

DAĞITIK SİSTEMLERDE UYDU GÖRÜNTÜSÜ TRANSFERİ: RASTER VE VEKTÖR TEMSİL KARŞILAŞTIRMASI

Süleyman EKEN*

Kocaeli Üniv. Müh. Fak. Bilgisayar
Mühendisliği Bölümü
suleyman.eken@kocaeli.edu.tr

Recep BOSTANCI

Kocaeli Üniv. Müh. Fak. Bilgisayar
Mühendisliği Bölümü
rp.bostanci@gmail.com

Ahmet SAYAR

Kocaeli Üniv. Müh. Fak.
Bilgisayar Mühendisliği Bölümü
ahmet.sayar@kocaeli.edu.tr

Geliş Tarihi: 25 Ekim 2012, Kabul Tarihi: 1 Ocak 2013

ÖZET

Dağıtık sistemlerde görüntü işleme geliştirmek sınırlı network bant genişliği ve görüntü boyutunun çok büyük olmasından dolayı zordur. Diğer yandan, çoğu zaman sadece uzaysal ve topolojik sorgular gerektiren uygulamalar piksel tabanlı bilgiye ihtiyaç duymazlar. Raster görüntünün vektör temsillerini kullanan bu tür uygulamaların daha iyi performans kazanımları vereceği düşünülmektedir. Biz bu makalede Landsat Gökçeada uydu görüntüsünü web servisleri yardımıyla bir makineden diğerine transfer ettiğimiz bir senaryo kurguladık ve değişik boyutlu harita verilerinin hem raster hem de vektör formatının iletim ve cevap sürelerini test edip analiz ettik.

Anahtar Kelimeler: İnternet Üzerinden Veri İletimi, Raster Veri, Vektör Veri, Web Servisler.

SATELLITE IMAGE TRANSFERRING IN DISTRIBUTED SYSTEMS: COMPARISON OF RASTER AND VECTOR REPRESENTATION

ABSTRACT

Distributed image processing applications are constrained by the limited network bandwidth and huge image sizes. On the other hand, in some cases, applications containing images include spatial and topological queries not requiring pixel-based information. In such applications, using vector representations of raster images is expected to give better performance results. In this paper, we setup a scenario to transfer a Landsat Gokceada satellite image from one machine to another through web services. We perform the tests for both vector and raster forms of the same image and analyze transmission and response times.

Keywords: Data Transferring over the Internet, Raster Data, Vector Data, Web Services.

1. GİRİŞ

Coğrafi Bilgi Sistemi (CBS), coğrafi verilerin; toplanması, depolanması, işlenmesi, yönetimi, mekânsal analizi, sorgulaması ve sunulması fonksiyonlarını yerine getiren donanım, yazılım, coğrafi veri ve yöntem bütünüdür [1]. CBS teknolojisi kentsel planlama, kartografya, çevresel etki değerlendirmesi, kaynak planlama, pazarlama, lojik, askeri uygulamalar, araç takip sistemleri ve diğer amaçlar için kullanılabilir.

Raster ve vektör veri kullanma CBS'nin işlevselliği için önemlidir. Raster veri hava fotoğrafı gibi herhangi dijital bir görüntüdür ve gri renk ton değerlerini içeren

iki boyutlu bir dizi formatındadır. CBS'de yaygın olarak kullanılan raster yükseklik modelleridir. Vektör veri ise CBS'de noktalar, çizgiler ve poligonlardan oluşan matematiksel olarak tanımlanabilen veri tipleridir. CBS'de nokta ile bir yangın vanasının konumu; çizgi ile yol, nehir gibi lineer özellikler; poligon ile bir bölgenin alanı veya bir üniversitenin çevresi gibi iki boyutlu özellikler temsil edilebilir. Üçünün içinde nokta ile en az detaylı bilgi ifade edilirken, poligon ile en fazla detaylı bilgi gösterilmiş olur.

İnternet, CBS kullanıcıları için mekânsal veri (raster veya vektör veri) transfer etmek, analizler gerçeklemek ve coğrafi sonuçları göstermek adına

* Sorumlu Yazar

önemli bir içerik yayınlama ortamı haline gelmiştir. Web teknolojilerinin gelişmesiyle coğrafi bilgiye dayalı uygulamalar artmıştır.

