

BİNA TAHSİS PLANLARINDAN 3-BOYUTLU ŞEHİR MODELLERİNİN ÜRETİLMESİ VE GÖRÜNTÜLENMESİ

PRODUCTION AND VISUALIZATION OF 3D CITY MODELS FROM BUILDING ALLOCATION PLANS

Oğuzcan Oğuz, Medeni Erol Aran, Türker Yılmaz, Uğur Güdükbay

Bilkent Üniversitesi, Bilgisayar Mühendisliği Bölümü
06800, Bilkent, ANKARA

{oğuzcan,merol,yturker,gudukbay}@cs.bilkent.edu.tr

Özetçe

Bu bildiriye sanal şehir modelleri oluşturulmasında kullanılmak üzere otomatik olarak çeşitli bina modelleri üretimini sağlayacak bir yöntem ve elde edilen verilerin görüntülenmesi için bir görüntüleme sistemi sunulacaktır. Modelleri üretmek için geliştirilen yöntem, bina modellerini otomatik olarak ve belli noktalarda olasılıksal olarak üretebilmektedir. Model üretimi, sistem içerisinde kullanılan kuralların ve istenen bina niteliklerinin ayarlanması ile yönlendirilebilir. Kullanılan yöntemin geliştirilmesinde şekil gramerleri baz alınmıştır. Sistem kadastro planları gibi planlar üzerinden 3-boyutlu şehir modellerini üretebilecek seviyededir. Sisteme girdi olarak modeli oluşturulacak şehrin taban planları, kuralların tanımları ve bazı nesnelere tanımları verildiğinde bina modelleri 3-boyutlu hale getirilmekte, istenen detaylar eklenebilmekte ve elde edilen veriler 3-boyutlu ortamda görüntülenebilmektedir.

Abstract

This paper presents a method for the automatic generation of different building models to be used to populate virtual cities and a system of visualization of generated city models. The proposed method incorporates randomness but the derivation process could be steered by the help of derivation rules and assigned attributes. The derivation method is inspired by the shape grammars. During the derivation process, the floor plans of the actual cities are used to generate 3D city models. Given the city plans, the derivation rules and definitions of some basic nesnects, the system generates 3D building models and the generated city model could be visualized.

1. Giriş

Şehir planları her belediye mevcut ve elektronik ortamlarda saklanmaktadır. Bu veriler 2 boyutlu olup üzerinde ilave planların yapılması, 3-boyutlu olarak görüntülenmesi ve her hususun detaylarıyla incelenmesi son 10 yıl içerisinde artan inceleme konularından biri olmuştur. Bu alanda örneğin Google firması, Google Earth isimli yazılımıyla uydu fotoğraflarını şehir planlarından elde edilen 3 boyutlu görüntülerle birleştirerek görsel bir coğrafi bilgi sistemini kullanıma açmıştır. Hali hazırda mevcut 3-boyutlu şehirler daha çok

ABD’nde bulunan birkaç şehirden oluşmaktadır. Benzer bir çalışma CitySurf isimli bir yazılım ile ülkemizde de geliştirilmiştir.

Büyük ve karmaşık sistemlerin modellenmesi bilgisayar grafiği açısından önemli bir problemdir. Gelişen bilgisayar sistemleri ve görüntüleme algoritmaları sanal ortam modellerine ihtiyacı artırmaktadır. Günümüzde bilgisayar sistemleri, oldukça karmaşık ve büyük üç boyutlu modellerin görüntülenmesine olanak vermektedir. Görüntüleme sistemleri askeri eğitim ve şehir planlamadan bilgisayar oyunlarına ve sanal turizme kadar birçok alanda kullanılmaktadır. Bu sistemlerde kullanılmak üzere, mevcut şehirlerin birebir modellerini oluşturmak zaman ve para gibi maliyetler açısından çok masraflı olacaktır.

Bu çalışmanın esas amacı, şehir planlarında yer alan 2 boyutlu verileri, 3-boyutlu ortama gerçekçi olarak taşımak ve elde edilen 3-boyutlu veriyi gerçek zamanlı olarak görüntülemektir. Yapılan çalışma halen başlangıç aşamasında olup zaman içerisinde çeşitli bina tasarımlarının eklenmesi ve görüntüleme algoritmalarının daha etkin hale getirilmesi ile her alana yaygınlaştırılabilir hale getirilecektir. Halihazırda, modellerin oluşturulmasında mevcut şehirlerin planlarından yararlanılmakta ve bu planlarındaki binaların 2-boyutlu taban planları kullanılarak her bina modeli ayrı ayrı oluşturulmaktadır. Elde edilen veriler görüntüleme algoritmasına transfer edilmek suretiyle gerçek zamanlı görüntüleme yapılabilmektedir.

