

# Hava fotoğraflarından çizgisel detayların yarı otomatik olarak belirlenmesi

Oktay EKER\*, D. Zafer ŞEKER

İTÜ Fen Bilimleri Enstitüsü, Jeodezi ve Fotogrametri Mühendisliği Programı, 34469, Ayazağa, İstanbul

## Özet

*Bu çalışmada; dijital hava fotoğraflarından çizgisel ve alansal detayların sınırlarının ve merkez hatlarının yarı otomatik olarak belirlenmesini sağlayan bir yöntem ve bu yöntemin uygulamaya konmasına yönelik bir yazılım geliştirilmiştir. Geliştirilen yöntem, görüntü bölümlenme ve düzey kümesi algoritmalarının birlikte kullanılmasına dayanmaktadır. Yöntemin uygulanabilirliğinin araştırılması amacıyla 1:35000 ölçekli siyah beyaz hava fotoğrafı üzerinde yarı otomatik detay çizme işlemleri gerçekleştirilmiştir. Bununla birlikte; İTÜ Ayazağa Kampüsünü içeren renkli ortofoto görüntüleri kullanılarak yöntemin doğruluk araştırması yapılmış ve binalarda  $\pm 0.463m$ , yollarda ise  $\pm 0.663$  karesel ortalama hatalar tespit edilmiştir. Yapılan doğruluk araştırması sonucunda, geliştirilen yöntemin, kullanılan dijital hava fotoğrafının  $\pm 1$  pikselinin boyutuna eşit olan bir hata kriterine sahip olduğu sonucuna ulaşılmıştır. Bununla birlikte; bu yöntemin fotogrametrik harita üretiminde ve CBS için fotogrametrik veri sağlanmasında yeni bir yöntem olarak kullanılabilmesi değerlendirilmiştir. Özellikle: Göller, sulu dereler ve binalar gibi homojen yapıdaki detayların sınırlarına ait vektör verilerin toplanmasında çok başarılı ve etkili bir şekilde kullanılabilmesi görülmüştür. İstenildiği takdirde, tolerans değerinin uygun olarak belirlenmesiyle, söz konusu detaylar üzerinde gözle ayırt edilemeyen sınıflandırmalar ve bölümlenmeler gerçekleştirilebileceği tespit edilmiştir. Kaliteli yolların sınırları ve/veya merkez hatları (kullanılan fotoğrafın ölçeğine ve mekânsal ayırma gücüne bağlı olarak) etkili ve hızlı bir şekilde çizilebileceği, ayrıca kırıklık toleransı değerleri değiştirilerek istenilen kırıklıkta vektör veriler elde edilebileceği sonucuna varılmıştır. Raster veriden vektör veriye dönüşümde hem sınırların hem de merkez hatların kullanılabilmesinin etkinliğe çok katkı sağlayacağı düşünülmektedir.*

**Anahtar Kelimeler:** Görüntü bölümlenme, düzey kümesi, yarı otomatik, dijital hava fotoğrafı.

\*Yazışmaların yapılacağı yazar: Oktay EKER. [oktay.eker@hgk.mil.tr](mailto:oktay.eker@hgk.mil.tr); Tel: (312) 595 22 81.

Bu makale, birinci yazar tarafından İTÜ Fen Bilimleri Enstitüsü Jeodezi ve Fotogrametri Mühendisliği Programı'nda tamamlanmış olan "Hava fotoğraflarından yarı otomatik olarak çizgisel detayların belirlenmesi" adlı doktora tezinden hazırlanmıştır. Makale metni 18.01.2006 tarihinde dergiye ulaşmış, 21.03.2006 tarihinde basım kararı alınmıştır. Makale ile ilgili tartışmalar 31.08.2007 tarihine kadar dergiye gönderilmelidir.

## Semi-automatic extraction of line features from aerial photographs

### Extended abstract

*Aerial photographs have been evaluated manually by the operators for a long time for the extraction of the vector data. Computer technology and digital image processing technologies have been developed and this development provides to perform these extraction processes automatically or semi-automatically. Automatic feature extraction studies are firstly motivated to carry out the extraction of roads from digital images because roads have characteristic attributes like width, surface type and geometrical shape which can be modelled more easily than the others.*

*The resolution of the images has a very important role in the automatic and semi-automatic extraction of the roads. Most known methods are based on the road tracing and the snakes algorithms.*

*Another method of automatic and semi-automatic feature extraction and classification of images is the image segmentation. In recent years, image segmentation and the front propagation of the segments have been carried out successfully by the Level Set and Fast Marching methods.*

*In this study, a semi-automatic line extraction method, based on the segmentation of the images using color-differences of the pixels and the propagation of fronts by the Level Set algorithms, is developed. An object-oriented application software is also developed to test the capabilities of the developed method.*

*Some semi-automatic feature extraction applications are made by the help of the developed software using a 1:35000 scale black/white aerial photograph for determining the capabilities of this method. Another application with 1:5000 scaled two ortho images which have 0.5m resolution of Ayazağa Campus of İstanbul Technical University. These ortho images are generated from 1:16000 scaled color aerial photographs. In this test area, an accuracy test is also carried out to find the accuracy of the developed method. In this accuracy test, vector data of roads and buildings are collected semi-automatically with the developed software and also manually with an experienced operator. The data collected by the operator are assumed the correct*

*ones and they are compared with the others collected by the software. The accuracy test is carried out in two groups. In the first group, on 422 road check points, measurements are made and the square mean root found as  $\pm 0.663m$ . In the second group, buildings are used and 281 check points are measured and the square mean root of this group is equal to  $\pm 0.463m$ .*

*As the results of the applications and tests, it can be said that the accuracy of this developed method is  $\pm 1$  pixel size of the used imagery. It can be used correctly for producing maps and collecting vector data for GIS. Especially for lakes, rivers and buildings can be collected very efficiently. Different classifications and segmentations, which an operator's can not see, can be made also with the adjusting of the tolerance value. Roads which have good quality can be vectorized from their center lines and/or boundaries according to the scale of the image used.*

*Some weak sides of this developed method and software are also found out. Especially on big scale aerial photographs, the obstacles on the features, as trees, cars and shadows, effects the extraction of the features negatively. Effects of this factor are reduced whether the scale of the image gets smaller. If the tolerance value is not be adjusted to the correct values, wrong features can be extracted. When a big size image is used, the software gives back some errors because the size of the arrays is directly proportional to the number of the pixels. The quality, contrast and noise of the image effect the feature extraction process. The surface attributes of the features also effect the success degree of the feature extraction.*

*If the noise and the contrast of the images are eliminated by the image process algorithms like edge detection algorithms and filters as anisotropic diffusion and the blanks that are generated by the obstacles on the feature can be interpolated by the different kinds of interpolation methods, more good results can be achieved by the developed method and the software. Also, for the image segmentation different types of segmentation like snakes, instead of color difference and for big size images pyramid levels can be used to increase the success degree of this method.*

**Keywords:** *Image segmentation, level set, semi-automatic, digital aerial photograph.*

## Giriş

Harita üretimi ve Coğrafi Bilgi Sistemlerinin (CBS) oluşturulması için gerekli olan vektör veriler, hava fotoğrafları üzerinden, çok uzun zamandır klasik yollarla ve operatörler tarafından elle kıymetlenilmekteydi. Bilgisayar teknolojisi ve dijital görüntü işleme alanlarındaki gelişmeler, günümüzde bu işlemlerin otomatikleşmesine olanak sağlamaktadır. Otomatikleşmenin hedefi hızı arttırmak ve değerlendirme masraflarını azaltmaktır. Bu kapsamda yapılan araştırma çalışmaları, öncelikle, yüzey kaplaması, geometrik şekil, genişlik gibi karakteristik özelliklere sahip olan ve modellenen, yollar gibi detayların, dijital görüntülerden otomatik ve yarı otomatik olarak çizilmesi üzerine yoğunlaşmıştır.

