

Hava ve Kara Araç Gruplarının Detaylı Arazi Verisi Üzerinde Gerçek Zamanlı Simülasyonu

Selçuk SÜMENGEN

Mühendislik ve Doğa Bilimleri Fakültesi

Bilgisayar Grafiği Laboratuvarı

Sabancı Üniversitesi

selcuk@su.sabanciuniv.edu

Selim BALCISOY

Mühendislik ve Doğa Bilimleri Fakültesi

Bilgisayar Grafiği Laboratuvarı

Sabancı Üniversitesi

balcisoy@sabanciuniv.edu

ÖZET

Geliştirilen sistemde, araç gruplarının detaylı arazi verisi üzerinde gerçek zamanlı simülasyonu sırasında karşılaşılan temel ihtiyaçların bazılarında mevcut yöntemler geliştirilerek, bazılarında da yeni yaklaşımlar getirilerek cevap verilmiştir. Arazi görselleştirme sırasında uygulanan başlıca yöntemler arasında görünür yüzey tespiti, detay seviyesi değişimi, geçişlerde arazi sürekliliğinin korunması sayılabilir. Ayrıca gezegenin bir bütün olarak görselleştirilmesi üzerine yeni bir metod geliştirilerek daha önce arazi görselleştirme sırasında sıkça kullanılan sınırlandırılmış ağaç yapısına dayalı metotlara işlevsellik kazandırılmaktadır. Araçların kontrolü ve yön tespit kabiliyetleri ise yapay zekâ algoritmalarından yola çıkarak hazırlanmıştır.

1. GİRİŞ

Karmaşık simülasyon ortamlarını gerçek zamanlı ve etkileşimli görselleştirmek için son zamanlara kadar pahalı ve amaca uygun olarak geliştirilmiş donanımlara ihtiyaç duyuluyordu. Grafik kartları teknolojisinde ve işlemci hızlarındaki ilerlemeler piyasadaki satın alınabilecek ucuz donanımlar ile karmaşık simülasyon ortamların gerçek zamanlı canlandırılmasına izin vermektedir. Bu çerçevede laboratuvarımızda bir arazi görselleştirme motoru, kullanıcı ara yüzü, çoklu araç (kara ve hava) kontrolü yapay zekâsı ve çarpışma modelleri tasarlanıp geliştirilmiştir. Simülasyon ortamı, iyi donanımlı bir kişisel bilgisayarda, yüzlerce aracı detaylı yükseklik verisi olan bir alanda gerçek zamanlı simülasyonuna olanak vermektedir.

2. ARAZİ GÖRESELLEŞTİRME

2.1 İlgili Çalışmalar

Belirli aralıklarla alınmış örneklerden oluşan yükseklik verisini arazi görselleştirmek üzere kullanmadan önce poligonlara ve ardından yüzeylere dönüştürmek gerekmektedir. Genel olarak bu işlem her üç noktaya karşılık gelen yükseklik verisinden bir üçgen oluşturarak yapılır. Geniş arazileri görselleştirme sırasında detaylı veri kullanıldığı zaman sıradan

üçgenlere ayırma metotları ile çok sayıda üçgen ortaya çıkmaktadır. Mevcut donanımlar üretilen çok sayıda üçgeni seri şekilde görüntüleyebilecek kapasiteye sahip olmalarına rağmen üst sınırları bulunmaktadır. Yükseklik verisinin çözünürlüğü arttıkça bu sınır zorlanmakta ve ek olarak tüm sistemde gözle görülür bir performans düşüklüğü oluşmaktadır.

Genel olarak kullanılan metotlar, arazinin farklı bölgelerinin, eğimlerine ve kameradan uzaklıklarına göre farklı detaylarda üçgenlere ayrılmasında ve üçgen sayısını düşürerek çok daha yüksek çözünürlükte verilerin görüntülenebilmesine dayanmaktadır.

2.1.1 Düzensiz Ağlar

Üçgenlere ayrılmış düzensiz ağlar olarak da bilinirler[8]. Üçgen oluşturmak üzere Delaunay üçgenleri yaygın olarak kullanılmaktadır [6].