2. İlgili Çalışmalar

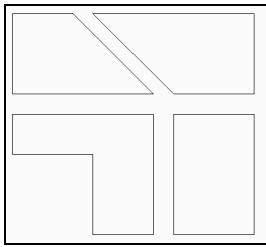
Mevcut şehir modellerinin oluşturulmasında başarı sağlayan bir yaklaşım fotogrametri ve resim işleme ile bina ve sokakların modellerinin oluşturulmasıdır. Bir başka yaklaşım ise havadan lazer tarayıcılar yardımıyla uzaktan algılama teknolojilerinin kullanılmasını baz alır. Bu yaklaşımlar oldukça başarılı ve güvenilir şehir modelleri sağlamasına karşın, otomatik olarak üretim yapmadıklarından çoğunlukla pratik değillerdir.

Şehir modellemede gramerlerin yani L-sistemlerinin kullanılması yeni değildir [1]. Önceki bir çalışmada, L-sistemleri kullanılarak sokak yapıları oluşturulmuştur [2]. Ayrım gramerleri yardımıyla detaylı bina modellerinin oluşturulması oldukça başarılı sonuçlar vermiştir [3]. Ayrım gramerleri küme ve şekil gramerlerinin bir sentezidir [4]. Ayrım

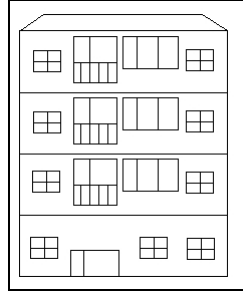
gramerleri 3 boyutlu nesnelere aynı hacim içinde diğer 3 boyutlu nesnelere ayırır ya da dönüştürür. Bina modelinin üretim süreci bütün nesnelere daha fazla ayrılmayacak son nesnelere sona erer ve bu son nesnelere bina modelini temsil eder. Model oluşturma işlemi, değerleri sayısal olarak belirlenmiş nitelikler yardımıyla yönlendirilir, böylece üretilen bina modeli amaçlanan nitelikleri ya da mimari yönelimleri yansıtır. Yine bu çalışmada kontrol gramerleri kullanılmıştır. Kontrol gramerleri, model oluşturma sürecinde amaçlanan tasarım niteliklerine tutarlı bir biçimde ulaşılmasını sağlar.

3. Model Üretim Sistemi

Sistem AutoCAD firmasının “data exchange file” (DXF) model formatını kullanmaktadır. DXF formatı oldukça basit, standart ve genel olarak kabul gördüğünden mevcut 3 boyutlu model formatlarının en temeli olarak kabul edilebilir. Sistem çıktı olarak bina modellerini DXF formatında oluşturmaktadır. DXF formatı yüzey desenlerini saklayamaz ancak katman bilgisi bu amaçla kullanılabilir. Sistem girdi olarak şehir planlarını DXF formatında alır ve binaların taban planlarını kullanarak her binanın modelini ayrı ayrı oluşturur (Şekil 1 (a)).



(a)



(b)

Şekil 1. (a) basit bir 2-boyutlu şehir planı. (b) aynı tip katlardan oluşan bir bina yüzü.

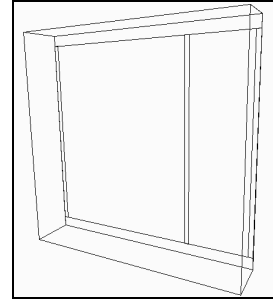
Bina modelleri, her biri poligon taban planların bir kenarına karşılık gelecek şekilde bina yüzeylerinden oluşur. Bina planının her kenarı ayrı ayrı işlenir. Bina planının bir kenarına karşılık gelecek şekilde oluşturulan bir yüzey belli sayıda kattan oluşur. Binanın bir yüzeyi aynı tip katlardan oluşabilir ya da katların tipleri farklı olabilir.