Gerçekleştirilen çalışmalar, otomatik ve yarı otomatik yol çiziminde, kullanılan dijital görüntünün çözünürlüğünün çok önemli bir rolü olduğunu göstermiştir. Ortaya konan yöntemler, kullanılan dijital görüntünün çözünürlüğüne göre; düşük, yüksek ve her ikisinin birlikte kullanıldığı çok çözünürlükte yol çizim yöntemleri olarak sınıflandırılabilir (Laptev, 1997). Bu yöntemler arasında özellikle yol izleme ve Aktif Kontur Modeller (snakes) tabanlı yöntemler öne çıkmaktadır (Kass ve Witkin, 1987).

Otomatik ve yarı otomatik olarak detay çizme ve görüntüler üzerinde sınıflandırma işlemlerinde kullanılan bir başka yöntem de görüntü bölümlenmedir (image segmentation). Görüntü bölümlenme, tıbbi görüntüler üzerinde detay çıkarma ve yorumlama konularında çok sık başvurulan bir yöntemdir (Adalsteinsson ve Sethian, 1995). Son yıllarda, görüntü üzerindeki bölümlenme sonucunda elde edilen yüzeylerin görüntü üzerinde gelişmesini ve yayılmasını sağlamak için Düzey Kümesi ve Hızlı İlerleme (Level Set and Fast Marching) yöntemleri başarıyla kullanılmaktadır (Malladi vd., 1994).

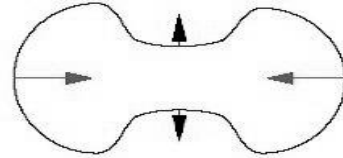
## Düzey kümesi algoritmaları

Dünya üzerindeki tüm nesnelere fiziksel sınırlara sahiptir. Bu sınırları; dinamik ve statik olmak üzere iki ayrı sınıfta değerlendirmek mümkündür. Dinamik sınırlara örnek olarak okyanustaki

dalgaların kırılmasını verebiliriz. Hava fotoğraflarındaki detayların sınırlarını ise statik sınırlara örnek olarak gösterebiliriz. Sınırlarla gösterilen her bir nesne bir yüzeydir. Bu yüzeyler gelişen ya da büyüyen yüzeyler olarak kabul edilirse; bunların gelişme hızları ve yönlerinin çok iyi bilindiği durumlarda bile şekillerinin doğru olarak izlenmesi oldukça zordur. Birinci zor nokta; bir kar tanesindeki kadar karışık şekillerin keskin köşelerinin belirlenmesidir. İkinci zor nokta ise uzak kenarların birbirlerine dolanması ve karışması durumunda ortaya çıkan problemin çözümüdür. Bu probleme örnek olarak, bir orman yangının sınırlarının, birbirinden ayrı alevlerin birlikte yanması ve kıvılcımların rüzgarla birlikte saçılmasıyla değişiklik göstermesini verebiliriz. Üçüncü zor nokta ise, uygun bir şekilde temsil yolu bulunması durumunda bile uç ve daha büyük boyutlu dalgalı bir sınırdan söz etmenin zorluğudur (Sethian, 1997).

Düzey kümesi yöntemleri, keskin köşeli, topolojik değişimlere uğramış ve üç boyutlu gelişen yüzeylerin izlenmesine matematiksel ve programlanabilir araçlar sağlamaktadırlar.

Bir eğrinin karakteristik özelliği, herhangi bir noktadaki eğriliğidir. Eğrilik, eğrinin kıvrılma hızını gösterir.



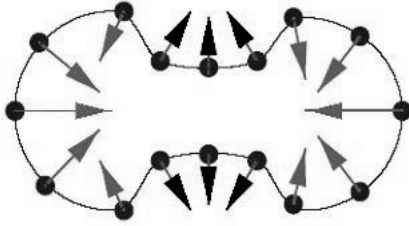
Şekil 1. Eğriliğe bağlı olarak değişen hareketlerin yön ve büyüklükleri (Sethian, 1997)

Eğrinin her bir parçasının, eğrilikle orantılı bir hızda eğriye dik olarak hareket ettiği varsayılabilir. Eğrinin saat yönünde veya saat yönünün tersinde hareket etmesine bağlı olarak eğrilik pozitif ya da negatif değer alabilir. Bunun sonucunda eğrinin bazı bölümleri dışa hareket ederken diğer bölümleri de içeri doğru hareket eder. Şekil 1'deki gibi koyu renkli oklar eğriliğin negatif olduğu bölgeleri, açık renkli oklar ise eğriliğin pozitif olduğu bölgeleri göstermektedir. Oklar farklı uzunlukta oldukları için açık renkli

oklardaki kuvvet büyüklüğü, koyu renklidekilerden daha büyüktür (Sethian, 1997).

Bir yüzey boyunca yürürken geçilen noktaların  $x$  ve  $y$  koordinatlarını,  $xy$  düzleminde bu noktaları çizen bir başka kişi tarafından kaydedildiği varsayalım. Yürüme hızına “parameterizasyon”, diğer kişi tarafından noktaların birleştirilmesiyle çizilen eğriye de “görüntü” adı verilir. Hızlı veya yavaş olarak yürünse de hep aynı sınır çizilecektir. Bu yaklaşımın avantajı, koordinat sistemine olan bağımlılığın ortadan kaldırılmasıdır.

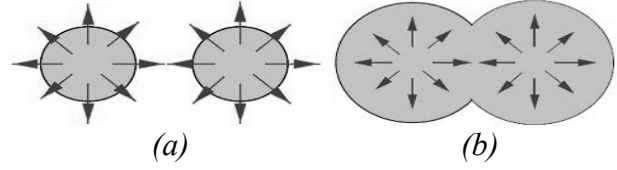
Bir yüzeyin, parameterize edilmiş sunumu bir sayısal algoritmanın omurgasını oluşturacak şekilde kullanılmak istendiğinde eğrinin etrafında düzenli aralıklarla işaretler yerleştirilmeli ve bu işaretler bir iple birbirlerine bağlanmalıdır. İşaretler ve bağlı oldukları ip bir sınır oluşturur (Şekil 2) (Sethian, 1997).



