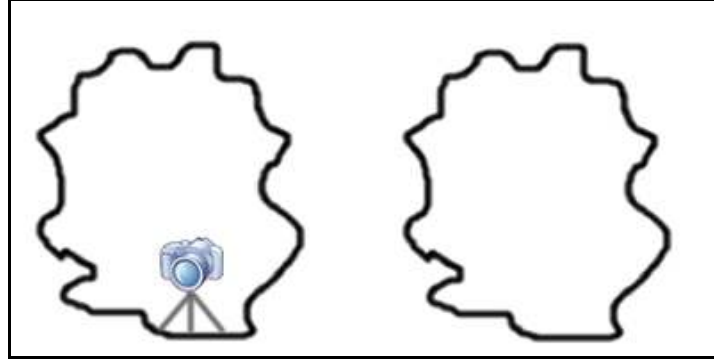
dikine kesit şekli şematik olarak çizilir (Eskişehir Mağara Araştırma Derneği Eğitim Notları, 2005).

## 2.6 Fotoğrafla Haritalama Yöntemi

Geleneksel harita çıkartma yönteminde araştırmacının hassasiyetine olan bağımlılık ölçümlerin gerçeği ifade etmesinde büyük bir etmendir. Araştırmacı her ne kadar büyük bir özveri ile ölçümler yapsa da kesit detayının el ile çizilmesi, modelleme yapılması sırasında Şekil 2.14'te görüldüğü gibi bir kesite ait sınırlardaki detayların kaybolmasına yani gerçeği yansıtmayan görüntülerin elde edilmesine neden olmaktadır. Bu nedenle, modelleme esnasında yeni bir harita çıkartma tekniği olan fotoğrafla haritalama yönteminin kullanılması Şekil 2.15'de görüldüğü gibi elde edilen verilerin gerçeği birebir yansıtmasını sağlamaktadır (Canbek ve Adar, 2005).



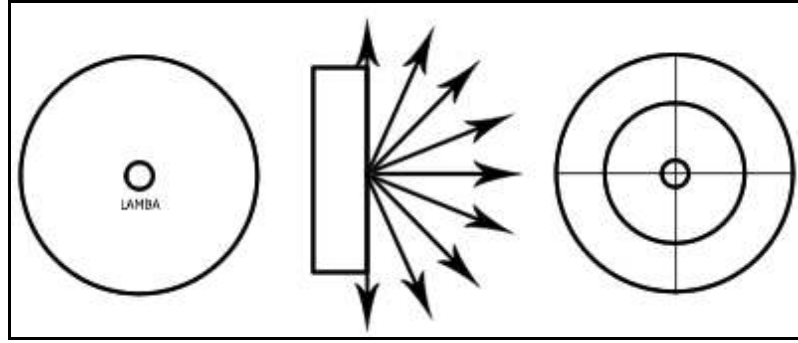
Şekil 2.14 Geleneksel harita çıkartma yöntemi



Şekil 2.15 Fotoğrafla haritalama yöntemi ile harita çıkartma

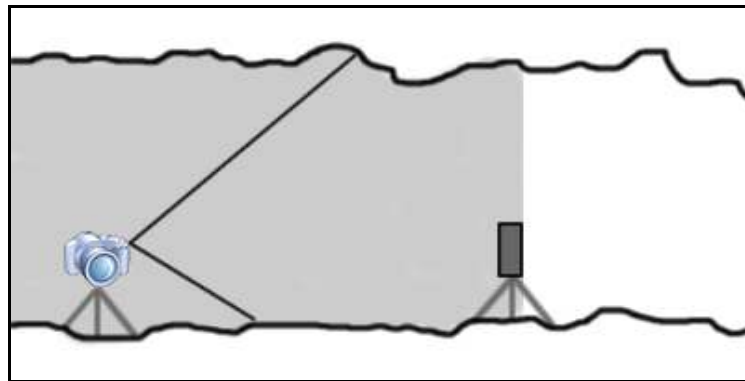
Fotoğrafla haritalama yöntemi geleneksel harita çıkartma yöntemlerine göre daha az verinin araştırmacı tarafından kayıt altına alınmasını sağlamaktadır. Araştırmacı her bir istasyonun bulunduğu konum ile birlikte ilgili fotoğraf numarasını kayıt altına almaktadır. İstasyonun bulunduğu konuma koyulan 180 derecelik ışık kaynağı ve istasyonun gerisine koyulan bir fotoğraf makinesi ile çekilen resimler modelleme yapılırken kullanılmaktadır. Işık kaynağı keskin, gölgeli sınır noktalarının oluşabilmesi için küçük ve yüksek yoğunlukta olmalıdır. Yarım küreyi eşit dağılımla aydınlatabilecek ışık kaynağı oluşturabilmek için aydınlatacak lamba bir tarafı açık silindirik bir yüzeyin Şekil 2.16'nın sol tarafında gösterildiği gibi tam ortasına koyulmalıdır. Işık kaynağını oluşturacak sistem üç bacaklı fotoğraf sehpası üzerine yerleştirildikten sonra Şekil 2.16'nın sağ tarafında gösterildiği gibi silindirik yapının arka tarafına not edilmiş yarıçaplarla daireler çizilir. İç kısımda kalan daire resim için referans noktaları oluşturacaktır (Canbek ve Adar, 2005).



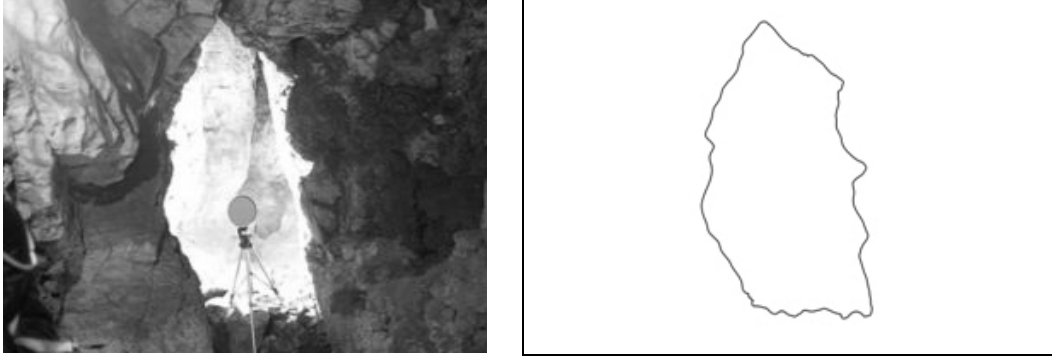


Şekil 2.16 Işık kaynağı

Çekilen resimlerin modelleme sırasında kullanılabilmesi için kullanılacak fotoğraf makinesi sabit lensli dijital bir makine olabilir (Şekil 2.17). Yakınlaştırma özelliği olan lensler veya dijital yakınlaştırma resim üzerinde bozulmaları oluşturacağı için kullanılmamalıdır. Oluşan bozulmalar resim işleme tekniklerinin uygulanması sırasında algoritmalar sayesinde düzeltilmektedir. Yüksek çözünürlüklü fotoğraf makinelerinin kullanılması ölçüm duyarlılığını arttıracaktır (Canbek ve Adar, 2005). Şekil 2.18’de fotoğraf ile haritalama tekniğinde fotoğraf makinesinden örneklenen bir görüntünün modelleme yapılmadan önce geldiği durum gösterilmektedir.



Şekil 2.17 Fotoğrafla haritalama yönteminin uygulanışı



Şekil 2.18 Fotoğrafla haritalama yöntemi ile elde edilen görüntü

Araştırma bölgesinden alınan resmin kullanılabilir hale getirilebilmesi amacıyla resim işleme teknikleri uygulanmaktadır. Bu teknikler paralel izdüşüm dönüşümü, rotasyon, resim iyileştirme yöntemi ve gölge filtresidir.

### 2.6.1 Paralel izdüşüm dönüşümü

Nesnenin ışık demetleri, fotoğraf makinesinin lenslerinden geçer ve fotoğraf makinesinin sensörlerinde poz oluşturur. Kamera lensleri ve dedektörler resmin çerçeve sınırlarına yakın bölümlerinde bozulmaya neden olur. Kesitten daha uzakta fotoğraflar çekmek bozulma sorunlarını hafifletir. Fakat mağaraların yapıları gereği küçük alanlarda bu durum yapılamaz. Çoğunlukla araştırmacılar resimleri kısa mesafelerde çekmek zorunda kalırlar. Ek olarak geniş açılı lensler daha çok bozulmaya neden olmaktadır. Resimler üzerinde ölçümler yapabilmek için oluşan bozulmalar düzeltilmelidir. Bu amaçla, 16x16'lık siyah beyaz kontrol yüzey resmi üzerinde fotoğraf çekilir ve yazılımla bozulmalara düzeltecek düzeltme filtresi oluşturulur. Kullanılan her farklı lens için düzeltme filtresi oluşturulmalıdır.

### 2.6.2 Rotasyon

Bir resim üzerinde yapılacak hesaplamaları doğru elde edebilmek için fotoğraf makinesi kesit yüzeyine dik olarak konumlandırılmalıdır. Mağaraların yapılarından ötürü, bu işlem kolaylıkla yapılamamaktadır. Bu nedenle, mağaranın fotoğrafı çekilirken kullanılan fotoğraf makinesinin kesit yüzeyine herhangi bir açı ile yerleştirilmesine olanak sağlayacak resim içerisindeki bir dairenin normalleştirilmiş eksenlerini bulmak için yöntem geliştirilmiştir. Resimler alındıktan sonra ışık kaynağının arka tarafında yer alan bir çemberin kullanımı ile normalleştirilmiş kesit yüzeyi ve fotoğraf makinesinin yönü hesaplanmaktadır. Kesit yüzeyinin normalleştirilmiş eksenleri üzerinde yer alan bir fotoğraf makinesinden alınan bir resim gibi döndürülmüş resim, açısı bilindiği için “Affine” dönüşümü kullanılarak tekrar oluşturulmaktadır. Döndürme işleminden sonra resim üzerinde piksel sayımı ile ölçüm yapılabilir. Bir pikselin gerçekte ne kadar uzunluğa karşılık geldiğini belirten birim uzunluk ışık kaynağının arkasında çizilmiş olan çemberin yarıçapının hesaplanması ile elde edilir.

### 2.6.3 Resim iyileştirme yöntemi

Mağaranın doğal şartlarından ötürü alınan fotoğraflar her zaman aynı kalitede olamamaktadır. Alınan her bir fotoğraf üzerinden gürültü ve gereksiz detayların kaldırılması gerekmektedir. Bu nedenle, medyan filtre kullanılır.

### 2.6.4 Gölge filtresi

Kesiti oluşturan sınır noktaları resim üzerindeki gölge sınırlarından oluşturulmaktadır. Bu aşamada, gölge sınırları keskinleştirilir ve kesitin sınır noktalarını oluşturan noktalar tespit edilir.

## BÖLÜM 3

### ÜÇ BOYUTLU MAĞARA MODELİ OLUŞTURMA

Bir mağaranın üç boyutlu modelinin oluşturulabilmesi için esnek bir programlama dili ile birlikte işlemleri kolaylaştırması için grafik işleme kütüphanelerinden herhangi birisi kullanılmalıdır. Bu çalışmada geliştirilen yazılım çerçevesinde C# programlama dili ve OpenGL grafik kütüphanesi kullanılmıştır. Mağara modelini oluşturabilmek için kullanıcıdan alınan veriler ile ön ayar dosyasından okunan değerler C# programlama dilinde yorumlandıktan sonra OpenGL kütüphanesindeki fonksiyonlara aktarılmaktadır. Deitel'e (2006) göre C# programlama dili tamamıyla nesne tabanlı, olaya dayalı, hızlı yazılım geliştirme sürecini destekleyici özellikler içermekte ve gereksinim duyulabilecek tüm yazılım kütüphaneleri rahatlıkla dâhil edilebilmektedir. Mağara modelleme ile ilgili sürecin nasıl uygulandığını detaylı olarak incelemek işleyişi anlamayı kolaylaştıracaktır. Ayrıca, grafik işleme kütüphanelerinden en yaygın kullanılan OpenGL ve Direct3D'nin özelliklerinin neler olduğu ve niçin OpenGL'in tercih edildiğini de ifade etmek gerekmektedir.

#### 3.1 Grafik Kütüphanelerine Genel Bakış

Günümüzde yazılım sektöründe kullanılan birçok grafik kütüphanesi olmasına rağmen bu grafik kütüphanelerinden ücretsiz olarak sunulan Silicon Graphics Inc firmasının ürettiği OpenGL ile Microsoft'un ürettiği ve DirectX'in bir alt kümesi olan Direct3D yazılımları bulunmaktadır.