Bina yüzleri oluşturulurken, yüzey nesnesi belli sayıda, 2 boyutlu yüzey şeklinde kat nesnelere dönüştürülür. Daha sonra oluşturulan bu kat nesnelere daha önceden tanımlanmış ayırım kuralları ile diğer ara nesnelere dönüştürülür (Şekil 1 (b)). Bu süreç tüm nesnelere son nesne olana dek devam eder. Bütün son nesnelere, bunlara balkon, kapı ya da pencereler örnek verilebilir, DXF formatında dosyalarda tanımlıdır. Son nesnelere boyutları ve konumları ayarlanarak bina yüzeyleri oluşturulur ve yine DXF formatında yazılır. Bir binanın yüzeylerinin hepsi üretildiğinde bina modeli tanımlanmış olur.

3.1. Nesnelere

Sistemde kullanılan iki tip nesne vardır:

- **Son Nesnelere:** Son nesnelere DXF formatında tanımlanmıştır ve herhangi bir üç boyutlu tasarım programıyla kolayca tasarlanabilir. Son nesnelere herhangi bir sayıdaki yüzeyden oluşur, bu yüzeyler DXF formatında 3DFACE olarak tanımlanmıştır ve birim boyutlara sahiptir (Şekil 2). Son nesnelere desenler ile kaplanmak istenirse, DXF formatındaki “layer” bilgileri bu amaçla kullanılabilir.
- **Ara Nesnelere:** Ara nesnelere, kural dosyasında tanımlanmış kurallara aracılığı ile son nesnelere ya da yine diğer ara nesnelere ayrılacak nesnelere. Bina üretim süreci kat nesnelere ile başlar. Bütün ara nesnelere, değerleri sayısal olarak belirlenmiş bazı niteliklere sahiptirler. Bu nitelikler, nesneye bazı kuralların uygulanmasını engelleyerek üretim sürecini yönlendirirler. Ara nesnelere şekil olarak basitçe 3 boyutta bir dikdörtgen yüzey olarak düşünülebilir.



Şekil 2. Bir pencereyi temsil edebilecek son nesne örneği.

3.2. Kurallar

Üretim sürecinde, ara nesnelere son nesne olana dek, kurallara yardımıyla şekillendirilir. Kurallar üretim sürecini yönlendirirler. Bütün ara nesnelere, kendilerine uygulanabilecek kuralların bulunduğu bir kural kümesine sahiptir. Bütün kurallar, değerleri sayısal olarak belirlenmiş niteliklere sahiptirler. Bu nitelikler ara nesnelere uygulanacak kuralın seçiminde önemli rol oynarlar. Bir ara nesne için kural seçimi yapılırken sadece nesnenin nitelikleri ile uygun niteliklere sahip kurallara dikkate alınır. Nesneye uygulanabilecek birden fazla kural olduğu durumda bazı niteliklere öncelik verilerek en uygun kural seçilebilir ya da kuralların önceden belirlenmiş istatistiksel ağırlıkları dikkate alınarak olasılıksal bir seçim yapılabilir. Nesnelere uygulanabilecek iki tip kural vardır: *olasılıksal ayırım* ve *sabit ayırım*.

3.2.1. Olasılıksal Ayırım

Olasılıksal ayırım bir ara nesneyi iki boyutta, ölçüleri belirli oranda olasılıksal olarak belirlenmiş sıra ve kolonlara ayırır. Kural içinde sonuçta ortaya çıkması istenen sıra ve kolonların ölçüleri için birer alt limit belirlenmiştir. Bir sıra veya kolonun boyutu verilen alt limitten büyük ve alt limitin iki katından küçük olacak şekilde belirlenir. Bir ara nesneye olasılıksal ayırım uygulandığında, ayırım sonucunda çıkan nesnelere tek çeşit olabilir, ancak bu bir son nesne ya da yine bir ara nesne olabilir. Şekil 3’de olasılıksal ayırım ile ilgili örnek verilmektedir.

```

<Floor>
  <Split Balcony="+ Window="+>
  <Random minWidth="2" minHeight="3">
  <Face Balcony="+ Window="+></Face>
  </Random>
  </Split>
  ...
</Floor>

```

Şekil 3. "Floor" adlı bir ara nesneye uygulanabilecek, "Split" adlı basit bir olasılıksal ayırım kuralı. Bu kural uygulandığında 2 boyutlu matris yapısında "Face" nesnelere oluşmakta, bu nesnelere son nesnelere ya da yine ara nesnelere olabilmektedir.