Şekil 2. Birbirine bağlı işaretlerin oluşturduğu eğrinin sınırı (Sethian, 1997)

Okların uzunlukları ve yönleri buldukları noktalardaki eğriliklerle belirlenir. Burada şöyle bir genişleme ya da ilerleme stratejisi uygulanabilir. Okların yönlerine bakarak işaretler ilerletilebilir ve yeni oklar hesaplanabilir ve işaretler yeniden ilerletilebilir. Böylece ne kadar çok işaret kullanılırsa, o kadar doğru bir cevap alınacağı beklentisi içine girilebilir. Ne yazık ki bu yaklaşım, bazıları kaçınılmaz bazıları da düzeltilebilir bir takım hatalar içermektedir. Şekil 2’ye dikkatlice bakıldığında da düzeltilebilecek hatalardan birini görmek mümkündür. İşaretler birbirlerinin üzerine geçmeye çalışacaklar ve geçeceklerdir. Bu durumda işaretleri birbirine bağlayan ipi düzenli bir şekilde bir arada tutmak zorlaşacaktır, ip dolanacak ve işaretler birbirlerine karışacaklardır. Bir çözüm olarak; yüzeyin ilerlemesini periyodik olarak durdurmak ve eğri

boyunca yeniden yürüyerek yeni eşit aralıklı işaretler yerleştirmek verilebilir.

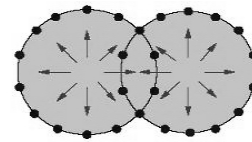


Şekil 3. (a) Başlangıç alevleri (b) Yayılan alevlerin sonraki zamandaki durumu (Sethian, 1997)

Yayılan sınırın topolojisi değiştirilmeye çalışıldığında ise çok daha ciddi bir problem ortaya çıkacaktır. Her ikisinin de sabit hızda dışarıya doğru yandığı ve genişlediği varsayılan iki ayrı dairesel alev ele alındığında yayılan yüzeyin şekli kolayca öngörülebilir (Şekil 3) (Sethian, 1997).

İki ayrı alev birlikte büyüdüğünden, gelişen yüzeyler birleşerek tek bir yayılan yüzey oluşturmaktadırlar. Bununla birlikte; parameterizasyona dayanan bir sayısal algoritma burada büyük bir hataya neden olacaktır. İlerleyen alevin gerçek kenarı izlenmek isteniyorsa iki alevin kesişim bölgesindeki iki çift işaret bir şekilde ortadan kaldırılmalıdır (Şekil 4) (Sethian, 1997).

Düzyer kümesi yaklaşımı bu probleme farklı bir çözüm getirmektedir. Arayüzeyin kendisini izlemek yerine, orijinal arayüzey ele alınmakta ve probleme ilave bir boyut daha eklenmektedir. Arayüzeyin içinde bulunduğu  $xy$  düzlemine ek olarak yüksekliğin ölçüldüğü  $z$  boyutunun da dahil edildiği yeni bir koordinat sistemi oluşturulmaktadır. Problem iki boyutlu bir problem den üç boyutlu bir probleme dönüştürülmüş olmaktadır.

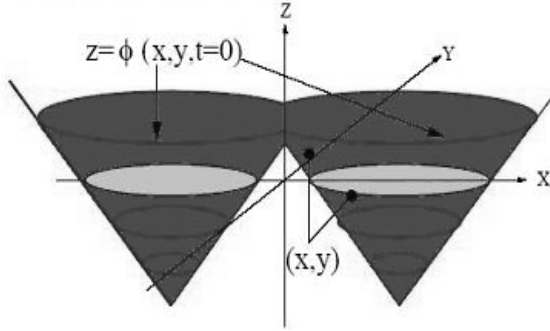


Şekil 4. Yayılan yüzey üzerindeki işaretler (Sethian, 1997)

$z = \phi(x, y, t = 0)$  olan bir fonksiyonu ele alalım. Bu fonksiyonda  $(x, y)$  koordinatlarıyla temsil

edilen bir nokta, girdi verisi olarak alınmakta ve bu noktaya  $z$  yüksekliği atanmaktadır.  $Z$  yüksekliği,  $t=0$  anındaki arayüzey ile  $t$  anındaki arayüzey arasındaki uzunluğu ifade etmektedir. Başka bir deyişle  $t$  anındaki arayüzeyin  $t=0$  anındaki arayüzeyden olan yüksekliği  $z$  değeri ile ifade edilmektedir. Böylece  $xy$  düzlemini arayüzeyde mutlak olarak kesen ve koyu renk ile renklendirilmiş bir yüzey oluşturulmaktadır (Şekil 5) (Sethian, 1997). Koyu renkli yüzey, düzey kümesi fonksiyonu olarak isimlendirilmektedir. Çünkü düzlem içerisindeki herhangi bir noktayı girdi olarak kabul etmekte ve çıktı olarak bir yükseklik vermektedir. Açık renkli arayüzey sıfır düzey kümesi olarak adlandırılmaktadır. Çünkü yüksekliği sıfır olan tüm noktaların birleşiminden oluşmaktadır (Şekil 5) (Sethian, 1997).

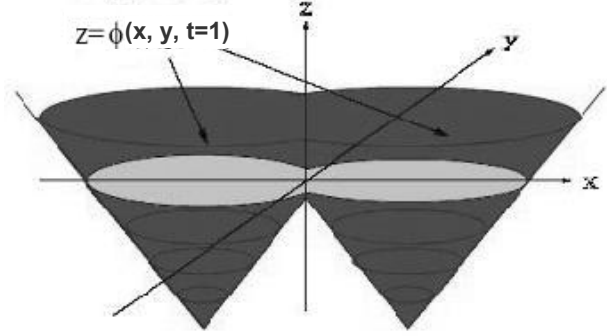
Düzyer Kümesi Fonksiyonu



Şekil 5. Düzyer kümesi yüzeyi her  $(x, y)$  noktasından açık renkli arayüzeye olan uzaklığı çıktı olarak vermektedir (Sethian, 1997)

Arayüzeyin gelişimini zamanında yakalayabilmek için  $\phi(x, y, t)$  yüzeyinin yüksekliğinin nasıl değiştirileceği belirlenmelidir. Düzyer kümesi fonksiyonunun genişlemesine, yükselmesine, alçalmasına izin vermek ve bütün işlemleri yerine getirmesini sağlamak amaç olarak belirlenmektedir. Arayüzeyin herhangi bir zamandaki konumunu bulmak için yüzey sıfır yükseklikte kesilmektedir. Başka bir deyişle sıfır konturu çizilmektedir. İlk bakışta, hareket eden eğrinin problemini, hareket eden yüzeyinkiyle değiştirmek akıllıca görünmeyebilir. Daha çok boyut daha çok iş demektir denebilir. Fakat burada getirilen ek boyut o kadar yararlıdır ki çarpışabilen ve birbirinden uzaklaşabilen işaretler yerine,

şimdi her  $(x, y)$  noktasında durup, düzey kümesi fonksiyonun o noktadaki yüksekliği ayarlanabilecektir. Böylece topolojik hatalar ortadan kaldırılabilir. Yayıldıklarında birleşerek tek bir alev meydana getiren iki ayrı alevin belirli bir zamandaki sıfır düzey kümeleri iki değil sadece bir eğri oluşturacaktır (Şekil 7) (Sethian, 1997).