2.1.2 İkili Ağaç Hiyerarşileri

İkizkenar dik üçgenlerin kenar ortaylarından ikiye ayrılarak bölünmesi esasına dayanır[24]. Sistemimizde kullandığımız sınırlandırılmış dörtlü ağaç yapıları (restricted quad-tree) bu gruba dâhildir[23].

2.1.3 İkili Ağaç Alanları

Ağaç yapısı üzerinde geniş alanların geometrik olarak saklanmasına üzerine kuruludur[16]. Parça alanlar önceden üçgenlere ayrılarak tampon bölgede tutulur. Parça aralarında yumuşak geçişi sağlamak için yakınsama teknikleri kullanılır.

2.1.4 Kesik Geometrik Alanlar

Bu teknik blok döşeme metotlarına benzemekle birlikte yeni sayılabilecek bir yaklaşımdır[22]. Bloklar haline saklanan yükseklik verisi görüntüleyici odaklı bir ağaç üzerine yerleştirilmiştir.

2.2 Sınırlandırılmış Dörtlü Ağaç Yapısı

Geliştirilen arazi görselleştirme motoru, yükseklik verisi sağlanan geniş alanları, yüzey eğimi ve uzaklığını göz önünde bulundurarak, farklı detay seviyelerinde

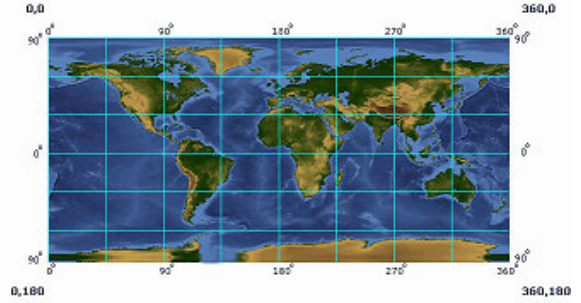
çizmektedir. Bu işlem sırasında geliştirdiğimiz algoritma veri saklamak için dörtlü ağaç yapısı (Şekil-1a) kullanmayı gerektirmektedir.



Dörtlü ağaç yapısında her bir düğümün temsil ettiği karesel alan, ait olunan atanın alanının $\frac{1}{4}$ üne karşılık gelir (Şekil-1b). Arazi bu şekilde her seferinde 4^e bölünerek ağaç yapısı üzerine yerleştirilirken, dallanmanın sonlandırılması için elde edilmek istenen detay seviyesi ve görülen bölge içerisinde kalan alan dikkate alınır. Burada dikkat edilmesi gereken nokta, çizilen yüzeyin sürekliliğinin sağlanması ve üzerinde boşlukların oluşmaması için, elde edilen alt seviye düğümlerin uygun şekilde üçgenlere ayrılması ve ekrana çizilmesi gerektiğidir. Kullanılan algoritma, ağaç oluşturulurken, alt seviye düğümlerin komşularıyla aralarındaki detay seviye farkını bir alt veya bir üst seviye ile sınırlandırır. Elde edilen en alt seviye düğümler, temsil ettikleri kareleri köşegenlerinden bölmek suretiyle en az 4 ayrı üçgen, ya da komşu düğüm detay seviyesi göz önüne alınarak, bu üçgenlerin bazılarının veya hepsinin tekrar bölünmesiyle en fazla 8 ayrı üçgen olarak ekrana çizilir ve 3×3 noktadan oluşur (Şekil-1c). Böylece poligon sayısı azaltılıp performans artışı sağlanırken, ekranda görülen arazi yüzeyi üzerinde bozulma olmadan detay seviyesi yüksek tutulabilmektedir.

2.2.1 Ağaç Oluşturulması

Ağaç yapısı içerisinde düğümler arazi üzerinde belirli bir alana karşılık gelmekte ve alt seviyelere inildikçe bu alan ufalmaktadır. Her bir düğüm dört ayrı alt düğüm ile bağlı olduğundan, her seferinde araziye de dört parçaya bölmek uygun olacaktır. En alt seviye yaprak düğümlerin en az 3×3 veri noktası içermesi gerektiğinden ve bu düğümlere ata düğüm dörde bölünerek ulaşıldığından L derinliğinde bir ağacın sorunsuz bölünebilmesi için $(2^{L+1} + 1) \times (2^{L+1} + 1)$ adet nokta içeren veriden oluşturulması gerekmektedir. Bu durum yükseklik verisinin kare şeklinde bir alana ait olmasını zorunlu kılar.