##### 3.1.1 Direct3D

Direct3D kütüphanesi Microsoft tarafından üretilen DirectX kütüphanesinin bir alt kümesidir. Direct3D kütüphanesi programcılara grafik donanımının avantajlarını

kullanabilmek için video donanımına direk erişimi sağlar. Direct3D kütüphanesi sadece Windows tabanlı Windows 95 ve üstü işletim sistemlerinde çalışmaktadır. Direct3D kütüphanesi üç boyutlu grafiklerin işlenebilmesi için performansın önemli olduğu noktalarda kullanılmaktadır ve donanım hızlandırmasının grafik kartının üzerinde bulunması halinde kullanımı sağlamaktadır. DirectX'in en önemli iki gücü, DirectX 8 ile gelen programlanabilir piksel desteği ve köşe gölgelendirme özelliğidir. Köşe gölgelendirme özelliği istenilen bir bölümün yapısı değiştirilerek kodlama ile güncellenebilme özelliğinin verilebilmesidir. Köşe gölgelendirme özelliği OpenGL tarafından da desteklenmektedir, ancak OpenGL içerisinde yer almayıp OpenGL eklentileri ile kullanılabilir. Grafik endüstrisinin çok hızlı gelişmesine rağmen DirectX'in güncellenme sıklığı bir yılı aşan sürelerde yapıldığı için yavaş kalmakta ve teknolojiyi takip edememektedir. Yazılım geliştirilirken grafik ile ilgili işlemlerde eğer DirectX kullanılırsa daha fazla kod yazılması gerekmektedir. Ancak, gelişen zaman içerisinde Direct3D Güncelleme 7 ile bir üçgen oluşturmak için yazılan kod satırı 800 iken, Direct3D Güncelleme 8 ile bir üçgen için yazılması gereken kod satırı 200 satıra düşmüştür. DirectX'in Windows tabanlı işletim sistemlerinde çalışması nedeni ile çoklu platform desteği sağlamak isteyen yazılımcıyı kısıtlamaktadır. Direct3D'nin kodları genel erişime de kapalı olduğu için herhangi bir işletim sisteminde çalışacak farklı dağıtımları hazırlanamamaktadır. Ayrıca, yazılımcı C programlama dili kullanmak zorundadır (Kolar, 2008).

### 3.1.2 OpenGL

OpenGL yani uzun ismi ile "Open Graphics Library" birçok farklı programlama dili ile geliştirilebilir ve birçok farklı işletim sisteminde iki boyutlu veya üç boyutlu bilgisayar grafikleri oluşturmak için kullanılmaktadır. OpenGL; Direct3D'nin desteklemediği Windows NT 4.0 işletim sisteminde, Windows'un tüm sürümlerinde, Unix - Linux türündeki tüm dağıtımlarda çalışabilmektedir. OpenGL ana yapısı ile birlikte yazılımcılara eklentiler ile geliştirme yapabileme fırsatı sunmaktadır. Açık kaynak kodlu olması nedeni ile herhangi bir firma OpenGL'in desteklediği 250 fonksiyonun üzerine kendi fonksiyonlarını ekleyebilmekte veya var olanları ihtiyaçları

doğrultusunda değiştirebilmektedir. OpenGL'in eklentilerle genişletilmesi özelliği gelişen donanım endüstrisinin yeni çıkan özelliklerinin hemen kullanılabilmesini sağlamaktadır. OpenGL hemen hemen tüm işletim sistemlerinde çalışabilir olması yanında, kararlılığı ve performansının hiç düşmemesi nedeni ile tercih edilmektedir. OpenGL internet bağlantısını kullanabilmekte, yani sunucu ve istemci yapısında programların geliştirilmesine olanak sağlamaktadır. Sunucu tarafının kararlılığı yüksek olan Linux dağıtımlarında yazılabilmesi de ayrıca geliştirilen yazılımın sorunsuz bir şekilde çalıştırılabilmesini sağlamaktadır. Günümüzde Windows işletim sistemi altında C# programlama dili ile yazılan bir uygulama, Mono aracılığı ile dosya sistemi kullanımı dışındaki kodlama üzerinde değişiklik yapmadan Linux altında kullanılabilmesini sağlamaktadır. OpenGL'e büyük bir artı kazandıran eklenti özelliği kimi durumlarda bir dezavantaja dönüşebilmektedir. Örneğin, bir yazılım içerisinde kullanılan eklentinin grafik kartına %100 bağımlı olması durumunda OpenGL'in diğer grafik kartlarında çalışmasını engelleyebilmektedir. Ayrıca, geliştirilen eklentiler farklı programlama dillerinde çalışmamaktadır, yani programlama diline bağımlı eklentiler yazılabilmektedir (Kolar, 2008).

### **3.1.3 Performans karşılaştırması**

OpenGL ile Direct3D arasında performans karşılaştırması yapabilmek birçok parametreye bağlıdır. Kimi yazılımcıya göre OpenGL hızlı çalışırken, kimisine göre de DirectX daha hızlı çalışmaktadır. Ancak, DirectX'in sadece Windows üzerinde çalıştığı düşünülürse ve Windows'un da performans olarak Linux dağıtımlarından daha düşük hızda çalıştığı gözlemlenirse, OpenGL'in daha performanslı çalıştığı görülebilir (Kolar, 2008).

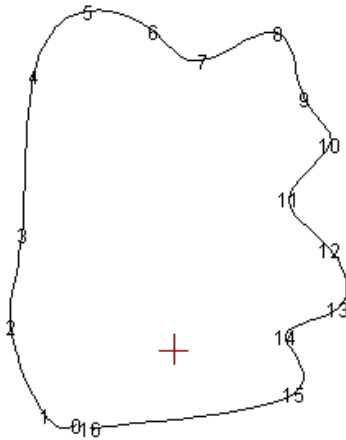
### **3.1.4 Tercih edilen grafik kütüphanesi**

Mono ile yazılan bir C# programının Linux dağıtımlarında da rahatlıkla çalıştırılabilmesi nedeni ile OpenGL tercih edilmiştir. Ayrıca, geliştirme sürecine

başlanmadan önce genel olarak örnek kodlar, dökümantasyonlar araştırılmış ve OpenGL ile ilgili olarak daha fazla bilgiye ulaşılması tercih edilmesinde büyük bir etmendir.

### 3.2 Kesit Fotoğraflarının Birleştirilmesi

Bu çalışmada geliştirilen yazılım, temel olarak fotoğrafla haritalama yöntemine dayanmaktadır. Mağara araştırması süresince çekilen fotoğrafların kullanılabilmesi için, resim işleme teknikleri kullanılmaktadır. Resim işleme teknikleri olarak paralel izdüşüm dönüşümü, rotasyon, resim iyileştirme yöntemi ve gölge filtresi sırasıyla uygulanmıştır. Her bir fotoğrafa uygulanan bu tekniklerden sonra elde edilen resimler, sınır noktalarından ve ışık kaynağının bulunduğu noktayı temsil eden referans noktalardan oluşmaktadır. Bir kesite ait referans nokta, fotoğraf makinesinin bulunduğu konum olarak kabul edildiğinde (Şekil 3.1), kesiti oluşturan bölgenin yapısı iki boyutlu uzayda noktalar ile ifade edilebilmekte ve “cubic spline” adı verilen eğriler ile birleştirilebilmektedir. Şekil 3.1’de kesiti oluşturan noktalar rakamlarla gösterilmiş ve “cubic spline” eğrisi ile birleştirilmiştir.



Şekil 3.1 Bir kesite ait noktalar ve birleştirilen “cubic spline”

Arařtırmacı, bir kesite ait sınırı oluřturan noktaları belirleyip o kesitin bařlangıç noktasından ne kadar uzaklıkta olduđu bilgisiyle sakladığında, topladıđı verileri modelleme yaparken kullanabilmelidir. Bu nedenle, kesitleri oluřturan noktalar ile bu kesitler arasındaki mesafelerden yola ıkararak mađara gibi jeolojik lmlerin  boyutlu modellenmesi C# programlama dili ve OpenGL ktphanesi kullanılarak yapılmıřtır.

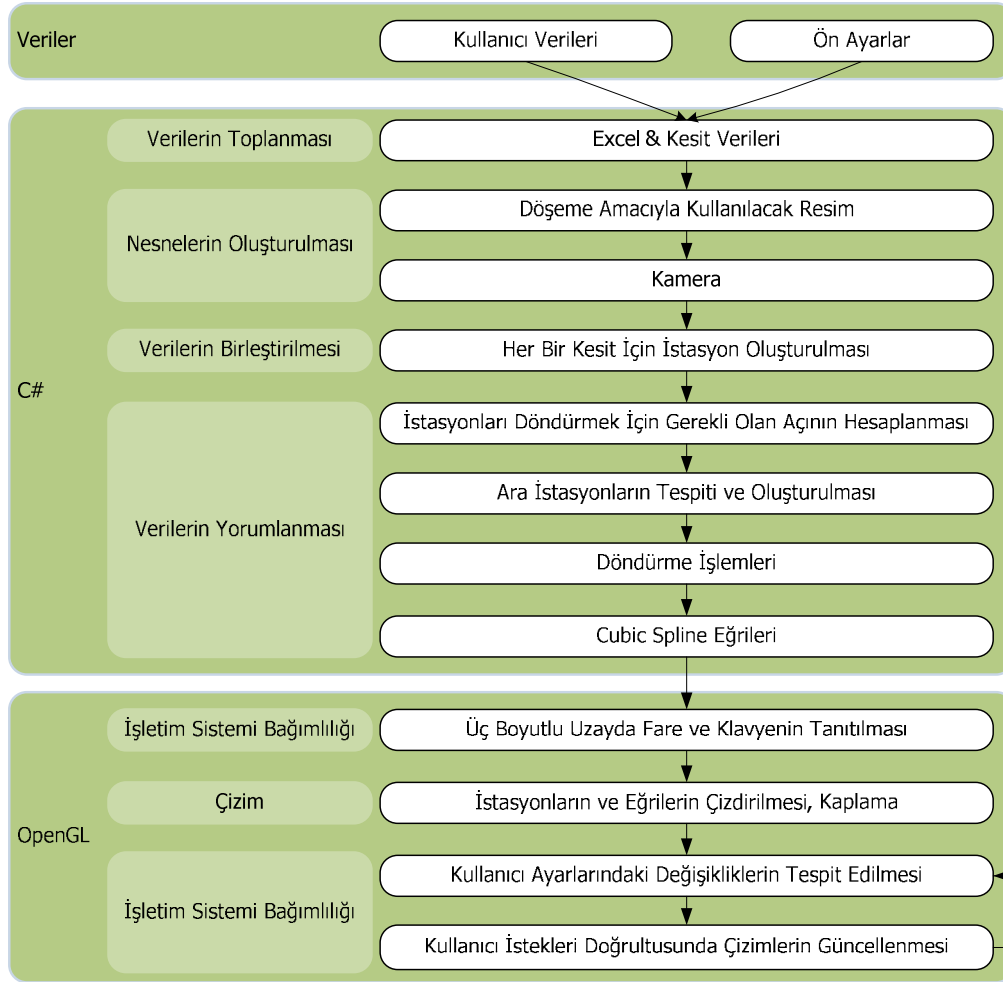
Herhangi bir mađaranın modellenmesi yapılırken kullanıcıdan alınan verilerde yer alan koordinatlar dođrultusunda, her kesit yapılan hesaplamaların ardından OpenGL fonksiyonları ile ekrana izdirilmiřtir. Herhangi bir modellemenin hızlı gerekleřtirilebilmesi iin tm hesaplamalar proje ilk aıldıđında kullanıcıdan istenilen verilerin alınmasının ardından yapılmıř ve bellekte kullanılmak zere saklanmıřtır.

Uygulama geliřtirilirken kullanılan model Őekil 3.3'de gsterildiđi gibi kullanıcıdan veri alınmasıyla bařlayıp girilen verilerin ekranda gsterilebilecek hale getirilmesiyle oluřan bir sreci iermektedir. Bu sre mađara verilerinin kullanıcıdan alınması ile bařlamaktadır. Kullanıcıdan alınan veriler, bir Excel dosyasında sıralı olarak listelenmelidir (Őekil 3.2). Her satırda, iřlem gren istasyona ait referans noktasının  boyutlu uzaydaki konumu ve kesit detay bilgilerinin hangi grnt dosyası ierisinden alınacađı belirtilmelidir.