Şekil 6. Koyu renkli düzey kümesi fonksiyonu hareket ettirilerek yeni açık renkli arayüzey oluşturulur (Sethian, 1997)

İşaretleme yöntemleri, basketboldaki adam adama savunmaya, düzey kümesi yöntemleri ise alan savunmasına benzetilebilir. Düzyer kümesi yaklaşımının üç veya daha büyük boyutlu problemlerin çözümünde kullanılmasının hiçbir farkı yoktur. Daha büyük boyutlu problemlerde görüntülemenin zorlaşması dışında stratejide hiçbir değişiklik yoktur. Dikkat edilmesi gereken konu; hangi boyutta çalışılırsa çalışılsın düzey kümesi fonksiyonu oluşturulurken, ilave olarak bir boyut eklenmektedir.

## Geliştirilen yöntem

Tasarlanan yarı otomatik veri toplama yöntemi düzey kümesi ile görüntü bölümlenme algoritmalarına dayandırılmıştır. Burada çözülmesi gereken üç problemle karşılaşmıştır. Birincisi; algoritmanın nasıl başlatılacağıdır. Bu problem, operatör tarafından çizilmek istenen detayın üzerinde herhangi bir noktanın (pikselin) işaretlenmesi suretiyle çözülmüştür. Böylece düzey kümesi algoritması operatörün seçmiş olduğu noktadan itibaren çalışmaya başlayacaktır.

İkinci problem ise işaretlenen noktadan itibaren hangi kritere göre detayın çizilmesi işleminin ilerleyip ilerlemeyeceğine karar verileceğidir.

Bu problem ise raster görüntülerin meydana geldiği her bir pikselin sahip olduğu renk değerlerinden faydalanılarak çözülmüştür. İşaretlenen noktanın sahip olduğu renk değeri veya belirlenen komşuluk derecesinde, komşu piksellerin renk değerleri ile ortalamasının alınarak bulunan renk değeri, komşu piksellerin renk değerleri ile karşılaştırılmış ve renk farkı belirlenen bir tolerans değeri içinde kalmışsa algoritma devam ettirilmiş, değilse durdurulmuştur. Başka bir deyişle renk farklılığına bağlı bir bölümlenme gerçekleştirilmiştir.

Çizilmesi istenen detay belirlendikten ve işaretleme işleri tamamlandıktan sonra bu detayın bir vektör veri olarak elde edilmesi gerekmektedir ki; bu sayede harita üretiminde veya CBS'nde kullanılabilir. Bu da üçüncü problemi oluşturmaktadır. Bu problemin çözümü için de raster veriden vektör veriye dönüşüm işlemi gerçekleştirilmiştir. Yukarıda bahsedilen üç problemi çözen yarı otomatik veri toplama yöntemi, iş akışı 5 ana adımdan oluşan bir işlemler bütünü şeklinde tasarlanmıştır. Bu işlem adımları:

- Operatör tarafından değerlendirilmek istenen detayın çizilmesi için başlangıç noktasının veya pikselinin seçimi,
- Seçilen görüntü pikselinin, komşu piksellerle olan renk farkından faydalanılarak görüntü bölümlenmesinin gerçekleştirilmesi,
- Düzey kümesi yöntemi kullanılarak bölümlenmenin yayılması ve yığın yapısı (heap sort) şeklinde depolanması,
- Bu yapıdaki piksellerden 1 bitlik (1 renkli) raster maske görüntüsünün elde edilmesi
- Elde edilen raster maskeden koordinatlı olarak, raster görüntüden vektöre dönüştürme işleminin gerçekleştirilmesini müteakiben detaylara ait bilinen bir formatta vektör verilerin elde edilmesidir.

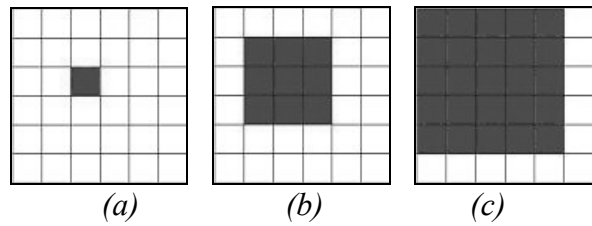
İlk 4 işlem adımı Borland C++ programlama dili kullanılarak ve sistem kütüphaneleri dışında hiçbir kütüphane dosyası kullanılmadan gerçek-

leştirilmiştir (Karagülle ve Pala, 2002). 5'nci adım ise internet ortamında bulunan, bir raster görüntüden vektöre dönüşüm açık kodunun, Visual C++ kullanılarak yeniden düzenlenip, kırıklık toleransı ve köşe koordinatları giriş seçeneklerinin ilave edilerek işlevsellik kazandırılmasıyla gerçekleştirilmiştir.

### Görüntü bölümlenme algoritması

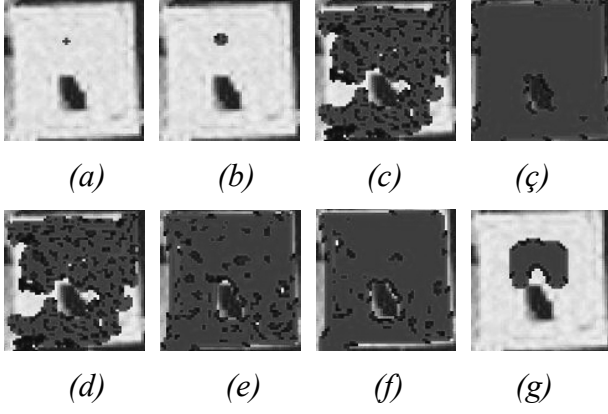
Görüntü bölümlenmesi piksellerdeki renk farkına dayandırılmış olup, bu basit algoritmaya operatör tarafından ayarlanabilir bir tolerans değeri eklenerek esneklik kazandırılmıştır. Böylece operatöre, kontrastlığın fazla olduğu yerlerdeki detaylarda toleransı yüksek tutarak sadece bir kez işaretlemek suretiyle büyük alanları seçebilme kabiliyeti kazandırılması öngörülmüştür. Bölümlenme algoritmasının eşik değeri olarak görüntünün genişliği ve boyunun çarpımının 2 katı kadar bir değer belirlenmiştir.