Şekil-2 Silindirik Yansım: Arazi yükseklik verisi, gezegen yüzeyinin silindirik izdüşümü alınarak her derece için belirli sayıda örnekler alınarak oluşturulur.

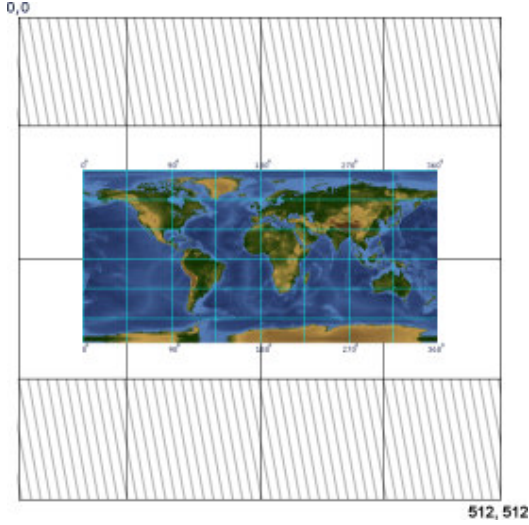
Gezegen yüzeylerinin silindirik yansım (Şekil-2) ile ölçeklendirilmesi nedeniyle sağlanan verilerin kare şeklinde bir alana karşılık gelmediği görülebilir. Uygulamada kullanılan Mars yüzey verisi $1^\circ \times 1^\circ$ alanların 2'nin katları çözünürlükte örneklenmesinden oluşmaktadır. Her $1^\circ \times 1^\circ$ alan kare şeklinde olduğu için sınırlandırılmış dörtlü ağaç yapısı ile bölünmeye uygundur. Fakat bir bütün olarak 360 meridyene karşılık 180 paralel bulunan küre şeklinde gezegenler, silindirik yansım sonrasında kare arazi şekline uymamaktadır. Genelleme yapılacak olunursa, silindirik yansımı alınan gezegenlerin N nokta/derece çözünürlükte yükseklik verisi, $(360 \times N) \times (180 \times N)$ büyüklüğünde dikdörtgen bir alan oluşturmaktadır. Dolayısıyla sürekli dörde bölünerek oluşturulan $(2^{L+1} + 1) \times (2^{L+1} + 1)$ noktaya ihtiyaç duyan sınırlandırılmış dörtlü ağaç yapısı doğrudan kullanılamamaktadır.

Bir diğer sorun ise üst bölümde bahsedilen detay geçişlerinde sağlanan sürekliliğin korunması sırasında ortaya çıkmaktadır. Dörtlü ağaç yapısı yükseklik verisi tekrar örneklenecek kare şekline getirildikten sonra kurulsu bile arazinin kare sınırlarına karşılık gelen alanındaki geçişlerde detay seviyeleri farklı olduğunda süreklilik sağlamamaktadır. Örneğin 359. meridyenden 0. meridyene geçerken yüzeylerin birbirini takip etmediği görülür.

Gezegenin bir bütün olarak görselleştirilmesi üzerine yapılan daha önceki bir çalışmada[5] bahsedilen sorunları aşabilmek için P-BDAM (Planet-Sized Batched Dynamic Adaptive Meshes) adında yapılar kullanılmıştır. Burada temel fikir gezegen yüzeyini bir küre olarak 6 yüzeyli bir küp üzerine yansımaktır. Daha sonra her bir yüzey kare arazi olarak kullanılmaktadır. Bu yaklaşım mevcut ölçüm metotlarıyla silindirik izdüşümünden alınan verilerin kare şeklinde tekrar örneklenebilmesi için ön işleme gerektirir. Ayrıca tekrar örnekleme ihtiyacından dolayı verilerin dinamik olarak değiştirilmesi sırasında performans kaybına yol açar ya da dinamik değişikliklere izin vermez.

Geliştirdiğimiz algoritmada ise mevcut coğrafi koordinat sistemine göre örneklenecek veriler doğrudan kullanılmış, dolayısıyla ön işlemeye gerek kalmamıştır. Ayrıca sınırlarda ortaya çıkabilecek yüzey uyumsuzluğu da önlenmiştir.