Hızla artan mekânsal veri miktarı ile network bant genişliği arasında bir ikilem vardır. Cevap süresinin (istemcinin sunucudan mekânsal veri istemesinden veriyi almasına kadar geçen süre) kısa olması web-CBS uygulamalarında önemlidir. Web üzerinden mekânsal veri transferinde kullanıcılar istek yaptıktan sonra veri indirilene kadar hiçbir işlem yapmadan beklemek zorundaydı. Bertolotto tarafından bu probleme çözüm getirildi [2]. Bu çözüm beraberinde etkili transfer tekniği eksikliğinden mimaride bir takım problemler getirdi.

Çoğu zaman görüntü içeren uygulamalar piksel bazlı bilgi gerektirmeyen, uzaysal ve topolojik sorgular içermektedir. Raster görüntünün vektör temsillerini kullanan bu tür uygulamaların daha iyi performans kazanımları vereceği düşünülmüştür. Bu makalede Landsat-7 Gökçeada uydu görüntüsünü web servisler yardımıyla bir makineden diğerine transfer eden bir senaryo gerçekleştirdik. Ayrıca değişik boyutlu harita verilerinin hem raster hem de vektör formatının iletim ve cevap sürelerini test edip analiz ettik.

Makalemizin geri kalan kısmı şu şekilde özetlenebilir: 2. bölüm raster ve vektör veri transferi ile ilgili çalışmaları kapsamaktadır. 3. bölümde web servis gerçekleştirmek için kenarlaştırmaya dayalı mimari ele alınmıştır. 4. bölümde önerilen yaklaşımla raster ve vektör veri transferinin performans değerlendirmeleri vardır. 5. bölümde ise yöntemin sonuçları tartışılmış ve gelecek çalışmalara yer verilmiştir.

2. İLGİLİ ÇALIŞMALAR

Şimdiye kadar raster veri (örneğin yüksek çözünürlüklü uydu görüntüsü) iletimiyle ilgili kapsamlı birçok çalışma yapıldı. Bu çalışmalar sıkıştırma [3], wavelet dönüşüm [4], fraktal geometriye dayalı [5, 6] ve hibrit sıkıştırma [7] metodlarını içerir. Ayrıca web üzerinden raster verinin daha az detaylı versiyonundan daha detaylı versiyonuna doğru artırımlı olarak transferini sağlayan yaygın bir teknik geliştirilmiştir [8]. Bu yöntemin avantajları, etkin veri transferi, hızlı yanıt süresi ve istenilen detayda veri göndererek disk ve zaman kazanımı şeklindedir.

Mekânsal vektör verisini transfer etmek için de birçok algoritma ve model geliştirilmiştir. Bu algoritmalara üçgenlenmiş düzensiz ağa (ÜDA) dayalı [9,10], ızgara tabanlı [11] metodlar örnek olarak verilebilir. İlk olarak Bertolotto ve Egenhofer [12] istemci-sunucu mimarisine dayalı artırımlı vektör verisi iletimini ortaya koymuşlardır. Buttenfield [13] artırımlı çözünürlük seviyelerinde geometri ve topolojiyi bozmadan çizgi inceltme metodu ile vektör

koordinatlarını iletmiştir. Han vb. [14] vektör veri iletmek için istemci-sunucu modeli geliştirmiştir. Yang vb. [15] poligonların kesişim noktalarını azaltarak artırımlı poligon verisi transferini gerçekleştirmişlerdir. Ai vb. [16] poligonların artırımlı iletimi için bir poligondaki topolojiyi dikkate alan, bir poligon ile çizgi ve başka poligon gibi diğer objeler arasındaki ilişkiyi önemsemeyen değişiklikler birikim modelini (DBM) ortaya koymuşlardır.

Yukarıda da söz edildiği üzere artırımlı olarak vektör verisini transfer etmek hala önemli bir problemdir. Bizim çalışmamız, vektör harita verisini transfer etmek için web servis tabanlı bir mimari sunmaktadır. Sunucudan istemciye iletilecek vektör verisinin büyüklüğü sunucu tarafında kenarlaştırma yöntemiyle azaltılarak veri web servis teknolojisi ile iletilmektedir. Landsat-7 Gökçeada uydu görüntüsünün hem raster hem de vektör hali iletilmek suretiyle yöntemin etkinliği deneysel sonuçlarla gösterilmiştir.