	A	B	C	D
1	X	Y	Z	Dosya
2	-80,60983	457,16107	-73,52420	5014
3	234,52865	1002,99693	-184,65903	5015
4	209,25348	1291,89340	-184,65903	5016
5	72,12135	1252,57139	-231,01158	5017
6	-403,71512	1125,07139	-363,00929	5018

Őekil 3.2 İstasyonların listelendiđi Excel dosyasına ait ierik





Şekil 3.3 Uygulama içerisinde kullanılan modelin açıklaması

Kesit detay bilgilerinin belirtildiği görüntü dosyasının içerisinde referans olarak alınacak noktanın ana resim üzerinde hangi konumda olduğu bilgisi ile birlikte resim üzerinden alınan noktalar ve her bir noktanın referans noktasından ne kadar uzaklıkta olduğu bilgisi bulunmaktadır. Ayrıca, bu bilgiler resim üzerinde yer alan noktalardan tespit edildiği için bir noktanın gerçekte ne kadar uzunlukta olduğu bilgisi de aynı dosya içerisinde yer almaktadır.

Kesit detay bilgilerinin belirtildiği görüntü dosyası, Excel tablosunda “Dosya” sütunu altında yer alan metnin başına “IMG\_”, sonuna ise “err.txt” eklentileri yapılarak Excel dosyasının bulunduğu konumda aranmaktadır. Dosya içeriğine bakıldığında “referans”, “piksel\_uzunlugu”, “nokta\_sayisi” ve {X=-819,Y=150} gibi satırlar görünmektedir.

Bir istasyona ait görüntü dosyası oluşturulurken fotoğraf makinesinden elde edilen resim ile gerçekte o resmin ne kadar genişlikte bir alanı kapsadığına dair bir oran belirtilmelidir. Bu oran, görüntü dosyasının içerisinde “piksel\_uzunlugu” olarak saklanmaktadır.

“nokta\_sayisi”, fotoğraf makinesindeki resimden elde edilen sınır çizgisinden kaç tane nokta alınarak görüntü dosyasının oluşturulduğunu ifade etmektedir. “referans”, fotoğraf makinesinin resim çekilirken nerede konumlandırıldığını ifade etmektedir. {X=-819,Y=150} gibi veriler, sınır çizgisi üzerinden alınan noktaların referans noktasına göre ne kadar uzaklıkta olduğunu piksel olarak ifade etmektedir.

Kullanıcı verileri alındıktan sonra, çizim aşamasında herhangi bir hesaplama yapılmayacağı için matematiksel hesaplamaları yönlendirecek bir ön ayar dosyasının tanımlanması gerekmiştir. Ön ayarlar ile ilgili dosya içerisinde ara istasyonların oluşturulup oluşturulmayacağı, oluşturulacaksa iki ana istasyon arasında kaç tane ara istasyon oluşturulacağı belirtilmiştir. Ayrıca, istasyon içerisinde nokta tarama modu ve tarama modu varsayılan tarama modu değilse merkez noktası referans nokta olmak şartıyla, merkezi çevreleyen sınır noktaları üzerinde kaç derece aralıklarla tarama yapılacağı tanımlanmıştır.

Ön ayarların tanımlandığı dosya “Ayarlar.ini” dosyasıdır ve programın bulunduğu yerde olmalıdır. Ön ayarlar, kullanıcı verilerinin yüklenmesi ile birlikte yapılan hesaplamaların farklılık göstermesini sağlamaktadır. Örneğin; kullanıcı, istasyonlar arasında yeni sanal istasyonların oluşturulmasını isteyerek, iki istasyonun birbirine bağlanması sırasında kullanılan “cubic spline” eğrilerinin farklı güzergâhları takip etmesini sağlayabilir. Bu sayede, farklı verilerle programın oluşturacağı görsellik

gerçeğe daha da çok yaklaşabilir. Ön ayar dosyası içerisinde “AraIstasyonOlusturulsunMu” parametresi sanal istasyonların oluşturulup oluşturulmayacağını belirtirken “AraIstasyonSayisi” ise iki istasyon arasında kaç sanal istasyonun oluşturulacağını belirtir.

Kesitler arası oluşturulan cubic spline eğrileri ile kaplama yapılmaktadır. “Cubic spline” eğrilerini oluşturacak noktaların seçimi ile ilgili çeşitli algoritmaların geliştirilmesi elde edilecek görüntünün kalitesini etkileyebileceği için yeni bir algoritma desteğinin projeye dâhil edilmesi düşünülmüştür. Bu nedenle ön ayarlar dosyası içerisinde nokta tarama modu ve tarama modu varsayılan tarama modu değilse kaç derece aralıklarla tarama yapılacağı tanımlanmıştır. Kullanıcı “IstasyonNoktaTaramaModu” olarak “0” seçtiğinde yazılım içerisinde tanımlanmış yapı ile görsellik oluşturulacaktır. Eğer kullanıcı tarama modu olarak “1” seçmişse ve kodlama ile yeni bir tarama modu eklenmişse (ki bu eklenti, sınırı oluşturan noktalar ile referans noktasının yapacağı açı ile ilgili olmalıdır) tarama açısı belirleyebilecektir. Varsayılan tarama modu, kullanıcıdan alınan verilerden elde edilen noktaların referans noktasına olan açısının karşılaştırılarak sıralanmasını içermektedir.

### **3.3 Nesnelerin Oluşturulması**

Nesnelerin oluşturulması süreci iki kısımdan oluşmaktadır. Birinci bölüm kaplanacak dokunun oluşturulmasını, ikinci bölüm ise kameranın nasıl kullanılacağını içermektedir.

#### **3.3.1 Doku kaplama**

Üç boyutlu uzayda çizilen nesnelere estetik bir anlam katmak için iki boyutlu resimler kullanılarak döşeme yapılmıştır. Bu iki boyutlu resimlerin kullanılmasına “doku kaplama” denilir. Heckbert’e (1989) göre doku kaplama, resim birleştirmesi için kullanılan bir tür gölgeleme tekniğidir. Duvara yapıştırılan bir duvar kâğıdı gibi üç

boyutlu bir sahnenin üzerine uygulanan doku resmi bu tekniğin alt yapısını oluşturur. Eğer bir masa modelleniyor olsaydı, masa üstü için dörtgenel bir kutu ve masa bacakları için dört silindirik yapının kullanılması gerekirdi. Bu model bilgisayar ortamına taşındığında gerçeği ifade etmeyen bir görüntü ile karşılaşılırdı. Görüntüyü gerçeğe benzetebilmek için günlerce uğraşarak boyama algoritmaları oluşturulabilirdi. Günlerce uğraşmak yerine bir tahta parçasının resmi çekilip bilgisayar ortamında modelin üzerine birbirini tekrarlayan yapıda yapıştırılırdı elde edilen görüntünün uzun emekler harcamadan gerçekçi bir ifadeye büründüğü gözlemlenirdi. Bu nedenle, geliştirilen yazılım içerisinde mağaranın dış duvarlarını oluşturabilmek için doku kaplama yani döşeme yapılmıştır.

Hill'e (2001) göre döşeme yapılması amacıyla kullanılacak resim parçacıklarının OpenGL fonksiyonlarına giriş yapılmadan önce yüklenmesi gerekmektedir. Yüklenecek döşeme resimleri "BMP" formatında olması ve yüklerken de her bir piksel değeri için 24 bit ve her bir renk için 8 bitlik parçalar halinde sisteme tanıtılması gerekmektedir. Bir resmin döşeme olarak yüklenmeden önce OpenGL fonksiyonlarından "glEnable" fonksiyonu ile iki boyutlu döşeme özelliği aktif hale getirilmiş ve resim yükleme işlemi tamamlandıktan sonra aynı özellik pasif hale döndürülmüştür. Resmin OpenGL tarafından daha rahat kullanımı için Hill'e (2001) göre "GL\_TEXTURE\_MIN\_FILTER" ve "GL\_TEXTURE\_MAG\_FILTER" özellikleri lineer, "GL\_TEXTURE\_WRAP\_S" ve "GL\_TEXTURE\_WRAP\_T" özellikleri tekrarlayan olarak belirtilmesi gerekmektedir. "GL\_TEXTURE\_MIN\_FILTER" ve "GL\_TEXTURE\_MAG\_FILTER" özellikleri bir resmin döşeneceği alanın resmin boyutlarından farklı olması sırasında yapılacak dönüşümü belirtmektedir. "GL\_TEXTURE\_WRAP\_S" ve "GL\_TEXTURE\_WRAP\_T" özellikleri ise resmin döşeneceği alan, resmin boyutlarından farklı ise döşemenin tekrarlayan bir yapıda olup olmayacağını göstermektedir. Resmin yüklenebilmesi için, öncelikle resmin boyutlarında çizim nesnesi üzerinden bir dikdörtgen oluşturulmuştur. Ardından resim dikey olarak döndürüldüğünde rotasyon yapıp yapılmayacağı belirtilmiştir. Hill'e (2001) göre rotasyon yapılması döşeme yapıldıktan sonra kullanıcı tarafından farklılıkların gözle görülmesini sağlamaktadır. Bu nedenle, rotasyon yapılmaması göze hoş bir görünüm elde edilmesine neden olmaktadır. Resmin yüklenebilmesi için

“Imaging.BitmapData” sınıfından bir deęişkenin resim boyutlarını kapsayabilecek bir alanı bellekte saklaması ve başka bir programın aynı adresteki belleęe ulaşmasını engellemek gerekmektedir. Bu nedenle 24 bit RGB formatında bir resim alanı oluşturulup karşılık gelen bitler kilitlemiştir. Son işlem olarak OpenGL fonksiyonlarından “glTexImage2D” fonksiyonu ile RGB formatında iki boyutlu döşeme resmi için ayrılan deęişkene yüklenmesi sağlanmıştır.

### 3.3.2 Kamera

Kullanılacak döşeme resimlerinin yüklenmesinin ardından OpenGL ilk kullanıma hazırlanmalıdır. Hill’e (2001) göre kullanıcının gözüne hoş görünebilecek bir gölge modu yumuşak geçişli ve ekranın temizlenmesi için atanan renk siyah olmalıdır. Ardından, son kullanıcıya sanal gerçekliği sağlamak için kamera ile ilgili bir sınıf oluşturulup rastgele seçilmiş bir başlangıç noktasına konumlandırılmıştır. Kamera sınıfı içerisinde işlemleri kolaylaştırmak için koordinatları tutabilecek bir vektör sınıfı ilerde doğabilecek kodlama karmaşasını önlemek amacıyla oluşturulmuştur.

Kameranın bulunduğu lokasyonu deęiştirirken son kullanıcı ileri-geri gidebilir, sola-saęa veya yukarıya-aşāğıya doğru bakabilir. Bu nedenle kamera sınıfı içerisinde dönüşler ile ilgili işlemleri yapabilecek fonksiyonlar tanımlanmıştır. Hill’e (2001) göre kamera nesnesi üç parametreden oluşmaktadır. Bu parametrelerden kameranın bulunduğu konum ile kameranın baktığı konumu belirtecek koordinatların tanımlanması kullanıcının hareketlerini sağlamasına yardımcı olmaktadır.

Herhangi bir noktada kullanıcı herhangi bir yönde ileri veya geri gitmek isterse kameranın baktığı konumu belirten koordinatlardan, bulunduğu konumu belirten koordinatların çıkartılması ve hızı belirten bir deęişkenle oluşan farkların çarpılması ne kadarlık bir hareketin olacağını belirtmektedir. Kameranın durduğu ve baktığı koordinatlara bu fark eklenerek, kullanıcıda ileri veya geri gitmiş hissi oluşturulmuştur.

Kullanıcının sola veya sağa doğru dönmesi ise kameranın baktığı konumu değiştirmekle çözülebilir. Bu nedenle, yapılan testlerde bakış koordinatlarından x ve z bileşenlerinin dönüş hareketlerinde etken olduğu saptanmıştır. Bakılan konumu değiştirebilmek için kameranın bulunduğu konum ile x ve z bileşenleri ayrı ayrı işlemlerden geçirilmiştir. Z bileşenini bulabilmek için öncelikle hız değişkeninin sinüs değeri ile aradaki fark vektörünün x bileşeni çarpılır. Ardından hız değişkeninin kosinüsü ile fark vektörünün z bileşeni çarpılır. İki çarpım sonucu ile kameranın bulunduğu pozisyonun z bileşeninin toplanması, derinlik ile ilgili yeni bakış konumunun z bileşenini vermektedir. X bileşenini bulabilmek için öncelikle hız değişkeninin kosinüs değeri ile aradaki fark vektörünün x bileşeni çarpılır. Çarpım sonucu ile kameranın bulunduğu pozisyonun z bileşeni toplanır. Ardından hız değişkeninin sinüsü ile fark vektörünün z bileşeni çarpılıp bir önceki toplamdan çıkartılarak yatay eksen üzerindeki yeni bakış konumunun x bileşeni elde edilir.