Bölümlenme, ilk olarak işaretlenen detay noktası temel alınarak başlatılır. Bu detay noktasına denk gelen pikselin ve/veya belirlenen komşuluk derecesine göre komşu piksellerin renk değerleri ile ortalaması alınması sonucunda elde edilen renk değeri referans olarak alınır. Komşuluk derecesi 0 ise, o pikselin renk değeri, komşuluk derecesi 1 ise etrafındaki 8 piksel ile birlikte ortalaması alınması sonucunda bulunan renk değeri başlangıç olarak alınır (Şekil 7).



Şekil 7. Komşuluk derecesi kavramı (a) Komşuluk derecesi 0 (b) Komşuluk derecesi 1 (c) Komşuluk derecesi 2

Burada referans noktasının renk değerleri ile komşu piksellerin renk değerleri arasındaki fark hesaplanırken üç banttaki (kırmızı, mavi ve yeşil) renk farklılıkları ayrı ayrı hesaplanır ve her üçünün de tolerans değerinden küçük olup olmadığı araştırılır.



Şekil 8. Tolerans değeri ve komşu piksel derecesi değerlerinin detay çıkarma işlemlerine etkileri

Tolerans değeri ve komşu piksel derecesi değerlerinin detay çizim işlemine etkileri Şekil 8’de gösterilmiştir. İlk dört şekilde komşuluk derecesi 0 olarak alınmış, başka bir deyişle işaretlenen noktanın renk değerleri referans olarak alınmış olup, diğer piksellerle bir ortalama alma işlemi gerçekleştirilmemiş ve sadece tolerans değerleri sırasıyla 1, 5, 10 ve 25 olarak ayarlanmıştır (Şekil 8 (a), (b), (c), (ç)). İkinci dört şekilde ise tolerans değeri 10 olarak ayarlanmış olup, sabit olarak alınmış ve sadece komşuluk derecesi sırasıyla 1, 2 ve 16 olarak değiştirilmiştir (Şekil 8 (d), (e), (f), (g)). Burada farklılıkların hatasız olarak tespit edilebilmesi amacıyla her şekilde aynı piksel başlangıç pikseli olarak seçilmiştir (Şekil 8 (a)). Bu araştırma göstermiştir ki; komşu piksel derecesinin artırılması her zaman iyi sonuçlar vermemektedir. Özellikle seçilen piksele yakın pikseller değerlendirilmeye katıldığında sonuçlar oldukça iyileşmekte fakat pikselden çok uzaklaşınca aynı sonuçları elde etmek için daha büyük tolerans değerine ihtiyaç duyulmaktadır.

### Düzye kümesi ile yayılma ve depolama

Düzye kümesi algoritması ile yayılmanın yada detay boyunca ilerlemenin gerçekleştirilebilmesi için gerekli olan başlangıç değerine, geliştirilen yöntemin yarı otomatik doğasıyla birlikte çözüm bulunmuştur. Yüzeyin yayılmasına, operatör tarafından işaretlenen pikselden itibaren başlanacaktır ve sıfır düzey kümesi fonksiyonu bu pikselin konumu ile tanımlanacaktır.

Düzye kümesi fonksiyonu için gerekli olan diğer bir bileşen ise yayılmanın kontrolünü sağlayan sınır değeridir. Sınır değeri için çözüm yukarıda anlatılan renk farkının hesaplanmasıyla gerçekleştirilmiştir.

Geriye, yayılmanın başlatıldıktan sonra, sınır değerini sağlayan komşu piksellerden hangisinden devam edeceği probleminin çözümü kalmıştır. Aslında bu problemin çözümünde, “Hızlı İlerleme” (Fast-Marching) yönteminden faydalanılabilir. Fonksiyon, en küçük değere sahip pikselden itibaren yayılmaya devam edebilir (Sethian, 1998). Fakat, en küçük değer neyi ifade edecektir?

Geliştirilen yöntemde, en küçük değer neyi temsil etmelidir sorusu için operatör tarafından işaretlenen ilk piksel sıfır düzey kümesi fonksiyonu olarak kabul edilmiş ve her komşu piksel (doğu, batı, kuzey, güney yönündeki ilk pikseller) yukarıda açıklanan renk farkı algoritmasıyla kontrol edilmiş ve şartı sağlayan pikseller için sıfır düzey kümesinden (ilk işaretlenen piksel) olan mesafeler hesaplanmıştır ve böylece en küçük mesafeye sahip pikselden itibaren yayılmanın devam ettirilmesi sağlanmıştır.

Burada kendimize şu soruyu sorabiliriz: Mesafeler eşitse durum ne olacaktır? Bu başlangıçtan sonraki ilk adımda ortaya çıkabilecek bir problemdir çünkü başlangıçta dört komşu piksel vardır ve şartı sağlayan pikseller aynı uzaklığa (sadece bir birim uzunluğa) sahiptirler. Böyle bir durumda, en küçük renk farkına sahip pikselden itibaren yayılma devam ettirilerek problemle adil bir çözüm getirilmiştir.

Yayılmanın tamamlanabilmesi için güncelleme işleminin yapılması gerekmektedir (Sethian, 1998). Güncelleme işleminde, mesafenin hesaplanabilmesi için gerekli olan, kareselleştirme algoritmaları kullanılmıştır (Sethian, 1998).

İşaretlenen piksellerin depolanması ve erişimi için minimum yığın yapısı (heap sort) kullanılmıştır. Minimum yığın yapısında, en küçük mesafeye sahip görüntü hücresi en tepede olacak şekilde bir yapılanma sağlanmıştır. Yığın yapısının korunması için bazı fonksiyonlara ihtiyaç

duyulmaktadır. Yığına görüntü hücresinin eklenmesi, görüntü hücresinin yığında sabitlenmesi, en küçük mesafeye sahip görüntü hücresinin yığından çıkartılması ve yığın her yeni görüntü hücresi eklendiğinde veya çıkartıldığında yapısının yeniden güncellenmesi gerekmektedir.

Bu depolama yapısı sayesinde görüntü hücrelerine ulaşılması, yayılmanın test edilmesi ve hesaplanması ile görüntü hücrelerinin işaretlenmesi ve işaretlenen görüntü hücrelerinin depolanması gibi büyük hacimli işlemlerde hesaplama etkinliğinin artırılması amaçlanmıştır.

### Rasterdan vektöre dönüştürme

Seçilen detayın yarı otomatik olarak ekran üzerinde işaretlenmesinden sonra bu verilerin harita üretiminde, Coğrafi Bilgi Sistemi (CBS) için gerekli olan coğrafi veri tabanlarının oluşturulmasında kullanılabilmesi ve bu verilere anlam kazandırılabilmesi için vektör veri haline dönüştürülmesi gerekmektedir. Vektör veri yapısında coğrafi varlıklar nokta, çizgi veya alanlar olarak temsil edilirler. Bu modelde bir coğrafi varlık nokta olarak temsil edilecekse sadece bir koordinat çifti, çizgi şeklinde temsil edilecekse koordinat çiftlerinden oluşan bir küme, alan şeklinde temsil edilecekse başlangıç ve son koordinatları aynı olan koordinat çiftleri kümesi kullanılır.