Mevcut koordinat sistemini kullanabilmek için $(360 \times N) \times (180 \times N)$ adet yükseklik verisi içeren dikdörtgenin genişleterek kare haline getirilmesi yerine, uygun ölçülerde bir kare içerisinde merkezleri çakışacak şekilde yerleştirilmesi tercih edilmiştir. Bu ölçülerde bir dikdörtgeni muhafaza edebilecek $(2^L + 1) \times (2^L + 1)$ formuna uygun en küçük kare $(512 \times N) \times (512 \times N)$ boyutlarındadır. Sınırlanmış dörtlü ağaç yapısı kullanılırken bu kare esas alınmaktadır (Şekil-3).



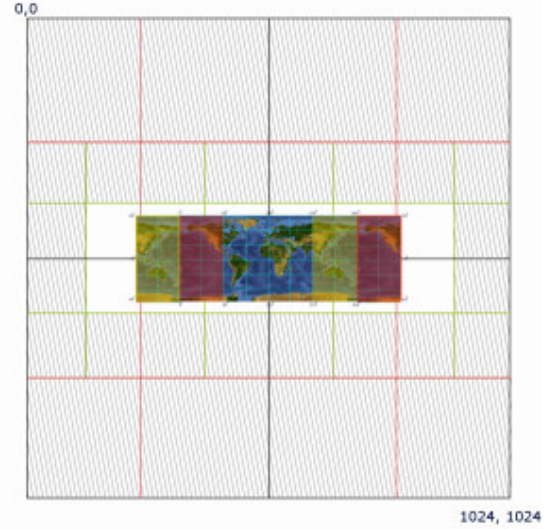
Şekil-3 Kare Yerleşim 1: Silindirik yansıtım nedeniyle dikdörtgen oluşturulan gezegen yükseklik verisinin sınırlanmış ağaç yapısı ile uyumlu hale getirilmesi.

Her iki kümedeki noktaların eşleştirilmesi sırasında kare içerisinde bazı alanlar açıkta kalmaktadır. Şekil - 3'te taralı alanlar dallanma esnasında sorun yaratır gibi gözükse de mevcut enlem ve boylamlar dâhilinde $0 - 360$, $0 - 180$ sınırlarından hiç çıkmayacağından ağacın dallanmasını sağlayan kontrol mekanizmasına ufak bir ekleme yapılarak bu sınırlar dâhilinde eşlenmemiş alanlara hiç bir zaman referans verilmemesi sağlanabilir. Bu durumda uygun işaretçiler kullanılarak hafıza ziyan edilmemiş ve ağaç bu eşlenmiş alanda dallandırılarak işlemci zamanı kaybına da neden olunmamış olur.

Kare sınırlarına karşılık gelen alanındaki geçişlerde ise sınırlanmış ağaç yapısı detay seviyesinin korunmasında genel olarak yeterli olurken, 0. meridyen üzerinden doğrudan 359. meridyene geldiğinde detay seviyeleri farkından dolayı yüzey uyumsuzluğu ortaya çıkabilmektedir. Ağaç yapısı oluşturulurken $0 - 90$ (Şekil-4 kırmızı alan) ve $270 - 360$ (Şekil-4 sarı alan) meridyen aralıklarına karşı gelen düğümler komşu olmadıklarından detay seviyesi değişikliklerinde kesintiye önlenebilir. Sınırlanmış ağaç yapısı işlevsiz kalacaktır. Bu uyumsuzluğu gidermek için ise ekranda görülebilecek en büyük küresel cisim yüzeyinin 180 meridyenden fazla olamayacağından yola çıkmıştır.

Gezegen görüntülenirken 0. meridyen üzerinde en fazla $0 - 90$ ve $270 - 360$ meridyen aralığına ait yükseklik verisine ihtiyaç duyulacaktır. Bunun yerine

$270 - 360$ meridyen aralığının bir kopyası olarak $-90 - 0$ meridyenler, $0 - 90$ meridyen aralığının bir kopyası olarak da $360 - 450$ meridyenler ayrıca temsil edilirse detay geçişlerinde ortaya çıkan yüzey uyumsuzlukları tamamen giderilmiş olur. Son olarak $(540 \times N) \times (180 \times N)$ boyutlarında dikdörtgen $(2^L + 1) \times (2^L + 1)$ formunu sağlayan en küçük kare olan $(1024 \times N) \times (1024 \times N)$ boyutlarında kare içerisinde yerleştirilir.