3. YÖNTEM

Raster görüntülerde çözünürlük ve renk derinliği arttıkça dosya boyutu artmaktadır. Örneğin 2"x3" büyüklüğünde çözünürlüğü 300 dpi ve renk derinliği 24-bit olan bir görüntünün boyutu 12Mb'm üzerinde olurken, bu görüntün vektör formatında boyutu 96-bit kadar olmaktadır. Büyük boyutlu (1Mb'tan fazla) dosyalar internet üzerinden transfer edilirken her iki tarafta da yüksek hızlı internet bağlantısının olmasını gerektirir. Vektör formatında hem işlemciyi yormamış hem de sınırlı bant genişliğinde veri transferi sağlanmış olur. Vektör harita verisinin de internet üzerinden iletim zamanını kısaltmak için veri ilk önce sunucu tarafında vektörleştirme metoduyla basitleştirilip web servisler yardımıyla istemciye gönderilir. İstemci de aldığı veriden tekrar haritayı oluşturabilir.

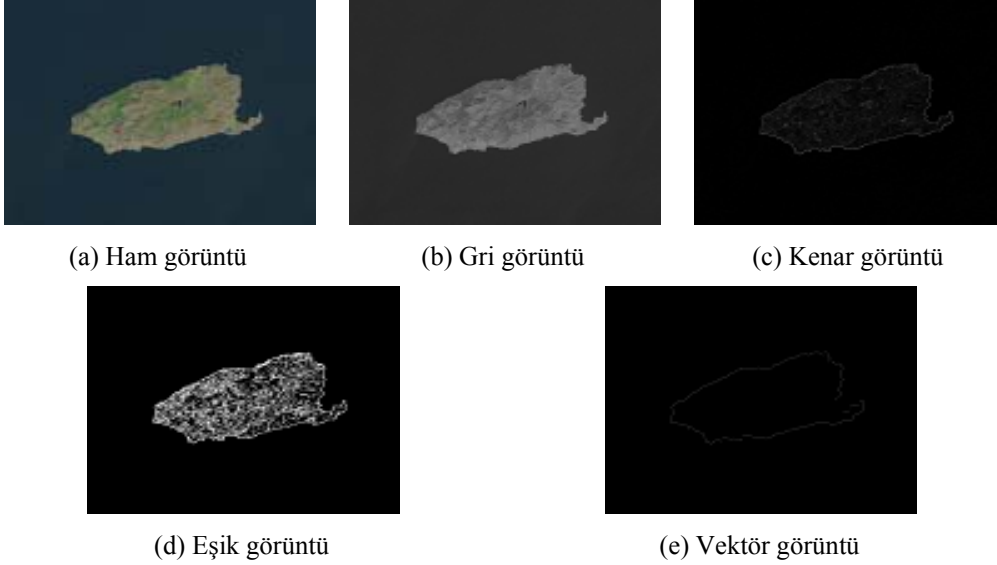
3.1 İletilecek Raster Harita Verisinin İndirgenmesi

İletilecek ham ada görüntüsü, resmi daha sonra işleyecek algoritmaların işini kolaylaştırmak için gri seviyeye çevrilmiştir (Şekil 1.b). Görüntüdeki önemli yapısal kısımların muhafaza edilip daha az anlamlı bilgilerin filtrelenmesi yoluyla işlenilecek veri sayısını azaltmak sebebiyle kenar bulma tekniklerinden Canny kenar saptama algoritması [17] ile gri görüntü kenar görüntüsüne çevrilmiştir (Şekil 1.c). Görüntü kirlilikleri kenar bulma algoritmalarının başarısını etkilemektedir. Uydu görüntülerinde görüntü kirlilikleri genelde havanın kapalı, bulutlu olmasından veya düşük olasılıkla da olsa farklı cisimlerden ortaya çıkar. Gürültülerin de kenar gibi algılanmaması için seçilen görüntülerin temiz olmasına dikkat ettik.