Kullanıcı eğer yukarıya veya aşağıya doğru bakmak isterse, kameranın bakmış olduğu konumun y bileşeni değiştirilerek yeni konum bulunabilir. Bu amaçla, bakış ile bulunduğu konum arasındaki fark vektörünün z bileşeni ile hız değişkeninin sinüsü çarpıldıktan sonra fark vektörünün y bileşeni hız değişkeninin kosinüsü ile çarpılması işleminden elde edilen sonuçtan çıkartılıp kameranın bulunduğu konumun y bileşeni ile toplandığında yeni bakış konumunun y bileşeni elde edilmiş olunur.

### 3.4 Verilerin Birleştirilmesi

Döşeme ve kamera ile ilgili işlemlerin yapılmasından sonra ön ayar dosyası içerisindeki veriler ile kullanıcı verileri sisteme yüklenmiştir. Mağaranın istasyon adı verilen kesit özelliklerini belirten kullanıcı verilerinin, etkin kullanılabilmesi için istasyon adında bir sınıf oluşturulmuş ve o kesit ile ilgili hesaplanan verilerin aynı sınıfa bağlı ek sınıflarda tutulması sağlanmıştır. İstasyon adındaki bu sınıf her bir kesit için tekrar oluşturulup dizi üzerinde saklanmıştır. Böylece tek bir erişim noktası üzerinden tüm yapıya ulaşmak çizim esnasında oluşabilecek performans kayıplarını önlemiştir. Excel dosyası içerisinde yer alan kullanıcı verileri, ana veri olması nedeni ile yazılıma

ilk etapta aktarılmıştır. Bu nedenle, istasyon sınıfında kesitin referans noktasının konumu ve hangi dosyada kesit detaylarının bulunduğu bilgisi değişkenler üzerinde tutulmuştur. Ardından tüm kesit detayları dizi üzerinde yer alan istasyonlara aktarılmıştır.

Verilerin toplanması iki farklı aşamayı içermektedir. Birinci aşama ön ayarların okunması ile başlamaktadır. Ön ayarlarla ilgili olası parametreler “Ayarlar” adındaki bir C# sınıfında tutulmaktadır. Bu sınıf hem ön ayarları hem de programın çalışma anındaki ayarlarının değişikliklerini tutup gösterme anındaki yapılan işlemleri yönlendirir.

İkinci sınıf ise Excel içerisinde yer alan istasyon koordinatları ile başlamaktadır. Bu aşamada, “İstasyon” adındaki bir sınıf ile her bir kesite ait bulunduğu konum ile ilgili ana veriler yüklenmektedir. Ardından, her bir istasyonun detaylarının tanımlandığı dosyalardaki veriler “Image” sınıfı ile “İstasyon” sınıfındaki ilgili dizilere aktarılmaktadır.

“İstasyon” sınıfı proje kapsamında yapılan tüm aritmetik işlemlerin sonuçları ile birlikte her bir verinin değişiklik görmemiş halini “arraylist” adı verilen dizilerde saklamaktadır.

### **3.5 Verilerin Yorumlanması**

Verilerin yorumlanması süreci istasyonları döndürmek için gerekli olan açının hesaplanması, ara istasyonların tespiti ve oluşturulması, döndürme işlemleri ve “cubic spline” eğrilerini içermektedir.

### 3.5.1 Döndürmek için gerekli olan açının hesaplanması

Bir istasyondan, kendisinden sonra gelen istasyona nişan alınarak ölçüm yapıldığı için çizim sırasında oluşturulacak istasyonların da hesaplanan bir açı ile kendi eksenini etrafında  $y$  eksenine olan paralelliği korunacak şekilde döndürülmesi gerekmektedir. Bu açının hesaplanması, tüm istasyonlar üç boyutlu düzlemde ekranda gösterilirken üst taraftan bakıldığında yapılan ölçümler ile çizilen grafiklerin birbirine uyuşmasını sağlamaktadır. Kesiti kendi eksenini etrafında döndürebilmek için kesitin referans noktası merkezi olarak kabul edilmiştir. Ardından, referans noktalar arasındaki açının hesaplanması için, bir sonraki istasyonun bir önceki istasyona göre  $z$  ve  $x$  bileşenlerinin farkları alınarak ters tanjant fonksiyonuna verilip elde edilen değerden 90 derece çıkartılmıştır. İki referans nokta arasındaki döndürme açısı hesaplandıktan sonra hesaplamada ilk kullanılan kesite ait sınır noktaları, elde edilen açı uygulanarak döndürülmüştür. Ancak, yapılacak işlemleri tekrarlamamak amacıyla ve ön ayar dosyası içerisinde ara istasyonların oluşturulması istenilebileceği için döndürme açısı istasyon sınıfı içerisinde saklanmıştır.

### 3.5.2 Ara istasyonlar

Kullanıcı ön ayarları içerisinde ara istasyonların oluşturulması istenilmişse, ara istasyonlar interpolasyon yöntemi ile oluşturulabilir. İnterpolasyon belirli noktalardaki belirli değerleri alan bir fonksiyonun tanımlanması işlemidir (Kahaner vd., 1989). Daha açıklayıcı olmak gerekirse,  $(x_1, y_1, z_1)$  ve  $(x_2, y_2, z_2)$  noktalarından geçen ve  $x_1$ 'in  $x_2$ 'ye eşit olmadığı varsayılan, tekil bir birinci dereceden polinom oluşturulabilir. Bu ifadeden yola çıkılarak, iki istasyonun sıralanmış kesit eğrisi noktaları içerisinde iki nokta alınıp ilgili polinom tanımlanırsa, bu noktalar arasında oluşturulmak istenilen ara istasyon sayısı kadar yeni koordinat kümeleri elde edilir. Elde edilen koordinat kümeleri istasyon sınıfından türetilmiş yeni bir değişkende toplanıp referans olarak alınan iki istasyondan işlemlerde ilk kullanılan istasyon bilgileri ile birleştirilerek saklanılabilir. Örnek vermek gerekirse, kullanıcı verileri ile oluşturulan ana istasyonlar için döndürme açıları hesaplanıp aynı döndürme açıları ara istasyonlar için de



kullanılmıştır. Böylece işlem tekrarı ve performans düşüklüğü yaşanması engellenmiştir. Burada kesit eğrisi üzerindeki noktaların sıralanması referans noktası ile aralarındaki açığa göre yaptırılmıştır. Eğer iki noktanın açısı da aynı ise referans nokta ile aralarındaki mesafeye bakılarak sıralandırılmışlardır.

Temel işlemler bittikten sonra her bir istasyon için çizim aşamasında kullanılan yapılar hesaplanır. Bu yapılar sırasıyla döndürülmemiş istasyona ait taslak yapı, döndürülmüş istasyona ait taslak yapı, döndürülmemiş istasyona ait detaylı yapı ve döndürülmüş istasyona ait detaylı yapıdan oluşmaktadır. Bu yapıları son kullanıcı görememektedir, ancak yazılımı geliştirirken yapılan işlemlerin doğrulunun kontrol edilmesi açısından önemlidir.

### 3.5.3 İstasyonların döndürülmesi

Ara istasyonların oluşturulup oluşturulmayacağı doğrultusunda gerekli işlemlerin yapılmasının ardından, elde edilen veriler işlenerek birleştirilmiştir. Bu nedenle, öncelikle istasyonların döndürülmesi işlemi yapılmış, ardından istasyonlar arasındaki sınır noktalarını birleştirecek “cubic spline” eğrileri hesaplanmıştır.

İstasyonların kendi eksenleri çevresinde döndürülmesi işlemi Denklem 3.1’de gösterilen matris çarpımı üzerinden yapılabilir. Burada bir istasyonun kendi ekseninin referans noktasıyla kesiştiği varsayılmıştır. Denklem 3.1’de gösterilen matris çarpımı, Hoff’a (2000) göre sağ el kuralı ile saat yönünün tersine, y eksenini çevresinde döndürmek için kullanılmaktadır. Bu matris üzerinde kullanılan sinüs bileşenlerinin işaretlerinin yerleri değiştirildiğinde saat yönünde döndürme işlemi yapılabilmektedir. OpenGL fonksiyonları incelendiğinde döndürme işlemlerini yapan komutların olduğu görülmüştür. Ancak, döndürme işlemi yapıldıktan sonra yeni koordinatlar direkt olarak elde edilememiştir. Bunun yerine geri besleme mekanizması kullanılarak yeni koordinatların elde edilebileceği tespit edilmiştir. Yapılan testlerde geri besleme mekanizmasının oluşturulması, döndürme işleminin sıklığından ötürü ciddi performans

kaybına neden olduğu gözlemlenmiştir. Bu nedenle, döndürme işlemi OpenGL'e bırakılmayıp Denklem 3.1'de yer alan matris çarpımı üzerinden yaptırılmıştır.

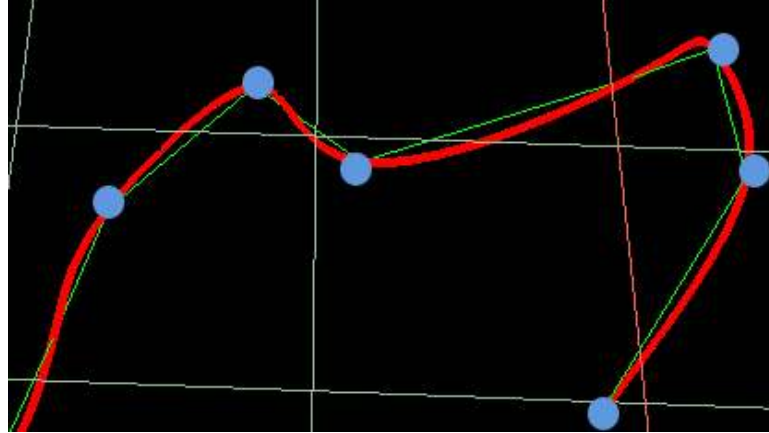
$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix} * \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (3.1)$$

Denklem 3.1'de gösterilen matris çarpımı incelendiğinde,  $\theta$  radyan cinsinden döndürme açısı,  $x$ ,  $y$ ,  $z$  döndürülmek istenilen koordinatı ve  $x'$ ,  $y'$ ,  $z'$  döndürülmüş koordinatı belirtmektedir. Bir kesite ait koordinatlar direk Denklem 3.1'deki matris çarpımında kullanıldığında elde edilen yeni koordinatlar kesitin  $y$  eksenine çevresinde döndürülmüş halidir. Ancak, döndürme işlemi kesitin referans noktası üzerinde yapılması gerektiği için öncelikle referans noktası  $(0, 0, 0)$  noktasını gösterecek şekilde taşınmıştır. Taşınma işlemi sırasında hesaplanan konumlar arası mesafe kesiti oluşturan tüm koordinatlara da aynı oranda uygulanmıştır. Ardından, döndürme işlemi için matris kullanılıp tüm koordinatlar başlangıç referans noktasına göre geri konumlandırılmıştır. Böylece bir kesitin kendi ekseninde döndürülmesi işlemi tamamlanmıştır.

### 3.5.4 “Cubic spline” eğrileri

Döndürme işlemleri yapıldıktan sonra matematiksel hesaplamalar arasında son bir adım olarak kesitleri birleştiren eğrilerin yani “cubic spline” eğrilerinin oluşturulması kalmaktadır. “Cubic spline” eğrisi, belirli sayıdaki noktalar üzerinden geçen, basit formüller kullanılarak parça parça oluşturulmuş eğrilerdir. Daha detaylı bakmak gerekirse,  $x_0 < x_1 < \dots < x_N$  varsayımı doğrultusunda  $(x_0, f_0), \dots, (x_N, f_N)$ 'nin düzlemde noktaları ifade eden bir set olduğu düşünülürse,  $p_i(x_i) = f_i, p_i(x_{i-1}) = f_{i-1}, p'_i(x_i) = p'_{i+1}(x_i), p''_i(x_i) = p''_{i+1}(x_i)$  kurallarını sağlayan  $p_1(x), \dots, p_n(x)$  polinomlarına “cubic spline” eğrisi denilmektedir.

“Cubic spline” eğrileri, Şekil 3.4’te kırmızı renkte gösterildiği gibi üçüncü dereceden polinomlardan tek tek kavisli eğriler kullanılarak oluşturulmuş eğri demetleridir. Bu eğri demetini oluşturan polinomlar  $m$  tane kontrol noktasından geçmektedir. Her bir polinomun ikinci türevi genellikle bitiş noktalarında sıfıra eşitlenmektedir. Bu nedenle  $m - 2$  denklemin oluşturduğu sistem sınırların oluşturulmasını sağlamaktadır. Doğal “cubic spline” adı verilen bu sistem basit bir üçlü diyagonal matrisin oluşmasına neden olur ve bu sistem polinomların katsayılarının girilmesi ile basit bir şekilde çözülebilir (Weisstein, 2009). Şekil 3.4’de de görüldüğü üzere altı istasyonu birbirine bağlayan yeşil renkteki çizgiler modelleme yaparken keskin dönüşler oluşturmaktadır. Bu nedenle, göze hoş görünen görüntüler elde edilmek istendiği için “cubic spline” eğrileri kullanılmıştır.



