

RASTER GÖRÜNTÜLERİN VEKTÖRİZASYONU VE JEODEZİ-FOTOGRAMETRİ MÜHENDİSLİĞİNDEKİ ÖNEMİ

H. Karabörk¹, M.Gündüz², İ.Ö.Bildirici¹, F.Yıldız¹

¹Selçuk Üniversitesi, Jeodezi ve Fotogrametri Mühendisliği Bölümü, Konya, karabork@selcuk.edu.tr, bildirici@selcuk.edu.tr, fyildiz@selcuk.edu.tr

²Selçuk Üniversitesi, Bilgisayar Mühendisliği Bölümü, Konya, mgunduz@selcuk.edu.tr

ÖZET

Vektörizasyon ya da diğer adıyla rasterden vektöre dönüşüm, mühendislik çizimlerini analog biçimden iki veya üç boyutlu bilgisayar destekli tasarım modellerine dönüştürmede önemli rol oynar. Çok sayıda vektörizasyon yöntemi ve ticari yazılım geliştirilmesine rağmen vektörizasyon işlemini çözülmüş bir problem olarak görmek için henüz erkendir. Bu çalışmada vektörizasyon yöntemleri, analog haritalardan mekansal veri kazanımında vektörizasyonun önemi ve seçilen vektörizasyon yazılımları ile elde edilen sonuçlar tartışılmaktadır. Vektörizasyon metodlarından Hough dönüşüm tabanlı metotlar, inceltme tabanlı metotlar, şekil tabanlı metotlar, grafik yürütme tabanlı metotlar, ağ desen tabanlı metotlar ve ayrık piksel tabanlı metotlar ele alınmıştır.

Anahtar Sözcükler: Vektörizasyon metotları, analog harita, mekansal veri kazanımı

ABSTRACT

RASTER TO VECTOR TRANSFORMATION AND ITS IMPORTANCE IN MAPPING

Vectorization, transforming raster images into vector drawings, plays an important role for converting analog engineering drawings to the 2D or 3D CAD data. Although many algorithms and software products have been developed, vectorization should not be considered being entirely solved. There are still problems. In this study, following topics are discussed: Vectorization methods, the importance of vectorization in spatial data capture from analog maps, and the results of some software products. Six methods are selected for discussion: Hough transformation based methods, thinning based methods, contour based methods, run graph based methods, mesh pattern based methods and sparse pixel based method.

Keywords: Vectorization methods, analog map, spatial data capture

1. GİRİŞ

Vektörizasyon, diğer bir adıyla rasterden vektöre dönüşüm, raster görüntüden vektörleri bulma işlemidir. Görüntü işleme teknikleri kapsamında günümüze birçok vektörizasyon tekniği geliştirilmiştir. Raster görüntülerin vektörizasyonu, doküman ve grafik çizimlerin analizi ve tanınması için yeni olmayan bir konu olmasına rağmen halen mükemmel olmaktan uzaktır ve gelişmeye devam etmekte olan bir konudur (Tombre, 1998). Geleneksel vektörizasyon metotları, bütün mühendislik çizimlerine uygulanmakla birlikte her bir mühendislik alanının kendine has çizim kuralları vardır. Bunları dikkate almadan gerçekleştirilen bir vektörizasyon işlemi, ilgili alanda yeterince başarılı olmayacaktır. Jeodezi ve Fotogrametri Mühendisliği uygulamalarında daha fazla otomasyon gerekliliği, veri tabanı üretimi ve güncellemesi gibi işlemlerin önemini artırmıştır. Coğrafi Bilgi Sistemlerinin en önemli bileşeni coğrafi verilerdir. Coğrafi veriler, genellikle hava ile uydu görüntülerinden ve analog haritalardan elde edilir. Bu verilerin Coğrafi Bilgi Sistemlerine hızlı bir şekilde aktarılması önemli bir konudur. Analog haritalardan veri kazanımında elle sayısallaştırmaya alternatif olarak yarı otomatik ya da otomatik vektörizasyon uygulanabilir. Bu çalışmada vektörizasyon algoritmalarından hough dönüşüm tabanlı metotlar, inceltme tabanlı metotlar, şekil tabanlı metotlar, grafik yürütme tabanlı metotlar, ağ desen tabanlı metotlar ve ayrık piksel tabanlı metotlar tanıtılmıştır. Ayrıca vektörizasyon işleminin Jeodezi ve Fotogrametri Mühendisliğinde önemi vurgulanmış ve bu amaçla geliştirilen yazılımlar incelenmiştir.