Şekil-4 Kare Yerleşim 2: Blok sınırında detay geçiş sürekliliği sağlamak üzere genişletilen yükseklik verisinin sınırlanmış ağaç yapısı ile uyumlu hale getirilmesi.

2.2.2 Ağaç Yapısının Dallandırılması

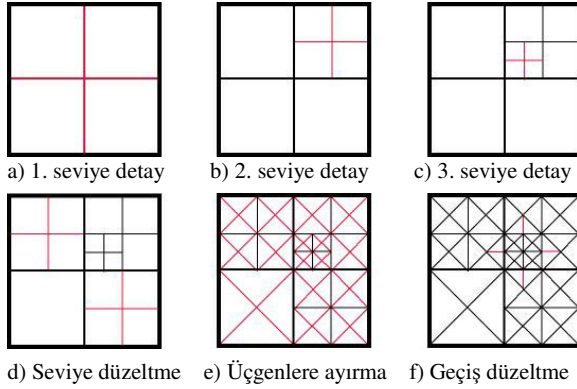
N çözünürlüklü yükseklik verisinin mevcut koordinat sistemi bozulmadan yerleştirildiği kare $(1024 \times N) \times (1024 \times N)$ boyutlarında olduğundan, her 1×1 derecelik alan ağaç yapısının 10. seviye düğümlerine karşılık gelmektedir. Bu nedenle başlangıç sırasında ağaç yapısı açılarak 10. seviye en düşük seviye olarak korunur. Burada dikkat edilmesi gereken, ağaç oluşturma sırasında bahsedilen eşleştirilme dolayısıyla gezegen izdüşümü dışına denk düşen düğümlerin açılmamasıdır.

10. seviye ve altında düşen düğümlerin açılıp açılmaması ise üzerinde bulunulan koordinata, gezegen yüzeyinden yüksekliğe, görüş açısına ve yüzey eğimine göre değişmektedir.

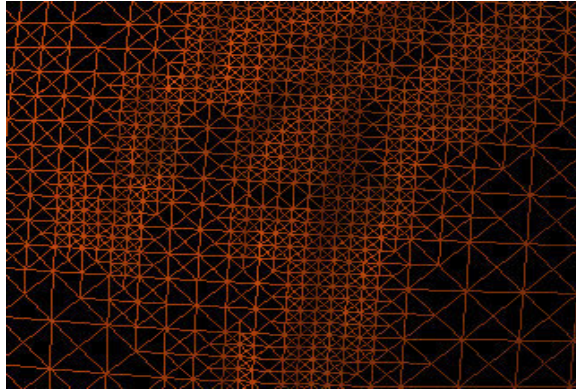
İlk olarak gezegen yüzeyinden yükseklik ve görüş açısı ele alındığında ekrana yansıtılacak olan arazi boyutları saptanır ve bunun dışında kalacak olan düğümler ön bellek süreci hariç dallandırılmaz. Bu sayede görünür yüzey tespiti yapılarak grafik kartı performansı önemli ölçüde artırılmış olur.

Görsel alan içerisinde düşen koordinatlara denk gelen düğümler ise belirli bir detay seviyesine kadar açılacaktır. Bu seviye belirlenirken performansın izin verdiği en yüksek kalitede resim elde etmek istendiğinden yüzeye yaklaşıldığında detay seviyesinin artması istenecektir. Ayrıca düz ve geniş alanlar için

daha az düşük detay, pürüzlü – eğimli alanlar için daha yüksek detay kullanılacaktır. Yüzeyin düz ya da pürüzlü olduğu düğüme ait yükseklik bilgilerinin standart sapmaları hesaplanarak görülebilir. Bu bilgi kameranın yüzeyden yüksekliği ile birleştirildiğinde görsel alan içerisine düşen düğümler için farklı detay seviyeleri ortaya çıkar.