Şekil 3.4 Cubic spline eğrisi ve istasyonlar ile ilişkisi

“Cubic spline” eğrisini oluşturan polinom Denklem 3.2’de verilmiştir (Kahaner vd., 1989). Bu formülde kullanılan ilk iki katsayı direkt bulunabilmektedir. Ancak formülün geri kalanında kullanılan son iki katsayıyı bulabilmek için Denklem 3.5 ve Denklem 3.6’da gösterildiği gibi birinci dereceden türevlerinin alınması gerekmektedir.

$$Y_i(u) = a_i + b_i u + c_i u^2 + d_i u^3 \quad (3.2)$$

$$Y_i(0) = y_i = a_i \quad (3.3)$$

$$Y_i(1) = y_{i+1} = a_i + b_i + c_i + d_i \quad (3.4)$$

$$Y_i^{(1)}(0) = D_i = b_i \quad (3.5)$$

$$Y_i^{(1)}(1) = D_{i+1} = b_i + 2c_i + 3d_i \quad (3.6)$$

Denklem 3.2, Denklem 3.3, Denklem 3.4, Denklem 3.5 ve Denklem 3.6'da gösterilen denklemler çözüldüğünde Denklem 3.7, Denklem 3.8, Denklem 3.9, ve Denklem 3.10'da gösterilen denklemler elde edilmektedir.

$$a_i = y_i \quad (3.7)$$

$$b_i = D_i \quad (3.8)$$

$$c_i = 3(y_{i+1} - y_i) - 2D_i - D_{i+1} \quad (3.9)$$

$$d_i = 2(y_i - y_{i+1}) + D_i + D_{i+1} \quad (3.10)$$

Yukarıdaki formüllerin çözümlenmesinin yerine “cubic spline” eğrisi daha basit bir yöntemle ifade edilebilir (Kahaner vd., 1989). Bunun için, yazarlara göre doğal “cubic spline” eğrisi aslında “Hermitive” interpolasyonunun özel bir durumu olduğu belirtilmektedir. Bu nedenle Denklem 3.11, Denklem 3.12, Denklem 3.13 ve Denklem 3.14'te gösterilen denklemlerin kullanılabileceği belirtilmektedir.

$$Y_{i-1}^{(2)}(1) = Y_i^{(2)}(0) \quad (3.11)$$

$$2c_{i-1} + 6d_{i-1} = 2c_i \quad (3.12)$$

$$2[2(y_i - y_{i-1}) - 2D_{i-1} - D_i] + 6[2(y_{i-1} - y_i) + D_{i-1} + D_i] \quad (3.13)$$

$$= 2[3(y_{i+1} - y_i) - 2D_i - D_{i+1}]$$

$$D_{i-1} + 4D_i + D_{i+1} = 3(y_{i+1} - y_{i-1}) \quad (3.14)$$

Elde edilen denklemlerin kısa bir biçimde ifade edilebilmesi matris üzerinde gösterilebilir. Matris üzerinde gösterim Denklem 3.15’de belirtilmiştir. Matris üzerindeki  $y$  değişkenleri “cubic spline” eğrisi oluşturulacak noktalardan oluşan diziyi ifade etmektedir. Burada dikkat edilmesi gereken nokta, kesitlerdeki sınır noktalarını oluşturan koordinatların sayısının birbirinden farklı olabileceğidir. Bu nedenle, tüm istasyonlardaki koordinat sayıları tespit edilmeli ve elde edilen koordinat sayılarından en düşüğü kadar dizi oluşturulmalıdır. Oluşturulan diziler içerisinde bir dizi bir istasyonun sınır noktalarından sadece bir tane koordinatı içermelidir. Eğer, bir dizi birden fazla noktayı içerirse, elde edilen OpenGL görüntüsündeki döşemeler birbirlerini kesecek ve son kullanıcının gözüne hoş görünmeyen şekiller elde edilmiş olacaktır.

$$\begin{bmatrix} 2 & 1 & 0 & \cdot & \cdot & \cdot & 0 \\ 1 & 4 & 1 & \cdot & \cdot & \cdot & 0 \\ 0 & 1 & 4 & 1 & \cdot & \cdot & 0 \\ 0 & 0 & 1 & 4 & 1 & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdot & \cdot & 1 & 4 & 1 \\ 0 & 0 & \cdot & \cdot & \cdot & 1 & 2 \end{bmatrix} \begin{bmatrix} D_0 \\ D_1 \\ D_2 \\ \cdot \\ \cdot \\ \cdot \\ D_m \end{bmatrix} = \begin{bmatrix} 3(y_1 - y_0) \\ 3(y_2 - y_0) \\ \cdot \\ \cdot \\ \cdot \\ 3(y_m - y_{m-2}) \\ 3(y_m - y_{m-1}) \end{bmatrix} \quad (3.15)$$

Yazılım içerisinde “cubic spline” eğrileri hem sınır noktaları hem de referans noktaları içerecek şekilde planlanmıştır. Kullanıcı mağaranın içerisini gezmek isterse sınır noktaları üzerinden oluşturulan eğri demetleri alt ve üst komşuları ile dörtgenler aracılığıyla birleştirilmiş ve OpenGL içerisinde tanımlanan fonksiyonların yapısı nedeni ile dörtgenlerin üzeri döşeme kaplandığında hoş görünebilecek bir renk ile boyanmıştır. Birleştirilen bu dörtgenler OpenGL fonksiyonları kullanılarak mağaranın iç yapısına uygun olabilecek bir döşeme ile kaplanmıştır.

Kullanıcının gerçekleştirebileceği hareketler arasında ileriye ve geriye doğru gittiği düşünülürse, daha rahat dolaşabilmesi için her bir istasyonun referans noktalarını birleştiren “cubic spline” eğrisi üzerinde hareket edebilmesi sağlanmıştır.

“Cubic spline” eğri demetlerinin oluşturulabilmesi için Denklem 3.15’de gösterilen matrisi tanımlayıp kullanan “CubicSpline.cs”, “calcCubicSpline.cs” ve “cubic.cs” adında üç farklı sınıf oluşturulmuştur. “cubic.cs” sınıfı Denklem 3.2’de gösterilen  $Y_i(u) = a_i + b_i u + c_i u^2 + d_i u^3$  formülünü tanıtmak için kullanılmaktadır. Bu sınıf, “CubicSpline.cs” tarafından yazılımcı isteğine bağlı aralıklarla “calcCubicSpline.cs” sınıfı üzerinden elde edilen değerlerden örnekler alınmasını sağlamaktadır. “calcCubicSpline.cs” sınıfı Denklem 3.15’de gösterilen matrisin yazılım içerisinde tanımlanması için kullanılmıştır. Sınıf ile aynı isimdeki bir fonksiyon ile giriş değeri olarak verilen noktaları içeren eğrinin tüm koordinatlarını hesaplayıp geri dönüş değeri olarak “cubic.cs” sınıfındaki “cubic” isminde bir diziyi doldurmaktadır. “CubicSpline.cs” sınıfı istasyonlardaki referans noktalarını birleştiren eğri ile sınır noktalarını tek tek birleştiren eğrileri oluşturacak noktaları dizi içerisinde tutup, calcCubicSpline fonksiyonunu çağırarak geri dönen “cubic” tipindeki değerlerden belirli aralıklarla örnekler alarak “arraylist”lerde saklamaktadır.

### 3.6 Kullanıcı Ayarlarındaki Değişikliklerin Tespit Edilmesi

Tezin kodlama ile ilgili bölümünde C# programlama dili ile birlikte OpenGL kütüphanesini kullanabilmek için “Tao Framework” adı altında bir altyapı kullanılmıştır. Ridge’e (2008) göre Tao yapısı farklı platform aynı kod mantığında geliştirilen medya uygulamalarında .Net ve Mono platformlarını faydalı hale getirmek için kullanılan teknikler topluluğudur. Bu altyapı temel OpenGL kütüphanesinin tüm özelliklerini barındırmasına ek olarak işletim sistemine bağlı olarak geliştirilen programın sürekli takip ettiği imleç hareketleri ile birlikte klavye hareketlerini de programdan alıp OpenGL kütüphanesine aktarmaktadır. Ancak, yapılan testler sırasında imleç hareketlerinin tam olarak yakalanamadığı görülmüş ve altyapının sunmuş olduğu özelliklerin kullanılmasındansa klasik bir yöntem olarak program üzerindeki değişikliklerin yine program üzerinde tutulmasına karar verilmiştir.

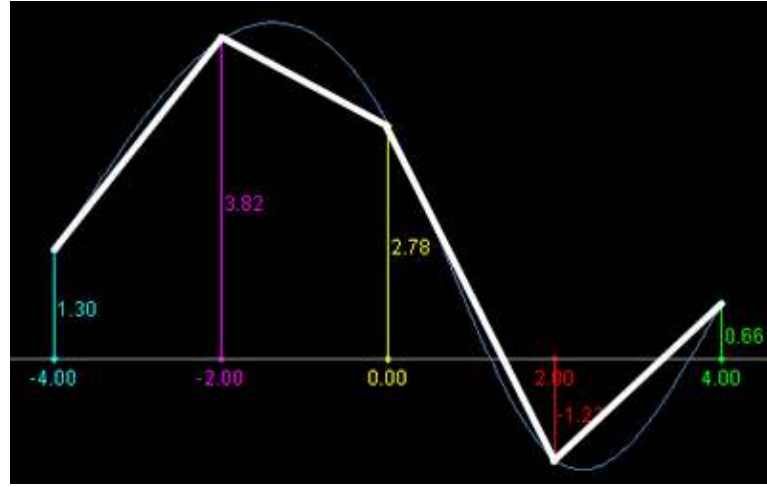
Kullanıcı klavyedeki yön tuşlarına bastığında veya program üzerindeki menülerden değişiklikler yaptığında C# içerisindeki “Ayarlar.cs” sınıfına değişiklikler değişkenlerin değerleri güncellenerek aktarılmaktadır. OpenGL, yapısı gereği her bir saniyede birçok kez görüntüyü yenilediği için, kullanıcının yaptığı değişiklikler yeni bir ekran çizimi sırasında işleme alınıp gösterime yansıtılmaktadır. Bu esnada yapılan kontroller çok az sayıda ve sadece karşılaştırmalardan oluştuğu için performansı çok az etkilemektedir.

Kullanıcının imleç üzerinde yaptığı işlemlerin OpenGL kütüphanesine aktarımı yapılmamıştır. İmleç hareketlerinin görüntüye yansıtılabilmesi için kullanıcının imleci program üzerinde OpenGL gösterim alanına getirmesi ve farenin sol tuşuna basması beklenmektedir. Kullanıcı farenin sol tuşuna bastığında imleç görünmez hale getirilip OpenGL gösterim alanının yatay ve dikey olarak tam ortasına taşınmaktadır. Kullanıcı fareyi hareket ettirdiğinde imleç dikey veya yatay olarak hareket ettikten sonra tekrar gösterim alanının merkezine taşınarak imlecin eski ve yeni koordinatları arasındaki fark hesaplanmıştır. Bu hesaplanan fark görüntüyü hangi yöne ne kadarlık bir açı ile döndürüleceğinin hesaplanmasında kullanılmıştır. Böylece kullanıcı görünmez haldeki imleç hareketini görmeden gerçek hayattaki hareketlerin programa yansıtılması sağlanmıştır.

### **3.7 Hacim ve Uzunluğun Hesaplanması**

“Cubic spline” eğrileri oluşturulduktan sonra mağaranın toplam hacmi ile birlikte uzunluğunun hesaplanması yapılabilmektedir.

Uzunluk hesaplaması referans noktaları birleştiren eğri üzerinde yapılır. Bu nedenle, haritalama ekibi tarafında yapılan ölçümlerdeki toplam uzaklıktan daha fazla ve daha gerçekçi olduğu düşünülmektedir. Uzunluk hesaplaması iki kesitin referans noktalarını birleştiren eğrinin uzunluğu kullanılmaktadır. Şekil 3.5’te görüldüğü gibi haritalama ekibinin yapmış olduğu ölçüm beyaz renkteki çizgi ile ifade edilmektedir. Geliştirilen uygulama içerisindeki ölçüm ise mavi eğri ile belirtilmektedir.