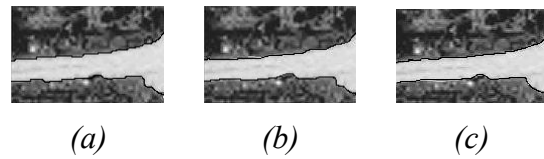
Rasterdan nokta detaya dönüşüm yapılırken, her bir piksele karşılık bir nokta detay oluşturulur. Çizgi detaya dönüşüm yapılırken, birbirine komşu sıralı pikselleri çevreleyen alanın merkezinden geçen çizgi elde edilir. Raster veri içerisinde aynı değere sahip birbirine komşu pikseller topluluğu bir alan oluştururlar. Bu alanın dış sınırı çevrilererek, raster veri alan tipindeki vektör detaya dönüştürülür (ESRI, 1997).

Rasterdan vektöre dönüşüm algoritmaları sadece iki tip veri (0 veya 1) içeren 1 bitlik görüntü dosyaları için geçerlidir. Bu yüzden geliştirilen yazılımda, görüntü hücrelerinin seçimi ve işaretlenmesi işlemleri tamamlandıktan sonra, işaretli pikseller dolu, diğerleri ise boş olacak şekilde bir görüntü dizisi oluşturulmaktadır. Bu görüntü dizisinin oluşturulması bir çeşit maskeleyme iş-

lemidir. Elde edilen maske raster görüntü daha sonra 1 bitlik (tek renkli) BMP formatında raster görüntü şeklinde kaydedilmektedir.

Elde edilen maske görüntü dosyasının, vektör veriye dönüştürülmesi amacıyla Davide Libenzi tarafından geliştirilen ve kendi internet sitesinde açık olarak bulunan Visual C++ kodu yeniden düzenlenmiş ve ilave imkanlar kazandırılmıştır. Ana programla bağlantısı kurularak, etkileşimli bir arayüzde raster veriden, detayların merkez hatlarının ve sınır hatlarının ayrı ayrı vektöre dönüştürülmesi sağlanmış, ana program üzerinde sol alt köşe koordinatları ile her iki boyutta (x,y) görüntü çözünürlüklerinin girilmesi suretiyle koordinatlı bir vektör veri elde edilmesi sağlanmıştır. Ayrıca etkileşimli arayüz üzerinde kırıklık toleransının girilmesiyle istenilen yumuşaklıkta ve kırıklıkta vektör veri elde etme olanağı sağlanmıştır. Kırıklık toleransı ile rasterdan vektöre dönüşüm sonucunda elde edilen vektörün kırıklığı ayarlanmaktadır.

Kırıklık toleransı 0 olduğunda hiç yumuşatma yapılmadan tüm pikseller hesaplamaya dahil edilir, kırıklık toleransı arttırıldıkça tüm pikseller değil, artan aralıklarla pikseller dikkate alınır ve elde edilen vektör daha düz bir hale gelir fakat kırıklık toleransının çok fazla arttırılması durumunda geometrik doğruluğun bozulması söz konusudur (Şekil 9 (a), (b), (c)).



Şekil 9. Kırıklık toleransının vektör veriye dönüştürme işlemine etkisi (a) Kırıklık toleransı=0 (b) Kırıklık toleransı=1 (c) Kırıklık toleransı=2

### Gerçekleştirilen testler

Hava fotoğraflarından yarı otomatik olarak çizgisel detayların belirlenebilmesi hedefi doğrultusunda geliştirilen ve önceki bölümlerde detaylarıyla açıklanan yazılımın, kullanılabilirliğinin, etkinliğinin ve geometrik hata aralıklarının belirlenebilmesi amacıyla iki grup test çalışması gerçekleştirilmiştir.

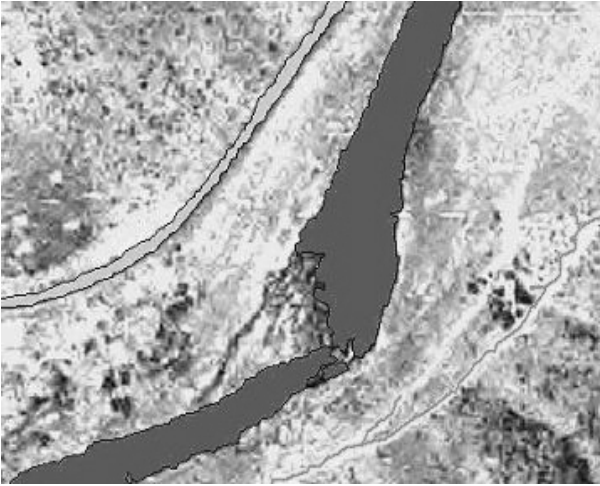


Birinci grup test çalışmasında 1:35000 ölçekli siyah/beyaz dijital hava fotoğrafı kullanılmıştır.

İkinci grup çalışmada ise İstanbul Teknik Üniversitesi, Ayazağa Kampüsü'nün bir bölümünü içeren 1:16 000 ölçekli renkli hava fotoğraflarından oluşturulan 0.5m çözünürlüklü 1:5000 ölçekli iki adet ortofoto görüntü kullanılarak geometrik hata kriterleri belirlenmeye çalışılmıştır.

### **Test 1**

1:35000 ölçekli, 20 mikronda taranarak sayısallaştırılmış siyah/beyaz hava fotoğrafındaki 3 ayrı çizgisel detayın çıkarılmasının araştırıldığı bu test çalışmasında, yüzeyi asfalt kaplı bir yol, sulu geniş bir dere ve bir patika sayısallaştırmaya çalışılmıştır. Asfalt yol ile sulu dere sınırlarından alan şeklinde, patika ise merkez hattından çizgisel olarak vektöre dönüştürülmüştür (Şekil 10). Asfalt yol ve sulu derenin işaretlenmesi sırasında sadece birkaç noktada işaretleme yapılmışken, patikanın işaretlenmesi sırasında tolerans değeri küçük tutulmuş ve küçük küçük adımlarla ilerlenmiştir çünkü patikanın arazi dokusundan ayırt edilmesi oldukça zordur.



*Şekil 10. 1:35000 ölçekli hava fotoğrafından elde edilen vektör veriler*

Bu çalışmada, fotoğraf ölçeğinin küçülmesiyle çizgisel detayları örten engellerin (ağaç, araba vb.) etkilerinin azaldığı tespit edilmiştir. Ölçekle birlikte bu engellerin yer aldığı piksel sayısı azaldığından kapsadıkları alanda azalmaktadır.

Böylece engellerden kaynaklanan boşluk miktarları daha küçük değerler almaktadır.