Şekil-5 Ağaç Yapısının Dalandırılması



Şekil-6 Sınırlandırılmış Ağaç Yapısının Gezegen Yüzeyi Üzerinde Görünümü

Düğümmler farklı detay seviyelerine göre açılırken (Şekil-5 a-b-c) bloklar arası geçişlerde olabilecek yüzey uyumsuzluklarını engellemek için bu işlem sırasında komşu düğümmlerin detay seviyelerine dikkat edilir. Eğer varsa aradaki detay seviye farkı en fazla bir seviye olmalıdır (Şekil-5 c-d). Bu kısıtlama iç içe döngülerle sağlanabilir. İkinci dikkat edilmesi gereken nokta ise detay seviye farkı bulunan düğümmlerde düşük detay seviyeli düğümün ortak köşeye bakan dik kenar üçgenin daha sonra ikiye bölünmek üzere işaretlenmesidir. Bunun nedeni üçgenlere bölmenin hemen sonrasında Şekil-5 e de görülen yüksek detay seviyeli karelerin bazı köşelerinin komşu düğüme ait karede karşılık bulamamasıdır. Daha sonra bu köşeler eğer aynı düzlem üzerinde değilse ise yüzeyde bir kırılma ve boşluk oluşturacaklardır. Bu köşelerin düşük detay seviyeli düğümmlerde karşılık bulabilmeleri için komşu düğümde yer alan dik üçgenin ikiye bölünmesi yeterli olur. Bu

şekilde detay seviyeleri arasında yumuşak geçişler mümkün olmaktadır (Şekil-5 f).

2.3 Verinin Yüklenmesi

Motorun yükleyici kısmı tamamı tek seferde yüklenmesi mümkün olmayan arazi yükseklik verisinin gerek olduğu kadarıyla belleğe alınmasını ve sık kullanılan kısımların tampon bölgede tutulmasını temin eder. Yükseklik bilgisi, görsel veri elde edilirken kullanılan ağaç yapısı üzerindeki anahtar düğümmlerde paketler halinde saklanır. Arazi üzerinde gösterilen alana ve detay seviyesine uygun olarak ön yükleme yapılarak gerekli verinin çizilmeden belirli bir süre önce hazır olması sağlanır. Aynı zamanda, geniş alanların bellekte aşırı yer kaplayan yükseklik verilerinin, tamamının tek seferde yüklenmesi yerine, hafızaya parçalar halinde alınıp kısıtlı bellek kullanımına olanak verir.

3. ARAÇ SİMULASYONU

3.1 Hedefe Yönelme

Yaratılan arazi üzerinde hareket eden yüzlerce araç, yapay zekâ kullanarak hedeflere grup halinde önceden belirlenen kurallara uyarak yönelmekte, araçlar arası olası fiziksel etkileşimler ise uygun çarpışma modelleri ile gerçeğe uygun olarak canlandırılmaktadır. Şu anda kullanıcılar amaca göre değişik senaryolarda araç sayılarını, cinsini (kara ve/veya hava), operasyon tipini (keşif, toplu harekât), izleme tipini (taktik, stratejik ve operasyon) değiştirebilmektedir. Esnek yazılım alt yapısı, yeni senaryoların da kısa sürede geliştirilmesine olanak verecek şekilde tasarlanmıştır.

3.2 Çarpışma Modellemesi

Araçların birbirleriyle ve gezegen yüzeyi ile olan fiziksel etkileşimi gerçeğe yakın olarak modellenmiştir. Çarpışma bulma algoritması her bir araç için iki ayrı küresel model kullanır. Bu sayede çarpışma tespiti iki aşamada yapılarak performans artışı sağlanır.

Birinci modelde araçlar boyutlarına göre aksamları dışarı taşmayacak şekilde tek bir küre içerisine yerleştirilmiştir. Kürelerin merkezleri ve yarıçapları bilindiğinden araçların çarpışma ihtimali bulunması kürelerin kesişip kesişmediklerine bakılarak anlaşılabilir. Burada merkezler arasında uzaklığın yarıçapları toplamından kısa olup olmadığı kontrol edilerek kısa olanlar çarpışma ihtimali ile ikinci aşamaya tabi tutulur.

İkinci aşamada daha detaylı bir küresel model kullanılarak çarpışma kesin olarak bulunurken aynı zamanda çarpışma noktaları da takriben saptanır. Bu modelde çarpışma ihtimali bulunan araçlar üç ayrı küre kullanarak takribi şekilleri oluşturulur. Her bir küre aracın belirli bir bölümünü temsil ettiğinden, çarpışma ihtimali bulunan araçların küreleri tüm ikili eşleştirmeler yapılarak kontrol edilir.