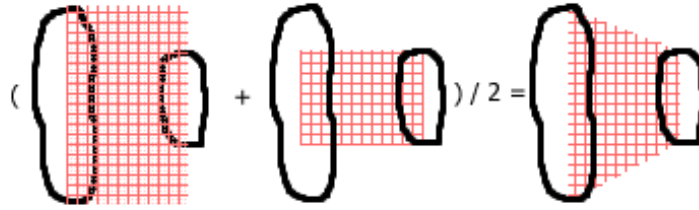


Şekil 3.5 Uzunluk hesaplamasında oluşan farkın gösterimi

Mağaranın toplam hacminin hesaplanabilmesi için de her bir istasyonun referans noktasından geçen “cubic spline” eğrisi kullanılmaktadır. Oluşturulan bir istasyonun ister gerçek bir istasyon isterse sonradan oluşturulan sanal bir istasyon olduğu düşünülürse birinci istasyonun kesit alanının hesaplanmasının ardından “cubic spline” eğrisi üzerinde ikinci istasyona kadar ilerletilir. Birinci istasyonun kesit alanı ile eğri uzunluğu çarpılır. Ardından, aynı eğri üzerinde ikinci istasyondan birinci istasyona doğru ilerletilir ve ikinci istasyonun kesit alanı ile eğri uzunluğu çarpılır. Elde edilen iki hacmin ortalaması alınır (Şekil 3.6). Böylece iki istasyon arasında oluşan hacim yaklaşık olarak bulunur. Bu işlem tüm istasyon ikilileri için yapıp elde edilen sonuç toplandığında mağaranın toplam hacmi elde edilir. Burada önemli olan nokta bir istasyonun kesit alanının hesaplanmasıdır. İstasyonun kesitini oluşturan noktaların oluşturduğu poligonun alanı farklı şekillerde bulunabilir. Bu yöntemlerden en çok bilinen poligonu üçgenlere bölüp her bir üçgenin alanını hesaplayarak poligonun toplam alanının bulunmasını sağlamaktır. Page (2009) ve Simmons'a (2008) göre bir poligonun alanı poligonu oluşturan noktaların birbirini takip edecek şekilde bir determinant üzerinde yazılması ile bulunabilir (Denklem 3.16).



$$A = \frac{1}{2} \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \dots & \dots \\ x_n & y_n \\ x_1 & y_1 \end{vmatrix} \quad (3.16)$$



Şekil 3.6 Hacmin hesaplanması

### 3.8 Çizim

Tüm hesaplamalar yapıldıktan sonra çizim aşamasına geçilmiştir. Dikkat edilmesi gereken nokta, çizim aşamasında yapılacak her türlü hesaplamaların performansı çok fazla düşürebileceğidir. Bu nedenle, çizim aşamasında kullanılacak her türlü veri önceden planlanmış ve hesaplanmıştır. Çizim işlemleri yapıldıktan sonra, kullanıcının istediği hareketleri gerçekleştirirken, çizilen nesnelerin konumları değiştirilebilir. Ancak, bu yöntem işlemciye çok fazla yüküneceği ve yeni hesaplamalar gerektireceği için kaçınılmıştır. Nesnelerin yeni koordinatlarını hesaplamak yerine kameranın bulunduğu ve baktığı konum bilgilerinin güncellenmesi daha hızlı olacağı için tercih edilmiştir. Kullanıcının mağara içerisinde gezinirken, yön duygusunu kaybedebileceği düşünülerek, kullanıcıya bulunduğu konum bilgileri, mağaranın neresinde olduğu ve yönünün nasıl değiştiği bilgileri sunulmuştur. Bu bilgilerden harita ve yön değişimi, OpenGL ekranı parçalara ayrılarak her bir parçanın kendisine özel çizimi yapılmıştır. Referans noktaları birleştiren “cubic spline” eğrisinin küçük bir ekranda çizilmesi ve kameranın tepe noktasına taşınıp eğrinin tam orta noktasına bakması harita oluşturmak amacıyla kullanılmıştır. Ayrıca, yön duygusunu oluşturabilmek için bir başka OpenGL ekranında (0,0,0) koordinatında yön oklarının

izilip kullanıcı hareketleri doęrultusunda bu ekrandaki kamera pozisyonunu deęiştirilmiştir. Bu esnada kameranın baktığı konum (0,0,0) koordinatını göstermektedir.

Her ne kadar hesaplama yapılmasa da programın çalışma anında oluşturulan tüm nesnelere verileri farklı farklı yapılarda ve konumlarda olması görüntüde kesintilere yol açmaktadır. Oluşan kesintilerin önüne geçebilmek için tüm veriler yüksek seviyeli diller arasında çoęunlukla kullanılan “HashMap”, “ArrayList” gibi yapılarda önceden her türlü hesaplaması yapılmış ve ihtiyaç duyulabilecek tüm özellikleri tanımlanmış sınıflarda depolanmıştır. Depolanan bu verilere tek bir nesneyi oluşturarak erişebilmek için programın ilk açılışında ana bir “ArrayList” yapısı oluşturulmuş ve ilerleyen aşamalarda oluşturulan tüm sınıflar bu ana yapıya eklenmiştir. Böylelikle OpenGL ekranında çizdirilen görüntülerin geçişleri arasında hiçbir kesintiyle karşılaşmadığı gibi gösterim performansı yüksek seviyede arttırılmıştır.

Yapılan testlerde oluşturulan on yedi istasyon ve bu istasyonların birbirleri arasındaki bağlantıların sağlanması ile bu bağlantılara yapılan kaplama üzerinden elde edilen performans gözle görülür bir biçimde yüksek olmasına rağmen istasyon sayısı daha da çok arttırıldığında performansı etkileyen ana yapının bilgisayarın işlemcisi ve ram kapasitesinin yüksekliği olduğu ortaya çıkmaktadır. Kısacası istasyon sayısının çok yüksek olduğu durumlarda işlemci ve ram kapasitesinin yüksek olduğu bilgisayarların kullanılması önemli bir etkidir.

## BÖLÜM 4

### UYGULAMANIN KULLANIMI

Üç boyutlu mağara gibi jeolojik ölçümlerin modellenmesi için geliştirilen yazılım kapsamında son kullanıcıya oluşturulan yapı üzerinde sanal gerçeklik adına farklı fiziksel aşamaları yapabilmesini sağlayan özellikler eklenmiştir. Bu özellikler genel olarak ışıklandırma ayarının değiştirilmesi, istasyonlar arasında çizilen doğruların ve “cubic spline” eğrilerinin gösterimi, kaplanan döşeme yapısının gizlenebilme özelliği, mağaranın bir prizma ile kaplanması ve prizma çevresinde dolaşma imkanının tanınması, istasyonlar arasında “cubic spline” eğrisi üzerinde dolaşma imkanı, serbest dolaşım ve imleç üzerinden belirli bir konuma doğru dönebilme gibi işlemleri içermektedir.

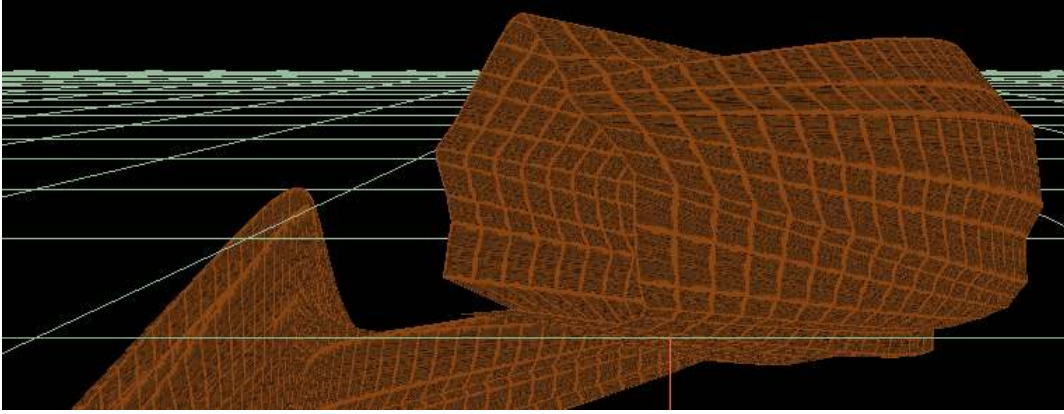
#### 4.1 Serbest Dolaşım Modu

Program açıldığında otomatik olarak serbest dolaşım yapılabilecek bir şekilde kullanıcının karşısına çıkmaktadır (Şekil 4.1). Kullanıcı farenin sol butonuna bastıktan sonra imleci sağa veya sola, yukarı veya aşağıya doğru ilerlettiğinde görmüş olduğu yapının sağına veya soluna dönebilmekte, yukarıya veya aşağıya doğru bakabilmektedir. Bu esnada klavye üzerinde yer alan yön tuşları ile istediği konuma doğru görüş alanını koruyarak ilerleyebilmektedir. Serbest dolaşım modunda ışığın etkisi kapatılmıştır, yani araştırmacı uzaktaki nesnelere yakındaki nesnelere yansıyan ışığı aynı şiddette görecektir.

Serbest dolaşım esnasında klavyedeki yön tuşlarının işlevleri aşağıda listelenmiştir:

- Sol Ok: Sola doğru yatay ilerleyebilme
- Sağ Ok: Sağa doğru yatay ilerleyebilme

- Üst Ok: İleriye doğru dikey gidebilme
- Alt Ok: Geriye doğru dikey gidebilme
- A Tuşu: Sola doğru dönebilme
- D Tuşu: Sağa doğru dönebilme
- W Tuşu: İleriye doğru dikey gidebilme
- S Tuşu: Geriye doğru dikey gidebilme
- Z Tuşu: Aşağıda doğru dönebilme
- X Tuşu: Yukarıya doğru dönebilme

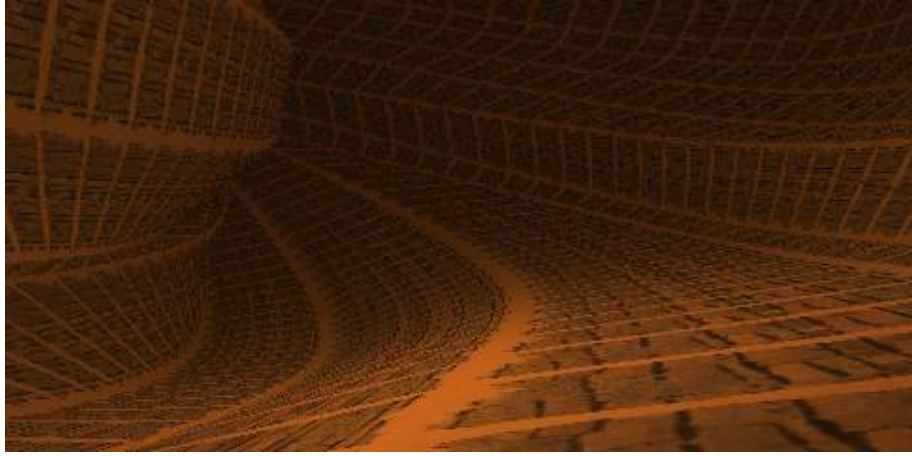


Şekil 4.1 Serbest dolaşım modu

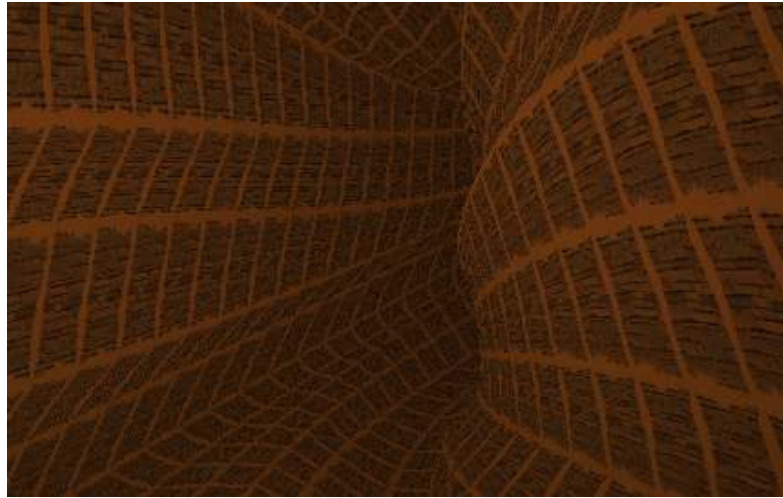
## 4.2 Yol Modu

Programın üst kısmında yer alan menü üzerinden “Görünüm” altında “Yol” yazan bölüm seçildiğinde kullanıcının konumu istasyonlar arası referans noktalarına ait “cubic spline” eğrisinin ilk konumu olan noktaya taşınır (Şekil 4.2 ve Şekil 4.3). Kullanıcı klavye üzerindeki kısa yolları kullanarak eğri üzerinde veya eğrinin dışında hareket edebilir. Buradaki temel amaç kullanıcıya mağara içerisinde dolaşmış hissi verebilmektedir. Yol modunda ışığın etkisi açılmıştır, yani araştırmacı uzaktaki nesnelere ile yakındaki nesnelere yansıyan ışığı farklı şiddette görecektir.