3.2.2. Sabit Ayırım

Sabit ayırım tipindeki kurallar, ara nesnelere belirli sayıda sıra ve kolona ayırır. Oluşacak sıra ve kolonların boyutlarının oranı kural içinde belirlenmiştir. Bir ara nesneye sabit ayırım kuralı uygulandığında, oluşan sıra ve kolonların boyutları kuralın uygulandığı ara nesnenin boyutlarıyla doğru orantılıdır. Sabit ayırım sonucunda oluşacak alt nesnelere sayısı ve tipleri kural içinde belirlidir. Şekil 4'de bir sabit ayırım kuralı verilmiştir.

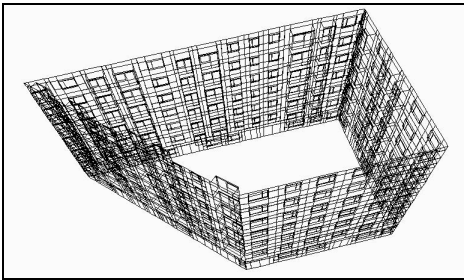
Elde edilen model örnekleri Şekil 5 ve 6'da görülmektedir. Şekil 7'de ise elde edilen bir bina modelinin doku kaplanmış hali görülmektedir. Modeller bu aşamadan sonra üç temel grafik hızlandırmasına (kapatılan alanların atılması, bakış piramidi dışında kalan alanların atılması ve başka yöne bakan poligonların atılması) sahip bir algoritmaya gönderilerek görüntülenebilmektedir.

```

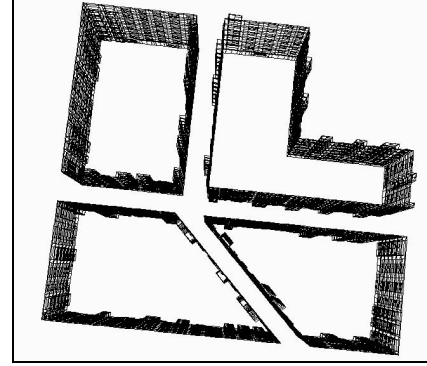
<Face>
<Balcony>
  <Fixed>
  <xProportions x1="1" x2="4" x3="1"></xProportions>
  <yProportions y1="4" y2="1"></yProportions>
  <Elements>
    <Wall></Wall>
    <Wall></Wall>
    <Balcony></Balcony>
    <Wall></Wall>
    <Wall></Wall>
    <Wall></Wall>
  </Elements>
  <Fixed>
</Balcony>
...
</Face>

```

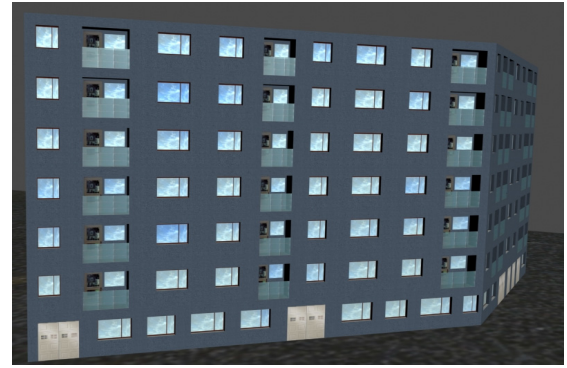
Şekil 4. "Face" ara nesnesine uygulanabilecek, "Balcony" adlı basit bir sabit ayırım kuralı. Ayırım sonucunda oluşacak sıra ve kolonların boyutlarının birbirlerine oranları kuralın içinde belirlenmiştir. <Element> başlığı altında, ayırım sonucunda oluşacak nesnelere tipleri belirlenir.



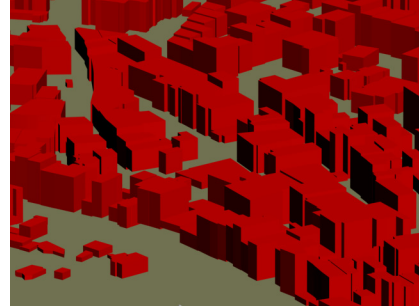
Şekil 5. Oluşturulmuş bir bina modeli.



Şekil 6. Dört binadan oluşan bir blok.



Şekil 7. Yüzeyi doku kaplanmış bina modeli.