Uygulamamızda kenar saptama işleminden geçmiş görüntüdeki aday resmin geri kalanından ayırmak için piksellerin gri seviye değerlerinin tüm imgedeki

dağılımlarının yanı sıra yerel özelliklerini de göz önüne alan Otsu eşikleme [18] yöntemi kullanılmıştır (Şekil 1.d). Otsu eşikleme filtresinden geçen resmin adayı temsil eden her bir pikseli 1, geri kalan kısmı 0 olarak kodlanmıştır. İmge üzerinde yataydan başlayarak satır satır soldan sağa tarandığında her satırda ilk rastlanan 1 değerli pikselleri adanın sol sınır değerlerini sağdan sola doğru tarandığında her satırda ilk rastlanan 1 değerli pikselleri de adanın sağ sınır değerlerini verir. Yine imge üzerinde dikey taramalar yaparak imgenin her sütun için rastlanan yukarıdan

aşağıya tarandığında ilk 1 değerli pikselleri adanın üst sınır, aşağıdan yukarıya tarandığında ilk rastlanan 1 değerli pikselleri de alt sınır değerlerini verir (Şekil 1.e) [19]. Burada görüntünün sınırını ifade eden piksellerin bulunması hedeflenmiştir. Sınır değerlerinde var olan kopukluklar şeklin konturunu vektörel olarak göstermede bir problem oluşturmamaktadır. Önemli olan konturu ifade eden nokta setlerinin sıralı bir şekilde elde edilebilmesidir.



Şekil 1.Raster Gökçeada harita verisinin indirgenmesi.

3.1 Vektör ve Raster Harita Verisini Web Servisler Aracılığıyla Transfer Etme

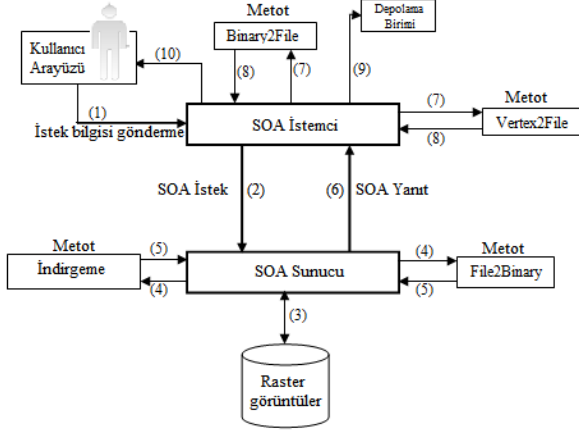
Servis Odaklı Mimari (SOA), birçok uygulamanın "servis" denilen kendi modüllerini veya fonksiyonlarını başka uygulamalar tarafından kullanılacak şekilde tasarlandığı bir mimari kavramdır. Web servisleri, SOA altyapısında kullanılan bir yöntemdir. Basitçe uygulamanın özellikleri ya da servisleri bir web sayfası gibi URL adresi olmakta ve diğer uygulama bu web adresini çağırarak uygulamanın özelliklerini kullanabilmektedir. Böylece bir uygulamanın özellikleri başka uygulamaların da kullanacağı şekilde açılmış olmaktadır. Web servis mantığında, kullanıcı, diğer makineye birtakım veriler gönderir ve geriye yine işlenmiş birtakım veriler alır. Alan tarafın, diğer makinede gerçekleştirilen uygulamalar, kullanılan algoritmalar, fonksiyonlar hakkında hiçbir bilgisi yoktur.

Web servis aracılığıyla vektör harita verisi transferi sağlayan mimarinin (Şekil 2) temel iki bileşeni vardır: SOA istemci ve SOA sunucu. Veri akışı adım numaraları, transfer çizgileri üzerinde parantez içinde belirtilmiştir.

Sistemin genel çalışma prensibi şu şekildedir: Kullanıcı arayüz aracılığıyla sunucu tarafından almak istediği haritanın türünü (raster veya vektör) seçer. İstek devamlı çalışır ve erişilebilir durumdaki web servise iletilir. Sunucu, istemcinin seçimine göre gerekli işlemi yaptıktan sonra gerekli bilgileri geri istemciye gönderir. Sistemin detaylı açıklaması:

Adım1: Kullanıcı arayüz aracılığıyla sunucu tarafından almak istediği haritanın türünü (raster veya vektör) seçer.

Adım2: SOA istek adımdır. Arayüz aracılığıyla girilen veri web servis sunucusuna gönderilir.



Şekil 2. Web servisleri aracılığıyla vektör ve raster veri gönderen mimarinin prototipi.

Adım3: Sunucu üzerinde işlem yapılacak görüntüye ulaşır.