İstasyonlar arasındaki eğri üzerinde dolaşmak için kullanıcı klavyedeki ileri okuna basmalıdır. Farenin sol tuşuna bastıktan sonra istediği bir konuma doğru bakabilir veya bu tarz temel işlemleri klavyedeki kısa yolları kullanarak yapabilir. Kullanıcı bu modda ilerlerken ekranın sol üst tarafında bir önceki konuma göre oluşan eğimi gösterilmektedir.



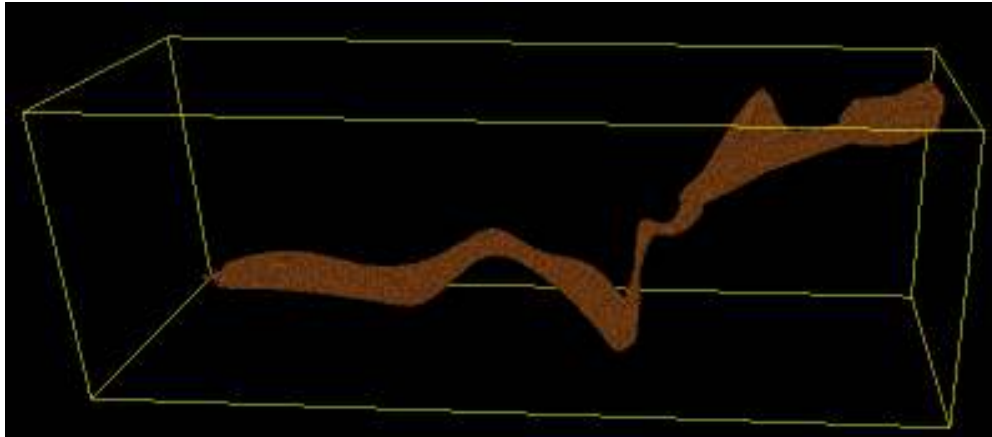
Şekil 4.2 Yol modu



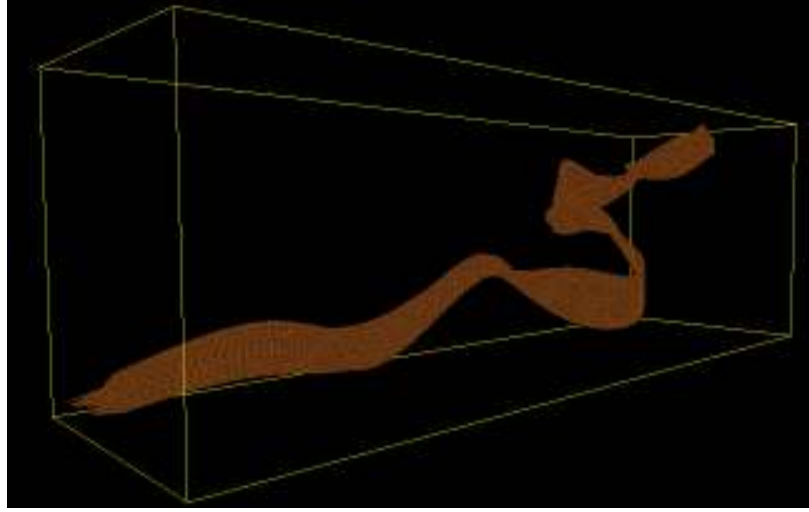
Şekil 4.3 Yol modu ve ışığın etkisi

### 4.3 Prizma Modu

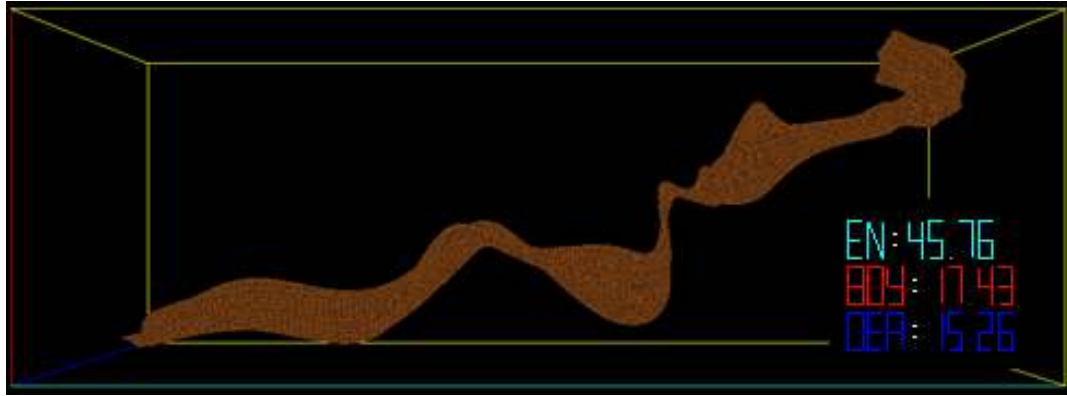
Programın üst kısmında yer alan menü üzerinden “Görünüm” altında “Prizma” yazan bölüm seçildiğinde mağarayı oluşturan tüm istasyonların bir prizma içerisinde görünmesini sağlayan özellik aktif hale gelmektedir. Kullanıcının bakış açısı, prizmanın dışında olacak şekilde konumlandırılmaktadır. Kullanıcı sol – sağ tuşlarını kullanarak, prizma merkezde olacak şekilde prizmanın çevresini saran gizli, yatay bir daire üzerinde hareket edebilmektedir. Kullanıcı yukarı – aşağı tuşlarını kullanarak, prizmanın üzerinde olduğu kesitin çevresini saran gizli, dikey bir daire üzerinde hareket edebilmektedir. Ayrıca, kullanıcı klavye üzerindeki “W”, “A”, ”S” ve ”D” tuşlarını kullanarak herhangi bir yöne doğru hareket edebilmektedir. Prizma modunda ışığın etkisi kapatılmıştır, yani araştırmacı uzaktaki nesnelere yakındaki nesnelere yansıyan ışığı aynı şiddette görecektir. Şekil 4.4’te modellenmiş mağara bir bütün şeklinde karşı taraftan görünmektedir. Şekil 4.5’de kullanıcının sağ-sol veya yukarı-aşağı tuş kombinasyonları ile mağara modelinin çevresinde farklı konumlardaki görüntüler gösterilmektedir. Şekil 4.6’da istasyonları kapsayan prizmanın en, boy ve genişlik bilgileri gösterilmiştir.



Şekil 4.4 İstasyonları kapsayan prizmanın gösterilmesi



Şekil 4.5 İstasyonları kapsayan prizmanın çevresinde dolaşım

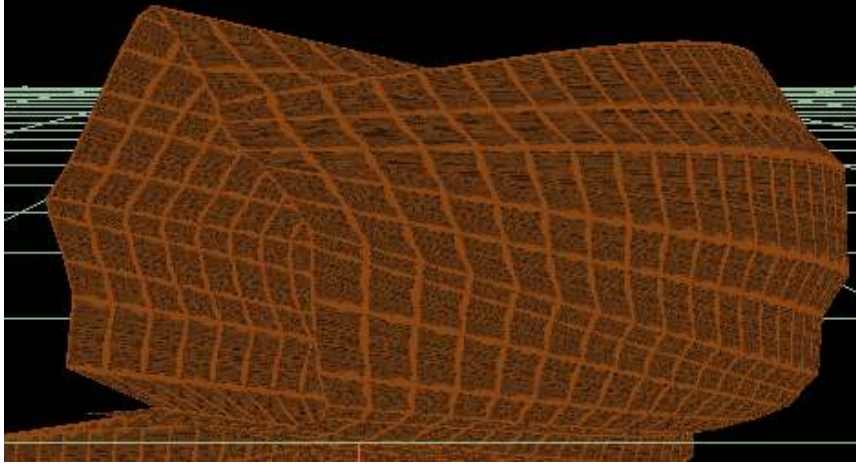


Şekil 4.6 İstasyonları kapsayan prizmanın çevresinde detay bilgileri

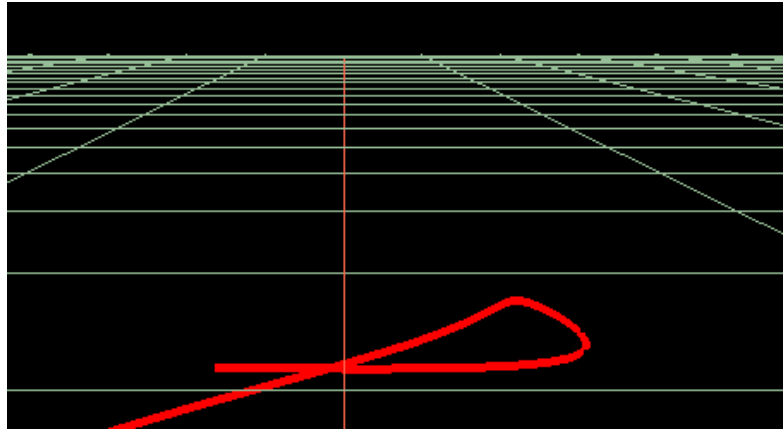
#### 4.4 İstasyonlar Arası Dolgu Kaplama

Programın üst kısmında yer alan menü üzerinden “Görünüm” altında “Duvarlar” yazan bölüm seçildiğinde varsayılan olarak gelen istasyonlar arası sınır noktalarına ait “cubic spline” eğrileri çevresinde kaplanan dolgu gizlenmektedir ve varsayılan olarak gizli gelen istasyonlar arası çizilen ve her bir istasyonun referans noktasından geçen

“cubic spline” eğrileri gösterilmektedir. Şekil 4.7’de mağara modeli kaplanan dolgu ile birlikte gösterilmektedir. Şekil 4.8’de ise kaplanan dolgu silindiğinde araştırmacının göreceği “cubic spline” eğrisi gösterilmiştir. Bu özellik istasyonlar arası “cubic spline” eğrilerinin incelenmesi istenildiğinde kullanıcıya kolaylık sağlamaktadır.



Şekil 4.7 Dolgunun gösterimi aktifken istasyonlar

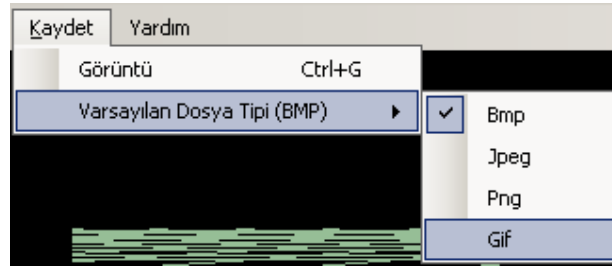


Şekil 4.8 Dolgunun gösterimi pasifken oluşan eğri



#### 4.5 Ekran Görüntüsünün Saklanması

Araştırmacı mağara üzerinde incelemesini yaparken kendisi için önemli olan bir bölgenin ekran görüntüsünü saklamayı isteyebilir. Bu nedenle program içerisinde anlık ekran görüntüsünün yakalanıp Jpeg, Bmp, Gif veya Png formatlarında saklayabilmesi özelliği konulmuştur. Ekran görüntüsünü almak isteyen kullanıcı programın üst kısmında yer alan menü üzerinden Şekil 4.9’da gösterildiği gibi “Kaydet” altında “Varsayılan Dosya Tipi” yazan bölümden hangi formatta görüntünün yakalanmasını istediğini belirtmelidir. Ardından aynı menü üzerinde yer alan “Görüntü” işlem adımından sonra ekran görüntüsünün nereye kayıt edileceği kullanıcıya sorulacaktır.



Şekil 4.9 Ekran görüntüsü ile ilgili menüler

#### 4.6 Yeni Bir Modelin Yüklenmesi

Birden fazla mağara modelinin incelenmesi düşünüldüğünde her defasında programı kapatıp tekrar başlatmak tüm değerlerin sıfırlanması açısından yararlıdır. Ancak, kullanıcı açısından bakıldığında bir araştırmanın tam ortasında programı kapatıp açmak olumsuz bir durumdur. Bu nedenle, geliştirilen yazılıma yeni bir mağara modelinin yüklenebilmesi özelliği eklenmiştir. Kullanıcıya, yeni bir mağara modelini yükleyebilmek için üst tarafta yer alan menü üzerinde yer alan “Yeni” adından yüklenecek mağara modeline ait Excel dosyasını seçme imkânı verilmiştir.