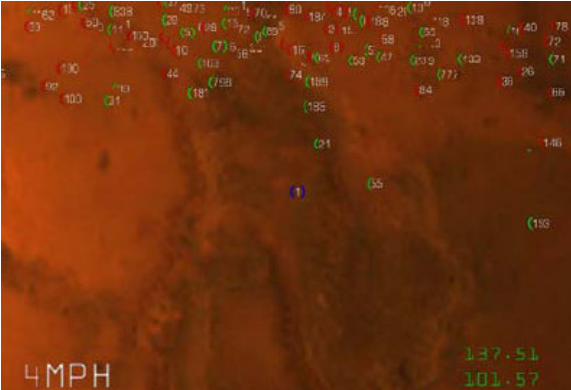
Bulunan çarpışma noktasından, diğer aracın yön ve hızı da göz önünde bulundurularak, aracın yeni konumu ve hız vektörü hesaplanır.

4. SONUÇLAR VE DEĞERLENDİRME



Şekil-7 Yakın Plan Araç Kontrolü

Alt yapımızı test etmek için NASA'dan 1,93 GB (128 nokta/derece) Mars yüzey bilgisi ve aynı kaynaktan Ay (4 nokta/derece) yüzey bilgileri kullanılmıştır. Ekteki resimler Pentium 4 2.66 GHz, 1 GB Ram, ATI Radeon 9800 Grafik kartı kullanılarak yüzün üzerinde aracın gerçek zamanlı (60 kare/saniye) simülasyondan alınmıştır. Şekil - 7 yakın plan araç kontrolü, Şekil - 8 ise genel simülasyon yönetimi sırasında alınmıştır.



Şekil-8 Simülasyon Yönetimi

Mevcut yapımız şu an bir gezegenin bütünü veya belirli bir parçasını gerçek zamanlı olarak görüntüleyebilme yeteneğine sahiptir. Aynı zamanda yüzlerce hava ve kara aracının kontrolüne olanak sağlayan simülasyon ortamına da olanak sağlamaktadır. Yakın gelecekte sistemimizi yüksek çözünürlüklü doku örnekleriyle çalışacak düzeye getirmeyi planlamaktayız.

5. KAYNAKLAR

- [1] Apu, R. A. and Gavrilova M. L., "Adaptive Mesh Generation for Real-Time Terrain Modelling", Proceedings of the twentieth annual symposium on Computational geometry, Brooklyn, New York, USA, 2004.
- [2] Atılım, Ç., Bilgisayar Grafikleri, Seçkin Yayınevi, 2003.