2. VEKTÖRİZASYON METOTLARI

2.1 Hough Dönüşüm Tabanlı Metotlar

Hough Dönüşüm Tabanlı Metotlarda (Hough transformation based methods), ikili görüntüdeki (siyah-beyaz) uzaysal uzatılmış desenler daha kısa parametre uzayına dönüştürmek suretiyle vektörizasyon gerçekleştirilmektedir. Bu dönüşüm zor olan görüntü uzayındaki tespit problemini, parametre uzayında daha kolay çözülen yerel doruk tespit problemine dönüştürmektedir. Hough Dönüşüm Tabanlı Metotların düz çizgileri tespit etmesinin bir yolu, çizgiyi eğimi ve kesişim noktalarına göre parametrize etmektir. Düz çizgiler aşağıdaki eşitlikte tanımlanmaktadır.

$$y = mx + c \quad (1)$$

(x,y) koordinat eksenindeki her çizgi, (m,c) ekseninde bir noktaya karşılık gelmektedir. (x,y) eksenindeki bir noktadan ise sonsuz sayıda doğru geçebilmektedir. Bu doğruların eğimleri ve sonlandırılma noktaları (m,c) düzleminde bir çizgi oluşturmaktadır. Bu çizgi ise 2 eşitliği ile tanımlanmaktadır.

$$c = -xm + y \quad (2)$$

(m,c) düzlemi, dikdörtgen biçiminde kutucuklara ayrılmıştır. Bu kutucuklar ikinci eşitlikteki doğruya yer alan (x,y) düzlemindeki her siyah pikseli toplamaktadır. Her siyah piksel için ikinci eşitlik çizildikten sonra, bu doğrunun geçtiği hücreler artırılmaktadır. Görüntü uzayındaki tüm pikseller bu şekilde hesaplandıktan sonra, dönüşüm uzayındaki doruk noktaları, çizgileri ifade etmektedir. Gürültü dikkate alınır, belirlenen bir eşik değerinin üzerindeki her bir doruk eşitlik 1 de tanımlanan bir çizgiyi oluşturmak için kullanılmaktadır. Uygulamada bu çizgi aynı doğrultuda olan pek çok çizginin bileşkesi olabilmektedir. Bundan dolayı orijinal görüntüdeki varsayılan çizgi pikselleri takip edilerek bu parçaların son noktaları bulunur. Çizgi kalınlığı da aynı zamanda çizgi izleme işlemi sırasında her bir pikseldeki kalınlık kontrol edilerek belirlenmektedir. m ve c değerleri orijinal görüntüde kırık ve gürültülü çizgilere ait olan noktalar için (m, c) düzleminde doruklar oluşması beklendiğinden, Hough Dönüşüm Metodu gürültülü görüntülerdeki çizgileri de tespit edebilmektedir (Liu ve Dori, 1999).

Bu algoritmanın en basit versiyonu çizgileri tanımasına rağmen, daha karmaşık şekillere uyarlanabilir. Yöntem, her bir piksel üzerinde en az bir kez işlem yaptığı için dikkate değer bir hesaplama zamanını gerektirir.

2.2 İncelme Tabanlı Metotlar

İncelme Tabanlı Metotlar (thinning based methods), kenar izlemeden önce, bir piksel kalınlığında iskelet çıkarmak ve orta eksen noktalarını bulmak amacıyla kullanılmaktadır. Literatürde, iskeletleştirme, çekirdek çizgi tespiti, orta eksen dönüşümü veya simetrik eksen dönüşümü olarak da ifade edilmektedir. Bu yöntemde girdi bir görüntü, çıktı ise morfolojik işlemler uygulanmış girdi görüntüden elde edilen siyah piksellerden oluşan iskelettir. Bir siyah alanın iskeleti topolojik olarak orijinal görüntünün eşdeğeri fakat miktar olarak en küçük halidir. Bu nedenle orijinal görüntüye oranla analiz edilmesi ve kullanılması daha kolaydır. İncelme algoritmaları; yinelemeli sınır erozyonu, uzaklık dönüşümü ve uygun iskelet olmak üzere üç ana grupta toplanabilir.