#### 4.7 Konumla İlgili Temel Bilgiler

Ekranın sol bölümü kullanıcının temel bilgilere erişmesini kolaylaştırmak amacıyla geliştirilmiştir. Bu bölüm OpenGL ile OpenGL grafik arabirimi dışında kalan program arasındaki iletişimde doğabilecek performans sorununu engelleyebilmek için tamamen OpenGL içerisinde görünecek şekilde ayarlanmıştır. Bunun için grafik arabirimi 4 farklı parçaya ayrılmıştır. Bu bölümde öncelikle araştırmacının düzlem üzerinde hangi koordinatlarda bulunduğunu belirten bir bölüm vardır. Ayrıca, koordinatların hemen altında hesaplanan toplam hacim metre küp üzerinden ve toplam uzunluk metre cinsinden belirtilmektedir. Kullanıcı, programın üst kısmında yer alan menü üzerinden “Görünüm” altında “Koordinatlar” yazan bölümü seçili hale getirerek koordinatların görünmesini sağlayabilir veya seçimi kaldırarak koordinatları gizleyebilmektedir. Şekil 4.10’da ekranın sol kısmında OpenGL alanı içerisinde gösterilen koordinatlar yer almaktadır. Ayrıca hacim hesaplaması sırasında oluşan hata da gösterilmektedir.

```
X : - 18.31804
Y : 6.984662
Z : 43.66845
U : 68.23
H : 136.48
E : 43.64
```

Şekil 4.10 Koordinatların gösterimi

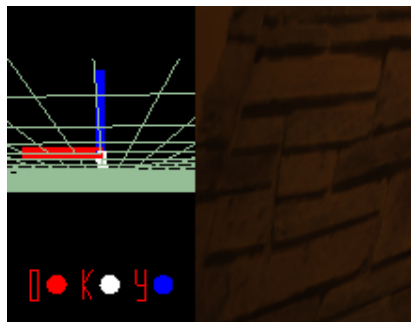
Koordinatların yer aldığı bölümün altında araştırmacının mağaranın neresinde olduğunu daha rahat anlaması için o anki konumu mağaranın üstten görünüşünü sağlayan küçük bir harita üzerinde gösterilmektedir. Şekil 4.11’de ekran görüntüsü verilen bu harita her bir istasyonun referans noktasından geçen “cubic spline” eğrilerinin birleştirilmesi ile oluşturulmuştur. Böylece kullanıcı grafiklerin birbirine

çok yakın olması durumunda nerede olduğunu daha rahat kavramaktadır. Kullanıcı, programın üst kısmında yer alan menü üzerinden “Görünüm” altında “Küçük Harita” yazan bölümü seçili hale getirerek küçük haritanın görünmesini sağlayabilir veya seçimi kaldırarak küçük haritayı gizleyebilmektedir.



Şekil 4.11 Harita üzerinde kullanıcının konumu

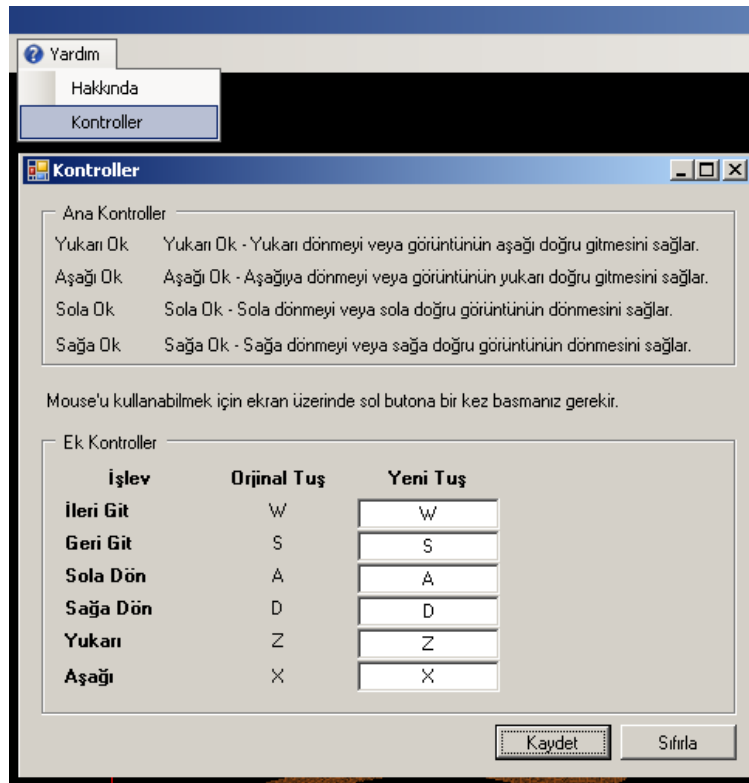
Kullanıcı mağara içerisinde nereye ne kadar döndüğünü tam olarak anlayamayacağı düşünülerek eksenlerin OpenGL içerisinde Şekil 4.12’de ekran görüntüsündeki gibi gösterilmesine ihtiyaç duyulmuştur. Kullanıcı herhangi bir yöne doğru dönerken eksenler de doğu, kuzey ve yukarıyı gösterecek şekilde güncellenmektedir. Kullanıcı, programın üst kısmında yer alan menü üzerinden “Görünüm” altında “Eksenler” yazan bölümü seçili hale getirerek eksenlerin görünmesini sağlayabilir veya seçimi kaldırarak eksenleri gizleyebilmektedir.



Şekil 4.12 Eksenler

#### 4.8 Klavye Kısa Yollarının Değiştirilmesi

Araştırmacının yazılım üzerinde esnekliğini arttırabilmek için klavye kısa yollarının değiştirilebilmesi özelliği eklenmiştir. Şekil 4.13 üzerinde görüleceği gibi temel hareketleri sağlayan W – S – A – D – Z ve X tuşlarının değiştirilmesi kullanıcı tarafından Yardım menüsü altındaki Kontroller ekranı üzerinden yapılmaktadır.



Şekil 4.13 Klavye kısa yollarının değiştirilmesi

## BÖLÜM 5

### SONUÇ

Bu çalışmada mağara ve benzeri diğer jeolojik yapılar üzerinde yapılan ölçüm sonuçlarını üç boyutlu modelleyen ve kullanıcının incelemesini sağlayan bir uygulama geliştirilmiştir. Kapalı hacimlerin modellenmesinde kullanılan uygulama ile kullanıcı üç boyutlu modelin içinde ve dışında gezinebilmekte, uzunluk, alan ve hacim ölçümleri yapabilmektedir.

C# programlama dili ile OpenGL kütüphanesi kullanılarak, programın hem Windows hem de Linux tabanlı bilgisayarlarda çalışabilir olması hedeflenmiştir. C# programlama dili .Net Framework üzerinden Windows tabanlı işletim sistemleri üzerinde çalışabilirken, Mono projesi ile Linux tabanlı işletim sistemlerinde de çalışabilmektedir.

Uygulama, mağara ve kapalı hacim modellemesi yapan diğer programlardan farklı olarak, istasyon en kesitlerinin şekillerinin belirlenmesinde fotoğraflardan yararlanmaktadır. Belirli aralıklarla çekilen her bir fotoğrafta yer alan kesit görüntüsünden sınır noktaları matematiksel işlemler sonucunda elde edilmektedir. Ardından kesitler üç boyutlu uzayda konumlandırılmakta ve bir istasyondan kendisinden sonra gelen istasyona nişan alınarak ölçüm yapıldığı için her bir kesit kendi eksenini etrafında döndürülmektedir. İstasyon aralarındaki boşluğun doldurulmasında kesit şekilleri arasında morfolojik dönüşüm uygulanmakta, bu amaçla cubic-spline eğrileri kullanılmaktadır. Ayrıca, fotoğraflarda belirtilen referans noktaları cubic-spline eğrisi ile birleştirilerek kullanıcıya mağara içerisinde üzerinde ilerleyebileceği gizli bir yol oluşturulmuştur. En kesit şekillerinin belirlenmesinde ölçekli fotoğraf kullanılması, programın öncüllerinden çok daha gerçekçi modeller üretebilmesini sağlamanın yanı sıra, kesit alanı ve hacim hesabındaki duyarlılığı da oldukça arttırmaktadır. Kesit alanının hesaplanması sırasında yaygın olarak kullanılan üçgenlere bölme yöntemi kullanılmamış, onun yerine kesiti oluşturulan noktaların bir poligonu oluşturduğu

düşünülmüş ve poligonu oluşturan noktaların birbirini takip edecek şekilde bir determinant üzerinde yazılması ile alan hesaplaması yapılmıştır. Araştırmacının mağara içerisinde önceden oluşturulan yolu izlediği düşünülerek iki istasyon arasındaki hacmin hesaplanabilmesi için hem başlangıç hem de bitiş noktasındaki kesit alanları ile cubic spline eğrisinin iki kesit arasındaki mesafesi ile çarpılıp ortalaması alınmıştır. Böylece oluşturulan modelin yapısına uygun bir hacim hesaplaması elde edilmiştir. Araştırmacının gerçeğe daha yakın sonuç elde etmek istediği varsayımı ile yeni sanal istasyonların eklenebilmesi özelliği oluşturulmuştur. Sanal istasyon sayısı arttırıldıkça hacim ve uzunluk hesaplamaları daha doğru sonuçlar vermektedir.

Kullanıcı oluşturulan üç boyutlu modeli dışarıdan inceleyebildiği gibi, kapalı hacim içerisinde de gezinebilmektedir. Aydınlatma yöntemi olarak başa takılan lamba seçilmiş, böylece kullanıcıda mağara içinde dolaşma duygusunun uyandırılması amaçlanmıştır. Mağara içerisinde araştırma yapılırken kullanıcının mağara içerisinde hangi konumda olduğu ve yön bilgileri uygulamaya eklenen harita ile eksen eğrileri sayesinde detaylı olarak gösterilmiştir. Araştırma esnasında kullanıcının bulunduğu konumun ekran görüntülerinin alınabilmesi sağlanmıştır.

Uygulamanın hızlı çalışabilmesi amacıyla çizim esnasında yapılan hesaplamaları en alt seviyeye indirecek bir mimari geliştirilmiştir. Bu nedenle, mağarayı oluşturacak yapının hesaplamalarla ilgili bölümleri sadece uygulamaya veri yüklendiği ilk anda yapılıp sonradan kullanılmak üzere geliştirilen nesnelere aktarılmaktadır. Oluşturulan tüm nesnelere tek bir nesneye bağlanmış ve veriye olan erişim hızı arttırılıp düzenli olması sağlanmıştır.

Geliştirilen uygulama mağara modellemenin yanı sıra, kapalı hacim ölçümü ve modellemesi yapılacak tüm alanlarda, örneğin inşaatlar ve mimari modellerde, tünel kazıları ve hafriyatlarda hak edişin hesaplanması amacıyla kullanılabilir.

## KAYNAKLAR DİZİNİ

Canbek, S., Adar, N., 2005, Precise Measurement of Surveying – Sections Using Image Processing Techniques, Proceeding of the 14th International Congress of Speleology, Athens

Deitel, H. M., 2006, Visual C# 2005 How to Program 2/e, Deitel & Associates

Eskişehir Mağara Araştırma Derneği Eğitim Notları, 2005

Hacettepe Üniversitesi Mağara Araştırma Kulübü, 2004

<http://www.humak.hacettepe.edu.tr/Documents/belge.html>, Erişim Tarihi:  
18.03.2010

Heckbert, P. S., 1989, Fundamentals of Texture Mapping and Image Warping

Hill, F. S., 2001, Computer Graphics Using OpenGL, Prentice Hall

Hoff, K. E., 2000, Deriving 2D and 3D Rotations

Kahaner, D., Moler, C., and Nash, S., 1989, Numerical Methods and Software, Prentice Hall

Kolar, S., 2008, Survey of Graphics Programming Languages : OpenGL vs Direct3D

Page, J., 2009, Area of a polygon,

<http://www.mathopenref.com/coordpolygonarea.html>, Erişim Tarihi: 18.03.2010

**KAYNAKLAR DİZİNİ (devam)**

Ridge, R., 2008, The Tao Framework, <http://www.taoframework.com/>, Erişim Tarihi:  
18.03.2010

Simmons, B., 2008, Area of a Convex Polygon,  
[http://www.mathwords.com/a/area\\_convex\\_polygon.htm](http://www.mathwords.com/a/area_convex_polygon.htm), Erişim Tarihi:  
18.03.2010

The Geological Society of America, 2008, The Value of Geologic Mapping

Türkiye Mağaracılar Birliği, 2007, Mağaracılıkta Ölçme Bilgisi ve Haritacılık, Eğitim  
Yayınları

Weisstein, E. W., "Cubic Spline.", MathWorld - A Wolfram Web Resource,  
<http://mathworld.wolfram.com/CubicSpline.html>, Erişim Tarihi: 18.03.2010

Wookey, B., 2001, <http://www.sat.dundee.ac.uk/arb/surveying/software.html>, Erişim  
Tarihi: 18.03.2010