- [3] Blow, J., "Terrain Rendering at High Levels of Detail.", Proceedings of the 2000 Game Developers Conference, San Jose, March 2000.
- [4] Cignoni, P., F. Ganovelli, E. Gobbetti, F. Marton, F. Ponchio and R. Scopigno, "Planet-sized batched dynamic adaptive meshes (p-bdam).", In Proceedings IEEE Visualization, Conference, Seattle USA, October 2003.
- [5] Cignoni, P., F. Ganovelli, E. Gobbetti, F. Marton, F. Ponchio and R. Scopigno, "Adaptive TetraPuzzles: Efficient Out-of-Core Construction and Visualization of Gigantic Multiresolution Polygonal Models", ACM Transactions on Graphics, ACM Press, Volume 23, August 2004.
- [6] Daniel Cohen-Or and Yishay Levanoni, "Temporal continuity of levels of detail in delaunay triangulated terrain," in IEEE Visualization '96, Roni Yagel and Gregory M. Nielson, Eds., San Francisco, California, Oct. 1996, pp. 37-42, IEEE.
- [7] Duchaineau M., M. Wolinsky, D. E. Sigi, M. C. Miller, C. Aldrich, and M. B. Mineev-Weinstein, "ROAMing terrain: Real-time optimally adapting meshes", Proceedings IEEE Visualization '97, Phoenix, Oct. 1997.
- [8] Evans, W., D. Kirkpatrick, and G. Townsend., "Right Triangular Irregular Networks", Algorithmica, Springer-Verlag New York, LLC, Special Issue on Algorithms for Geographical Information, 2001.
- [9] Go, J., T. Vu, and J. Kuffner, "Autonomous behaviors for interactive vehicle animations.", ACM SIGGRAPH/Eurographics Symposium on Computer Animation, Los Angeles, August 2004.
- [10] Gregorski, B., M. Duchaineau, P. Lindstrom, V. Pascucci and K. I., "Interactive view-dependent rendering of large IsoSurfaces.", In Proceedings of the 13th IEEE Visualization 2002 Conference, Boston, 2002.
- [11] Gross, M. H., R. Gatti and O. Staadt, "Fast Multiresolution Surface Meshing", Visualization '95, Atlanta, Georgia, October 1995
- [12] Hoppe, H., "Smooth view-dependent level-of-detail control and its application to terrain rendering.", IEEE Visualization '98, North Carolina, October 1998.
- [13] Hoppe, H., "View-dependent refinement of progressive meshes.", SIGGRAPH 97 Conference Proceedings, Annual Conference Series, Los Angeles, August 1997.
- [14] Kamphuis, A. and M.H. Overmars, "Finding Paths for Coherent Groups using Clearance", ACM SIGGRAPH/Eurographics Symposium on Computer Animation, Los Angeles, August 2004
- [15] Larsen, B. D. and N. J. Christensen, "Real-time Terrain Rendering using Smooth Hardware Optimized Level of Detail," Journal of WSCG, vol. 11(2), 2003.
- [16] Levenberg, J. "Fast view-dependent level-of-detail rendering using cached geometry.", In Proceedings of the 13th IEEE Visualization 2002 Conference, Boston, 2002.
- [17] Lindstrom P., D. Koller, L. Hodges, W. Ribarsky, N. Faust and G. Turner, "Level-of-detail Management for Real-Time Rendering of Phototextured Terrain.", GVU TR 95-06, 1995

- [18] Lindstrom, P. and V. Pascucci, "Visualization of Large Terrains Made Easy", IEEE Visualization 2001, California, October 2001.
- [19] Lindstrom, P. and V. Pascucci., "Terrain Simplification Simplified: A General Framework for View-Dependent Out-of-Core Visualization.", IEEE Transactions on Visualization and Computer Graphics, IEEE Computer Society, May 2002.
- [20] Lindstrom, P., D. Koller, L.F. Hodges, W. Ribarsky, N. Faust and G. Turner, "Real-Time, Continuous Level of Detail Rendering of Height Fields", International Conference on Computer Graphics and Interactive Techniques, New Orleans, Louisiana, USA, August 4-9, 1996.
- [21] Lloyd, B. and P. Egbert., "Horizon occlusion culling for real-time rendering of hierarchical terrains", Proceedings of the conference on Visualization '02, Boston, 2002.
- [22] Losasso, F. and Hoppe, H. "Geometry Clipmaps: terrain rendering using nested regular grids", ACM Transactions on Graphics, ACM Press, Volume 23, 2004-10-13
- [23] Pajarola, R., "Large scale terrain visualization using the restricted quadtree triangulation.", IEEE Visualization '98, North Carolina, 1998.
- [24] Renato, P., M. Antonijuan and R. Lario., "QuadTIN: Quadtree based Triangulated Irregular Networks", Proceedings of the conference on Visualization '02, Boston, 2002.
- [25] Rottger, S., W. Heidrich, P. Slussallek and H-P Seidel, "Real-Time Generation of Continuous Levels of Detail for Height Fields", Proc. 6th Int. Conf. in Central Europe on Computer Graphics and Visualization, 1998.
- [26] Seidelmann, P.K. and V.K. Abalakin, M. Bursa, M.E. Davies, C. De Bergh, J.H. Lieske, J. Oberst, J.L. Simon, E.M. Standish, P. Stooke, and P.C. Thomas, "Report of the IAU/LAG Working Group on Cartographic Coordinates and Rotational Elements of the Planets and Satellites", Preprint, Celestial Mechanics and Dynamical Astronomy, in press, 2002.
- [27] Ulrich T., "Rendering Massive Terrains using Chunked Level of Detail Control", SIGGRAPH 02, Texas USA, 2002.
- [28] Youbing, Z., Z. Ji, S. Jiaoying and P. Zhigeng, "A Fast Algorithm For Large Scale Terrain Walkthrough", CAD/Graphics'2004, Kunming, August 2004.