Adım4, 5: Kullanıcı raster görüntü isteğinde bulunmuşsa ilgili resim “File2Binary” metoduyla ikili diziye çevrilir, vektör görüntü isteğinde bulunmuşsa yukarıda da sözü edilen indirgeme yöntemiyle basitleştirilir. İndirgeme yöntemi ile basitleştirilme işleminden sonra istemci tarafına iletilecek olan bilgiler, 0 ve 1’lerden oluşan raster veri yapısında bir görüntü değildir. Bu bilgi, sınır değerlerini ifade eden poligon veri türünden vektör bir veridir. Aşağıda gösterildiği şekilde bir yapıya sahiptir:

POLYGON ((1 2) (3 4) (5 6) (7 8) (9 10) (1 2))

Adım6: SOA yanıt adıdır. İstenilen görüntünün ikili dizisi veya nokta(vertex) dizisi istemciye gönderilir.

Adım7, 8: Kullanıcı raster görüntü istemişse ikili diziden raster, vektör görüntü istemişse nokta dizisinden vektör görüntü oluşturulur. İstemci tarafında nokta dizisinden oluşturulan görüntü, orijinal halinden çok uzaktır. Bunun nedeni orijinal görüntünün renk ve doku gibi özelliklerinin kaybolmuş olmasıdır. İstemci tarafında görüntü üzerinde, alan, çevre ve yakınlık gibi mekânsal sorguların yapılacağı düşünülerek sadece görüntüyü ifade eden sınır değerlerinin bilinmesi yeterli görülmüştür.

Adım9: Oluşturulan görüntüler belirlenmiş bir dizine resim dosyası olarak kaydedilir.

Adım10: Görüntü arayüz ekranında gösterilir.

4. DENEYSEL SONUÇLAR ve ANALİZ

Raster ve vektör harita verisinin performansını ölçmek için yapılan testler ağ bant genişliği 10 Mps olan bir intranette gerçekleştirildi. Testler için web servislerle

sunucu-istemci mimarisi (RPC modeli) kuruldu. Farklı büyüklükteki veriler üzerinde iletim ve yanıt sürelerini analiz ettik. Web servisleri oluşturmak için Axis kullanıldı [20]. Uygulama; sunucu tarafı 2.3 GHz işlemci ve 4 GB belleğe sahip Tomcat 6.0 sunucusu, istemci tarafı ise 2.4 GHz işlemci ve 4 GB belleğe sahip bir web istemci üzerinde yapıldı.

İletim zamanı, ikili veya nokta dizisinin sunucudan istemciye gönderilme zamanıdır. Yanıt zamanı ise kullanıcının istediği veri türünü seçmesinden sonucun sunucu tarafından hesaplanıp istemcinin veriyi almasına kadar geçen süredir. Yanıt süresi şu şekilde hesaplanabilir:

$$T_Y(\text{yanıt}) = t_0(\text{istek}) + t_1(\text{file2binary}) + t_2(\text{iletim})$$

$t_0(\text{istek})$ zamanı kullanıcı isteğinin istemciden sunucuya iletilmesine, $t_1(\text{file2binary})$ raster verinin ikili dizisine dönüştürülmesine kadar geçen süredir. $t_2(\text{iletim})$ yukarıda açıklandı. $T_R(\text{yanıt})$ ise indirgeme (vektörleştirme) olmadan geçen yanıt süresidir. Sunucu tarafında vektörleştirme olursa yanıt süresi $T_{R\Box}(\text{yanıt})$ şu şekilde hesaplanabilir:

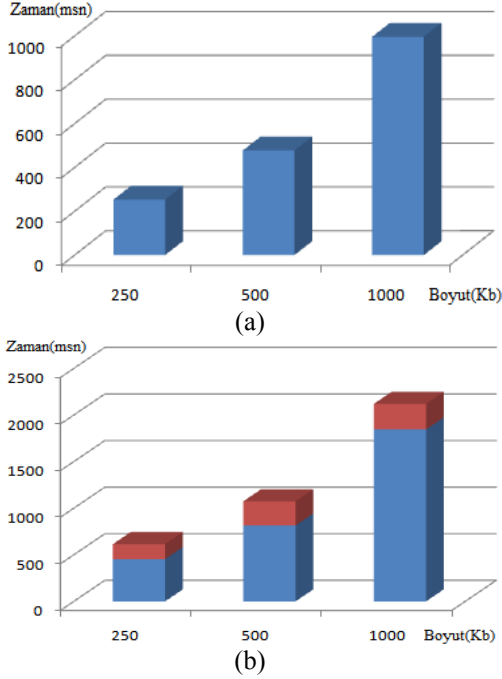
$$T_{R\Box}(\text{yanıt}) = t_0(\text{istek}) + t_v(\text{vektörleştirme}) + t_1(\text{iletim})$$