### **Test 2**

İstanbul Teknik Üniversitesi, Ayazağa Kampüsü'nün bir bölümünü içeren 1:16 000 ölçekli renkli hava fotoğraflarından oluşturulmuş 0.5 m çözünürlüğe sahip, iki adet 1:5000 ölçekli ortofoto görüntüden koordinatlı olarak yol ve bina detayları geliştirilen yazılımla yarı otomatik olarak çizilmiştir.

Elde edilen verilerin geometrik doğruluklarının tespit edilmesi amacıyla aynı detaylar bir uzman operatör yardımıyla tamamen elle çizilmiş ve bu veriler doğru olarak kabul edilerek, yarı otomatik toplanan verilerle karşılaştırılarak bir doğruluk araştırması yapılarak geliştirilen yöntemin hata kriterleri belirlenmeye çalışılmıştır. Yarı otomatik veri toplama işlemine başlamadan önce TIFF formatındaki ortofoto görüntüler JPEG görüntü formatına dönüştürülmüştür çünkü geliştirilen yazılım TIFF formatındaki görüntüleri okuyamamaktadır. Görüntüler üzerinde öncelikle yol detayları çıkarılmaya çalışılmıştır. Yol detaylarının çıkartılması tamamlandıktan sonra elde edilen maske görüntü yol.BMP dosyası olarak kaydedilmiştir. Daha sonra bina detayları üzerinde durulmuş ve aynı yöntemle bina detaylarında çıkartıldıktan sonra elde edilen maske görüntü bina.BMP dosyası adıyla kaydedilmiştir.

Çeşitli tolerans değerleri uygulanarak çıkartılan ve maske raster görüntüler şeklinde kaydedilen görüntüler, ana arayüzde orto görüntülerin sol alt köşe koordinatlarının girilmesi ve çözünürlüklerin 0.5m olarak ayarlanmasından sonra vektör işlemleri penceresinde kırıklık toleransı 0 olarak ayarlanıp ayrı ayrı yol.DXF ve bina.DXF dosyaları şeklinde vektöre dönüştürülmüşlerdir.

Kırıklık toleransının 0 olarak ayarlanmasının nedeni, bu çalışma sonucunda yöntemin hata kriterlerinin belirleneceğinden kırıklık toleransından kaynaklanacak hataların bu kriterleri etkilemesinin önüne geçmektir. Yarı otomatik detay çıkarma işlemi yaklaşık yarım gün sürmüştür ve bu veriler kesinlikle hiçbir şekilde düzel-

tilmemiş olup ham hallerinde doğruluk araştırmasına tabi tutulmuştur.

Yarı otomatik olarak detay çıkarma sırasında karşılaşılan en büyük sorun, bina çatılarının homojen olmayıp, bina çatılarında farklı yapılaşmaların olması ve resim ölçeğinin büyük olmasından dolayı yolların ağaç, otomobil gibi engeller tarafından kapatılmasıdır.

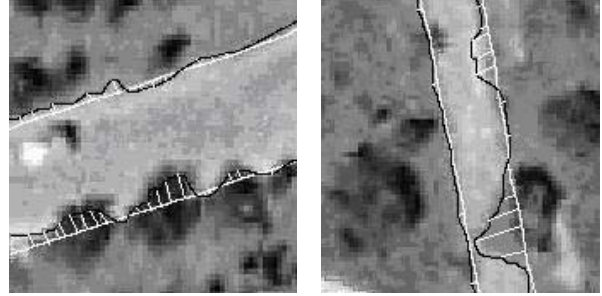
Yarı otomatik olarak çıkarılan detaylar, uzman bir fotogrametri operatörü tarafından yaklaşık yarım gün içinde tekrar kıymetlendirilmiştir Her iki veri toplama işlemide yaklaşık aynı süreler içerisinde gerçekleştirilebilmiştir. Doğruluk araştırması amacıyla, uzman operatör tarafından gerçekleştirilen kıymetlendirme sonuçları referans olarak alınmış ve yarı otomatik yöntem ile elde edilen vektör veriler karşılaştırılmıştır (Şekil 11). Bu karşılaştırma bina ve yol detayı olmak üzere iki ayrı grupta yapılmıştır.



Şekil 11. Yarı otomatik olarak yazılımla elde edilen vektör veriler (koyu renkli) ile uzman operatör tarafından çizilmiş vektör veriler (açık renkli)

Doğruluk araştırması amacıyla, her iki vektör veri ArcGIS programında üst üste açılmış ve detaylar üzerinde belirlenen noktalarda hızları gösteren dik çizgiler çizilmiş ve bu çizgilerin uzunlukları ölçülmüştür. Aslında bu çizgiler hata vektörlerini temsil etmektedirler. Yol detaylarına ilişkin 422 noktada, bina detaylarına ilişkin 281 noktada ölçüm yapılmıştır. Yapılan ölçümlere ve çizilen hata vektörlerine ilişkin örnekler Şekil 12’de gösterilmiştir.

Burada koyu renkli çizgiler yarı otomatik yöntemle elde edilen vektör verileri, açık renkli çizgiler ise uzman operatör tarafından kıymetlendirilen vektör verileri göstermektedir. Hata vektörleride açık renkli ve dik çizgilerle gösterilmiştir.



Şekil 12. Gerçekleştirilen hata ölçümleri

Hata vektörlerinin boylarının ölçülmesi sonucunda elde edilen hata miktarlarının her iki grup (yol ve bina) için ayrı ayrı ortalama ve karesel ortalama hataları hesaplanmıştır. Sonuçlar, Tablo 1’de gösterilmiştir.

Tablo 1. Hesaplanan ortalama ve karesel ortalama hata miktarları

	Ortalama Hata	Karesel Ortalama Hata	Minimum – Maksimum Hata
<b>Yollar (422 nokta)</b>	0.930 m	±0.663 m	0.099 m - 4.183 m
<b>Binalar (281 nokta)</b>	0.710 m	±0.463 m	0.031 m - 2.292 m

Karesel ortalama hata miktarlarına baktığımızda yaklaşık ±0.5m civarında olduğu görülmektedir. Bu da kullanılan orto görüntülerin çözünürlüğüne denk düşmektedir. Başka bir deyişle bu yöntemin karesel ortalama hatasının, kullanılan raster görüntünün bir pikselinin boyutuna eşit olduğu söylenebilir. Buradan şu sonuca varılabilir: Geliştirilen yöntemin hata kriteri kullanılan dijital hava fotoğrafının ±1 pikselinin boyutuyla sınırlıdır. Elde edilen sonuçlar ışığında, binalarda yollara göre daha iyi sonuçlar elde edildiği görülmüştür. Yollarda, yol kenarlarındaki ağaçlar ve gölgelerin, resim ölçeğinin büyük olmasından dolayı, hataların artmasına neden olduğu görülmüştür. Binaların kenarlarının daha düz-

gün olması nedeniyle daha düzgün vektörler elde edilmiştir ve bunun sonucunda hataların daha küçük olduğu gözlenmiştir.