Şekil 8. İstanbul Tarihi Yarımada'ya ait bina taban bilgisi ve yükseklik bilgisi kullanılarak elde edilmiş bir kesit

4. Üretilen Model Örneklerinin Kıyaslanması

Yalnızca bina tabanlarının ve yükseklik bilgilerinin kullanılmasıyla elde edilen bina modelleri hem istenen detayda bilgi vermemekte, hem de görüntülenen modelin mimari yapısını gerçekçi bir şekilde yansıtmamaktadır. Sunulan teknik, istenen mimari yapıdaki binaların kısa sürede üretilmesine olanak sağlamaktadır.

Şekil 8'de, İstanbul Tarihi Yarımada'ya ait bir kesit bina taban ve yükseklik bilgisi kullanılarak görsellenmiştir. Her ne kadar gölgeleme ve renklendirme yapılmış olsa da binalara ait temel yapı taşları (pencere, balkon vb.) modellenmediği için görüntü yeterli kadar gerçekçi değildir.

5. Görüntüleme

Şehir görüntüleme sistemleri, etkili bir gezinti yapmak için, görüntüye aktif olarak katılım göstermeyen verilerin atılmasına şiddetle ihtiyaç duymaktadır. Bakış piramidi dışındakilerin atılması ve arkada kalan bölgelerin atılması için etkili algoritmalar mevcuttur. Ancak, kapatılan bölgelerin atılması algoritmaları hâlâ çok maliyetlidir. Özellikle, nesne uzayına bağlı kapatılan bölgelerin atılması algoritmaları, genellikle her bir gezinti noktası ve her bir bakış açısı için görüntünün önceden hesaplanmasına ihtiyaç duymaktadır.

Saklanması gereken veri miktarı aşırı oranda arttığından, gezinti yapılabilecek alanın hücrelendirilmesi ve böylece önışlemeden geçen kapatılan bölgelerin atılması işleminin sonucu olan bilgi miktarının azaltılması büyük önem taşımaktadır. Kapatılan alanların atılmasını tespit etmek için ön işleme ihtiyaç duyan algoritmalar her bir bakış noktası için büyük miktarda veri depolamaya ihtiyaç duymaktadır. Mimari yapıların içinde yapılacak gezintilerde hücrelendirme işlemi odaların doğal hücrelere tekabül etmesi nedeniyle çok kolaydır [5]. Ancak, şehir gezintilerinde, hücrelendirme işlemi çoğunlukla model tasarım zamanında [6], yarı otomatik yöntemlerle [7] ya da bina tabanlarının yükseltilmesi ile model detaylarının azaltılmasını sağlayarak yapılmaktadır [8],[9],[10].

Bu çalışmamızda şehir modelinin gezinti yapılabilecek alanları tespit edilerek [11] bu alanlar üzerinden hızlandırma algoritmaları uygulanmıştır. Bu amaçla NVIDIA grafik işlemcisinin donanım özelliği olan OCCLUSION_QUERY (Kapatılan Sorgulaması) gerçekleştirilmektedir. Böylece daha önceden sadece OpenGL şablon tampon belleği ile yapılabilecek bu sorgulamayı çok büyük ölçüde hızlandırmıştır. Kapatılan yerlerin atılmasına yönelik işlemden sonra her hücre için potansiyel görünen bina listesi barındırılmakta ve koşturma zamanında kullanıcı hangi hücredeyse oraya ait görüntü listesi hızlı bir şekilde görüntülenmektedir. Görüntüleme esnasında ise kullanıcıya doğru dönmemiş nesne kesimleri de donanım tarafından elenerek hızlı bir görüntüleme sağlanmaktadır.

6. Sonuç ve Gelecek Çalışmalar

Bu çalışmada sanal şehirler modellerinde kullanılmak üzere sanal bina üretecek bir sistem sunulmuştur. Binalar, mevcut bir şehir planındaki izdüşümleri kullanılarak, şekil gramerleri yardımıyla olasılıksal olarak oluşturulmaktadır. Modeli üretim aşamasında nesnelere tanımlanmış kurallar gereği diğer alt nesnelere bölünmekte ve sonuçta bina modeli DXF formatında tanımlanmış son nesnelere oluşmaktadır. Oluşturulacak binaların çeşitleri daha önceden modellenmiş son nesnelere şekillerine ve üretim sırasında kullanılacak kural kümesine bağlıdır. Kuralları ve başlangıç nesnesinin niteliklerini belirleyerek bina üretimini yönlendirmek ve dolayısıyla üretim sonucunda oluşacak binanın şeklini belirlemek mümkündür.