$t_v(\text{vektörleştirme})$ sunucu tarafında vektörleştirme (indirgeme) için geçen süredir. Landsat-7 Gökçeada uydu görüntüsü için vektörleştirme ve vektörleştirme olmadanki durumlarına göre yanıt süreleri karşılaştırıldı. Ayrıca Gökçeada uydu görüntüsü parçalara ayrılarak farklı büyüklükler (Şekil 3) için sistemin ölçeklenebilirliği test edildi.



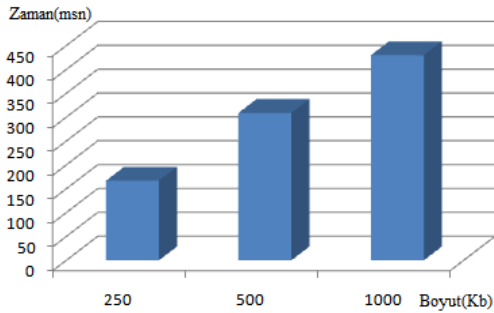
Şekil 3. Gökçeada uydu görüntüsünün parçaları.

Şekil 4(a)’da 1000 Kb boyutlu Gökçeada uydu görüntüsü ve boyutları 245-525 Kb arasında değişen parçaları için vektörizasyon işlemi olmadan gerekli yanıt sürelerini göstermektedir. Şekil 4(b) ise vektörleştirme işlemi yapıldığında gerekli yanıt sürelerini göstermektedir.



Şekil 4. (a) Vektörleştirme işlemi olmadan gerekli yanıt süreleri, **(b)** Vektörleştirme işlemi yapılmış gerekli yanıt süreleri.

Vektörleştirme işlemi, algoritmanın karmaşıklığından dolayı çok zaman aldığı için vektörleştirme işlemiyle iletim için de çok zaman almaktadır. Şekil 4(b)'deki mavi kısımlar vektörleştirme için gerekli süreyi göstermektedir. Görüldüğü üzere 1000Kb büyüklüğündeki Gökçeada uydu görüntüsünün yanıt süresinin yaklaşık $\frac{3}{4}$ 'ü (1745 msn) kadardır.

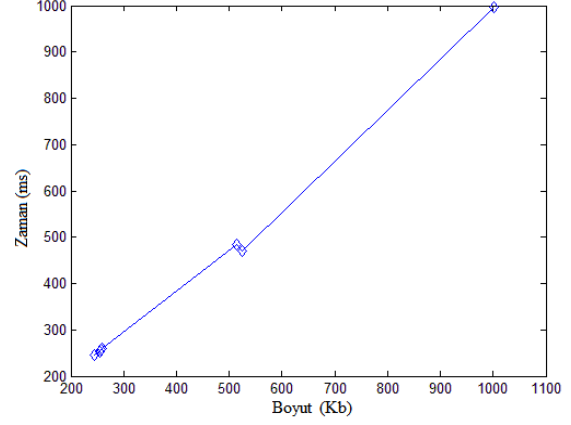


Şekil 5. Nokta iletimi için gerekli yanıt süresi.

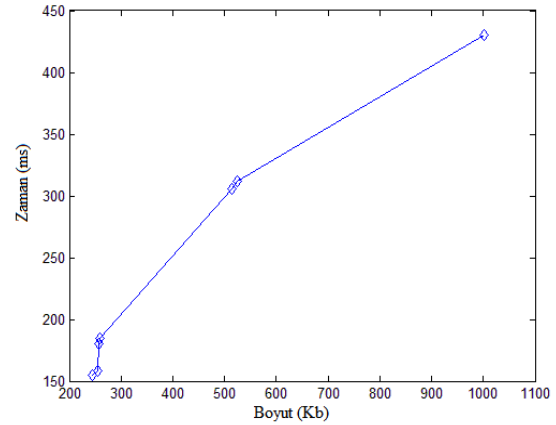
Vektörleştirme işlemi koşum anında çok zaman aldığı için, raster görüntünün vektörleştirme işleminin koşum zamanından önce yapılıp nokta dizilerinin veritabanı veya yerel diskte hazır olduğunu kabul edersek, nokta dizisinin iletimi için gerekli yanıt süresi Şekil 5'te verilmiştir. Şekil 5 nokta dizisinin iletimi için gerekli yanıt süresinin 1Mb büyüklüğündeki raster veri iletiminden dört kat daha hızlı olduğunu göstermektedir.