Bu test araştırması sonucunda; geliştirilen yarı otomatik yöntemle elde edilen vektör veriler bir operatör tarafından editlenerek düzeltilirse hata miktarlarının daha da düşürülebileceği fakat editlemeyle birlikte yeni bir iş gücü ve ilave zaman gerekeceğinden maliyetlerin de artacağı söylenebilir.

## **Sonuçlar**

Bu çalışmanın amaç ve hedeflerine paralel olarak, renk farkları ile düzey kümesi yöntemlerini baz alarak geliştirilen yarı otomatik düzey kümesiyle görüntü bölümlenmesi ve buna bağlı olarak çalışan rasterdan vektöre dönüştürme yazılımlarının uygulanabilirliğinin, işlevselliğinin ve etkinliğinin test edilmesi amacıyla farklı ölçek ve türlerdeki hava fotoğrafları ile uydu görüntüleri üzerinde yarı otomatik vektör toplama işlemleri ve geliştirilen yöntemin hata kriterlerinin belirlenebilmesi için bir doğruluk araştırması gerçekleştirilmiştir.

Yapılan doğruluk araştırması sonucunda, geliştirilen yöntemin, kullanılan dijital hava fotoğrafının, uydu görüntüsünün veya raster görüntünün  $\pm 1$  pikselinin boyutuna eşit olan bir hata kriterine sahip olduğu bulunmuştur.

Bununla birlikte; bu yöntemin fotogrametrik harita üretiminde ve CBS için fotogrametrik veri sağlanmasında yeni bir yöntem olarak kullanılabilirliği düşünülmektedir. Yapılan test çalışmaları sonucunda bu yöntemle veri toplama işlemlerinde diğer yöntemlere göre bazı avantajlar elde edilebileceği görülmüştür:

- Bu yöntem, özellikle göller, sulu dereler ve binalar gibi homojen yapıdaki detayların sınırlarına ait vektör verilerin toplanmasında çok başarılı ve etkili bir şekilde kullanılabilir.
- İstenildiği takdirde, tolerans değerinin ayarlanması suretiyle, söz konusu detaylar üzerinde gözle ayırt edilemeyen sınıflandırmalar ve bölümlenmeler gerçekleştirilebilecektir.

- Kaliteli yolların sınırları ve/veya merkez hatları (kullanılan fotoğrafın ölçeğine bağlı olarak) etkili ve hızlı bir şekilde çıkartılabilecektir.

- Kırıklık toleransı değerleri ayarlanarak istenilen kırıklıkta vektör veriler elde edilebilecektir.

- Rasterdan vektöre dönüşümde hem sınırların hemde merkez hatların kullanılabilmesi etkinliğe çok katkı sağlayacaktır.

Bu avantajlarla birlikte programın hiçbir kütüphane veya paket program dosyası kullanmaması ve başka hiçbir yazılıma bağlı olmaması, yazılımın pratikliğini ve kullanılabilirliğini arttırdığı düşünülmektedir.

Test çalışmaları sırasında yazılımın olumsuz olarak etkilendiği ve eksik kaldığı bazı faktörler de tespit edilmiş olup, bunlarda aşağıda sunulmuştur:

- Özellikle büyük ölçekli görüntülerde detaylar üzerinde bulunan engeller (ağaç, araba, gölge gibi) detay çıkarma işlemini olumsuz olarak etkilemektedir. Bu etki ölçek küçüldükçe azalmaktadır.

- Tolerans değerleri uygun olarak ayarlanmadığı takdirde yanlış detay çıkarımları gerçekleştirilmektedir. Tolerans yüksek tutulduğunda ilgilenilen detaylar atlanabilmekte, küçük tutulduğunda ise çok fazla data vererek küçük küçük adımlarla ilerleme sağlandığından yazılımın etkinliği düşürülmektedir.

- Çok büyük boyutlu görüntü dosyaları kullanıldığında, tasarlanan algoritmada piksellerin değerlerinin bilgisayar hafızasına kayıt edildiğinden, çok fazla hafızaya ihtiyaç duyulacağından bazı donanımsal hatalarla karşılaşabilmektedir.

- Görüntülerin kalitesi, kontrastlığı ve gürültü oranları algoritmanın başarısını önemli ölçüde etkilemektedir.

- Çizgisel detayların yüzey ve kaplama özellikleri ile kontrastlıkları da algoritmanın başarısını etkilemektedir.

Söz konusu problemlerin giderilmesi ve yazılımın daha etkin bir hale getirilebilmesi için; görüntülerdeki kontrastlığın artırılması, gürültü oranlarının azaltılması için görüntü işleme algoritmalarından faydalanılarak anisotropik difüzyon gibi filtreler ile kenar zenginleştirme algoritmalarının uygulanmasının, engellerden kaynaklanan boşlukların doldurulması için farklı enterpolasyon yöntemlerinin kullanılmasının, bölümlenme işlemlerinde renk farkı yerine daha gelişmiş, aktif kontur modeller (snakes) gibi, algoritmaların kullanılmasının ve büyük boyutlu görüntülerin bilgisayar ortamlarında daha kolay ele alınabilmesi için piramit seviyelerinin kullanılmasının faydalı olacağı değerlendirilmektedir.

Son olarak; harita üretiminde ve CBS için fotogrametrik olarak veri toplama işlemlerinde, tam otomatik yöntemlerin başarısının hala uzak görünmesinden dolayı, yarı otomatik çözümlere daha fazla ağırlık verilmesiyle ve bu konulardaki çalışmaların motive edilmesiyle, operatörün zekası ve değerlendirme yeteneğiyle birlikte bilgisayarların hesaplama hızı birleştirilerek önemli ilerlemelerin katedilebileceği değerlendirilmektedir.

## Kaynaklar

- Adalsteinsson, D. ve Sethian, J.A., (1995). Fast level set method for propagating interfaces, *Journal of Computational Physics*, **118**, 269-277.
- ESRI, (1997). *ARC/INFO User's Guide Cell-Based Modelling With GRID*, Redlands, USA.
- Karagülle, İ. ve Pala, Z., (2002). Borland C++ Builder 6, Türkmen Kitabevi, İstanbul.
- Kass, M., Witkin, A., (1987). Snakes: active contour models, *International Journal of Computer Vision*, 1, 4, 321-331.
- Laptev, I., (1997). Road extraction based on line extraction and snakes, *Master Thesis*, Kungliga Tekniska Högskolan (The Royal Institute of Technology), Stockholm, Sweden.
- Malladi, R., Sethian, J.A. ve Vemuri, B.C., (1994). Evolutionary fronts for topology-independent shape modeling and recovery, *Proceedings of Third European Conference on Computer Vision, Stockholm, Sweden, Lecture Notes in Computer Science*, **800**, 3-13.
- Sethian, J.A., (1997). Level set methods: An act of violence, *American Scientist*, 85, 3, 12-35.
- Sethian, J.A., (1998). Fast marching methods and level set methods for propagating interfaces, *von Karman Institute Lecture Series*, Computational Fluid Mechanics, Belgium.

---

<http://www.xmailserver.org/davide.html>