Gelecekte çeşitli türden ve mimari tarzdan yapıları da modelleyebilecek sınıf kütüphanelerini oluşturmayı planlamaktayız. Bu kütüphaneler kullanılarak şehirler gerçeğine daha uygun olarak modellenebilecektir. Unutulmamalıdır ki, amacımız şehirleri uydudan elde edilmiş görüntülerindeki gibi modellemek değil, gerçek planlarına uygun ancak bina şekilleri itibarıyla farz ve kabul edilen tarzlarında modelleyebilmektir.

Halen görüntüleme işlemlerinde depolanması gereken veri miktarını azaltmayı hedefleyen alternatif veri yapıları üzerinde çalışmalarımız devam etmektedir. Amacımız her bina için sadece işaretler kullanarak ve saklanması gereken veri miktarını çok büyük oranda azaltacak veri yapıları tasarlamaktır. Nesne unsurlarından bağımsız bir veri yapısı bu yeni veri yapılarını üretmemizde en önemli araç olacaktır. Aynı zamanda bu veri yapılarını kullanmak suretiyle bir şehir görüntüleme sistemi üzerinde de çalışmayı planlamaktayız.

Kapatılan alanların gösteriminde bina tabanlı işlemler yapmak yerine binaları ana eksenlerinden dilimlere ayırmak ve görüntü bilgilerini dilimler cinsinden saklamak suretiyle bu optimizasyonu sağlama düşüncesindeyiz. Bu konudaki motivasyonumuz ise, mevcut kapatılan bölgeleri tespit eden yöntemlerin binaların tümünü görünür ya da görünmez olarak kabul etmeleridir. Bu yöntemler, karmaşık mimari yapıların görüntülenmesinde grafik veriyolunu gereksiz yüklemekte ve kısmi görünürlük için büyük oranda veri saklamak zorunda kalmaktadırlar. Kısmi görüntüleme neticesinde çok büyük veri tasarrufu sağlanacağını değerlendirmekteyiz.

7. Teşekkür

Bu çalışma Türkiye Bilimsel ve Teknolojik Araştırma Kurumu (TÜBİTAK) tarafından EEEAG 104E029 ve 105E065 numaralı projeler kapsamında desteklenmektedir.

8. Kaynakça

- [1] Decoret, X., Sillion, F., "Street generation for city modelling", *Arch. and Urban Ambient Env.*, 2002.
- [2] Parish, Y.I.H., and Müller, P., "Procedural modeling of cities", *Proc. of ACM SIGGRAPH*, pp. 301-308, 2001.
- [3] Wonka, P., Wimmer, M., Sillion, F., and Ribarsky, W., "Instant Architecture", *ACM Transactions on Graphics (Proc. of Siggraph)*, 22(3): 669-677, 2003.
- [4] Stiny, G., "Introduction to shape and shape grammars", *Environment and Planning, B*, 7, 343-351, 1980.
- [5] Funkhouser, T.A., Sequin, C.H. and Teller, S.J., "Management of large amounts of data in interactive building walkthroughs," *Proc. of ACM Symposium. on Interactive 3D Graphics*, pp. 11-20, 1992.
- [6] Schaufler, G., Dorsey, J., Decoret, X., and Sillion, F., "Conservative volumetric visibility with occluder fusion," *Proc. of SIGGRAPH*, pp. 229-238, 2000.
- [7] Downs, L., Möller, T., and Séquin, C.H., "Occlusion horizons for driving through urban scenes," *Proc. of SIGGRAPH*, pp. 121-124, 2001.
- [8] Wonka, P., Wimmer, M., and Schmalstieg, D., "Visibility preprocessing with occluder fusion for urban walkthroughs," *Proc. of Rend. Tech.*, pp. 71-82, 2000.
- [9] Durand, F., Drettakis, G., Thollot, J., and Puech, C., "Conservative visibility preprocessing using extended projections," *Proc. of SIGGRAPH*, pp. 239-248, 2000.
- [10] Schmalstieg D., and Tobler, R.F., "Exploiting coherence in 2.5-D visibility computation," *Computers & Graphics*, 21(1):121-123, 1997.
- [11] Yılmaz, T., Güdükbay, U., "Extraction of 3D navigation space in virtual urban environments", *13th European Signal Processing Conference*, 2005.