Kullanıcı ara yüz aracılığıyla sunucu tarafından vektör formatta bir harita isteyip sunucu tarafında

vektörizasyon işlemi yapıldıktan sonraki veya vertex iletiminden sonraki haritanın kullanıcı tarafında oluşturulması için geçen süreler Şekil 6 ve Şekil 7'de gösterilmiştir.



Şekil 6. Vektörizasyon sonrası harita oluşumu için geçen süre.



Şekil 7. Vertex iletimi sonrası harita oluşumu için geçen süre.

5. BULGULAR VE TARTIŞMA

Hızla artan mekânsal veri miktarı ile network bant genişliği arasında bir çelişki vardır. Cevap süresinin (istemicinin sunucudan mekânsal veri istemesinden veriyi almasına kadar geçen süre) kısa olması web-CBS uygulamalarında önemlidir.

Bu çalışmada, vektör harita verisini transfer etmek için web servis tabanlı bir mimari sunulmuştur. Sunucudan istemciye iletilecek vektör verisinin büyüklüğü sunucu tarafında kenarlaştırma yöntemiyle azaltılarak veri web servis teknolojisi ile iletilmektedir. Yapılan testlerin sonuçları orijinal raster verisinin büyüklüğünün vektörleştirme işlemiyle azaldığını ve sistemin ölçeklenebilir olduğunu; fakat vektörleştirme algoritmasının çok zaman alıcı olmasından dolayı vektörleştirme işleminin çalışma

zamanında yapılması iletim hızını azalttığını göstermiştir. Daha sonra, iletilecek uydu görüntüsünün vektör halinin yerel diskte tutulduğunu kabul edip iletim işlemi gerçekleştirildi ve raster veri iletiminden hızlı olduğu tespit edildi.

Gelecek çalışmalarda vektörleştirme işleminin performansını artırarak sistemin daha büyük veriler üzerinde uygulanabilirliği araştırılacaktır. Ayrıca vektör harita verisinin iletim hızını artırmak için merkezi/dağıtık algoritmalar geliştirilecektir.

6. KAYNAKLAR

- [1] Fischer M.M., Nijkamp P., “Geographic Information Systems, Spatial Modeling, and Policy Evaluation”, Springer, New York, 1993.
- [2] Bertolotto M., Egenhofer M.J., (1999) “Progressive Vector Transmission”, *Proceedings of the 7th ACM international symposium on Advances in geographic information systems*, pp. 152-157.
- [3] Gonzales R.C., Woods R.E., “Digital Image Processing”, Addison-Wesley, 1993.
- [4] Martin M.B., Bell A.E., (2001) “New Image Compression Techniques Using Multiwavelets and Multiwavelet Packets”, *IEEE Transactions on Image Processing*, Vol. 10, No. 4, pp. 500-510.
- [5] Barnsley M.F., (1989) “Fractals Everywhere”, Academic Press, San Diego.
- [6] Woon W.M., Anthony Tung S.H., Tao Y., Siu C.T., Siong C.T., Lian T.Y. (2000) “Achieving Highdata Compression of Self-Similar Satelliteimages Using Fractal”, *Proc. Of IEEE International Symposium on Geoscience and Remote Sensing*, Vol. 2, pp. 609-611.
- [7] Ho A.T.-S., Siu-Chung T., Siong-Chai T., Lian-Teck Y., (1999) “A Novel Hybrid Bi-Orthogonal Wavelets/ADPCM Algorithm for Very Low Bit Rate Satellite Image Compression”, *Proc. Of IEEE International Symposium on Geoscience and Remote Sensing*, Vol. 4, pp. 2051-2053.
- [8] Rauschenbach U., Schumann H., (1999) “Demand-Driven Image Transmission with Levels of Details and Regions of Interest”, *Computer and Graphics*, Vol. 23, No. 6, pp. 857-866.
- [9] Magillo L.P., Puppo E., (2000) “Compressing Triangulated Irregular Networks”, *GeoInformatica*, Vol. 4, No. 1, pp. 67-88.
- [10] Park D., Cho H., Kim Y., (2001) “A TIN Compression Method Using Delaunay Triangulation”, *International Journal of Geographical Information Science*, Vol. 15, No. 3, pp. 255-270.
- [11] Wessel P., (2003) “Compression of Large Data Grids for Internet Transmission”, *Computers & Geosciences*, Vol. 29, No. 5, pp. 665-671.

[12] Bertolotto M., Egenhofer M.J., (2001) “Progressive Transmission of Vector Map Data Over the World Wide Web”, *GeoInformatica*, Vol. 5, No. 4, pp. 345-373.

[13] Battenfield B.P., (2002) “Transmitting Vector Geospatial Data Across the Internet”, *Lecture Notes in Computer Science*, pp. 2478, 2002.

[14] Han H., Tao V., Wu H., (2003) “Progressive Vector Data Transmission”, *Proceedings of the 6th AGILE*, pp. 103-113.

[15] Yang B.S., Purves R.S., Weibel R., (2004) “Implementation of Progressive Transmission Algorithms for Vector Map Data in Web-Based Visualization”, *Xth Congress of the International Society for Photogrammetry and Remote Sensing (ISPRS)*, pp. 12-23.

[16] Ai T., Li Z., Liu Y., (2004) “Progressive Transmission of Vector Data Based on Changes Accumulation Model”, *11th International Symposium on Spatial Data Handling*, pp. 85-96.

[17] Canny J., (1986) “A Computational Approach to Edge Detection”, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 8, No. 6, pp. 679-714.

[18] Otsu N., (1979) “A threshold selection method from gray level histograms”, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 9, No. 1, pp. 62-66.

[19] Eken S., Sayar A., (2011) “Vector Modeling of Island Satellite Images for Spatial Databases”, *Proc. of International Science and Technology Conference. (ISTEC 11)*, pp. 25-30

[20] <http://ws.apache.org/axis/> (accessed 24 May. 2012).

ÖZGEÇMİŞLER

Arş. Gör. Süleyman EKEN

1985 yılında Konya’da doğdu. İlk, orta ve lise öğrenimini Konya’da tamamladı. 2004 yılında girdiği Karadeniz Teknik Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü’nden 2009 yılında mezun oldu. 2012 yılında Kocaeli Üniversitesi Bilgisayar Mühendisliği Bilgisayar Bilimleri Anabilim Dalı’nda yüksek lisans eğitimini tamamladı. 2010 yılından itibaren Kocaeli Üniversitesi Bilgisayar Mühendisliği Bölümü’nde araştırma görevlisi olarak görev yapmaktadır. Mekânsal veri tabanları, CBS, görüntü işleme ve dağıtık hesaplama temel ilgi alanlarıdır.

Recep BOSTANCI

1987 yılında Karabük’te doğdu. 2007 yılında girdiği Kocaeli Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü’nden 2012 yılında mezun oldu. Temel ilgi alanları ileri veri tabanları ve uzaktan algılama görüntü işleme uygulamalarıdır.

Yrd. Doç. Dr. Ahmet SAYAR

Lisans eğitimini İTÜ İşletme Mühendisliği'nde tamamladı. Bilgisayar Bilimlerindeki yüksek lisansını Syracuse (Syracuse, NY, USA) Üniversitesi'nde, doktorasını ise Indiana(Bloomington, IN, USA)Üniversitesi'nde tamamladı. Doktora eğitimi sürecinde araştırmacı olarak Los Alamos Ulusal Laboratuvarı'nda (New Mexico, USA) ve Community

Grids Laboratuvarı'nda (Indiana, USA) çalıştı. Halen Kocaeli Üniversitesi Bilgisayar Mühendisliği Bölümü'nde yardımcı doçent olarak görev yapmaktadır. Temel araştırma alanları; dağıtık sistemler, CBS, dağıtık mekânsal veri tabanları, dağıtık sistemlerde Web servisleri ve bilimsel problem çözümünde Web teknolojileri uygulamalarıdır.