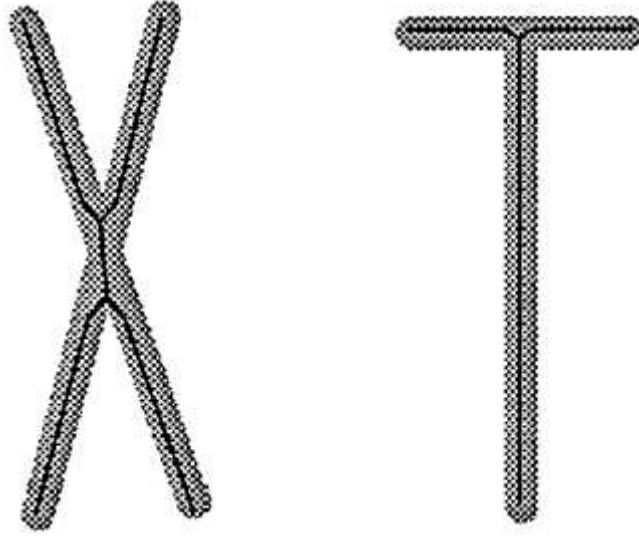
Yinelemeli sınır erozyonu, orta eksen ya da iskelet kalana kadar, sınır piksellerini silme düşüncesine dayanmaktadır. Bu yöntemde esas işlem, 3x3'lük bir pencereyi görüntü üzerinde gezdirmek ve pencerenin merkezindeki pikseli işaretlemek için bir takım kurallar uygulamaktır. Her taramanın sonunda, işaretlenmiş olan pikseller silinmektedir. Taramalar, silinecek hiç piksel kalmayana kadar devam etmektedir. 3x3'lük pencerede piksellerin kodlanması Şekil 1'de gösterilmiştir. Eğer aşağıdaki kuralların hepsi sağlanırsa, P noktası silinmek için işaretlenmektedir.

P ₃	P ₂	P ₁
P ₄	P	P ₀
P ₅	P ₆	P ₇

Şekil 1: P pikseli ve onun 3x3 komşuluğu

- P'nin 4 komşusundan en az biri beyaz olmalıdır (Ör: P_{2i}, i = 0...3). Örneğin P'nin bir kenar pikseli olması durumu.
- P'nin 8 komşusu içinde en az iki tanesinin siyah piksel olması gerekir (Ör: P_i, i = 0...7). Örneğin P, bir çizginin son pikseli olmamalıdır.
- 8 siyah komşudan en az biri işaretlenmemiş olmalıdır.
- P, bir kırılma noktası olmamalıdır.
- Eğer P₂ işaretlenmişse P₂'nin beyaz yapılması, P'yi bir kırılma noktası yapmamalıdır.
- Eğer P₄ işaretlenmişse P₄'nin beyaz yapılması, P'yi bir kırılma noktası yapmamalıdır.

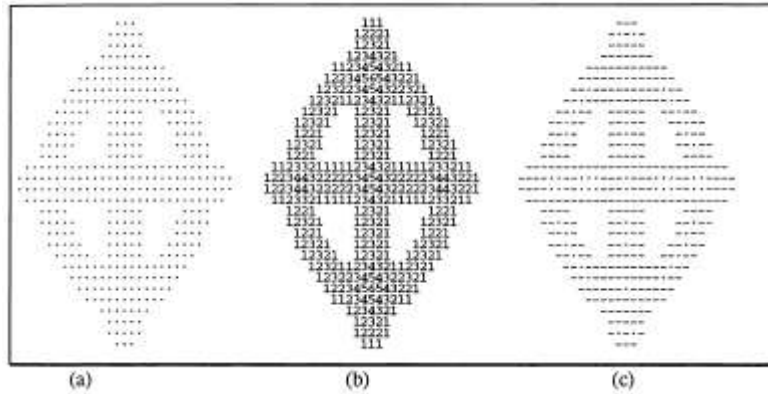
Bu tür algoritmalar, Şekil 2'de görüldüğü gibi, X ve T gibi kesişimleri bozabilmektedir. Her durumda, kesin doğrulukla çalışacak diye bir kesinlik yoktur. Yinelemeli sınır erozyonu için geliştirilen daha iyi yöntemler, pencere boyutunu değiştirmeyi veya işaretleme kurallarını iyileştirmeyi içermektedir. Örneğin kimi algoritmalar kare olmayan pencereler kullanırken, kimi algoritmalar k x k boyutunda pencereler kullanılmaktadır. Fakat bu düzenlemeler, güvenlik ve hız açısından çok az ilerleme sağlamışlardır.



Şekil 2: Kesişimlerdeki inceltme bozulmaları

Bir diğer yaklaşım uzaklık dönüşümüdür. Bu algoritmalar, resimdeki her pikselin yerine, piksele en yakın beyaz pikselin uzaklığını gösterecek şekilde, bir sayı atarlar. İki nokta arasındaki uzaklık, dörtlü zincir koduna göre, birbirine en yakın komşuların uzaklıkları olarak hesaplanır. Bu dönüşüm, bir fonksiyonla sıralı olarak tarama yapmakta ve bunu izleyen adımda ise ikinci bir fonksiyonla ters tarama işlemi gerçekleştirilmektedir.

Bir defa uzaklık fonksiyonu gerçekleştirildikten sonra, iskeleti bulmak için yerel maksimum işlemi uygulanır. Bu işlem, resmi tekrar oluşturmak için gerekli olan en az nokta sayısını sağlamak olarak gösterilmektedir. Şekil 3'de uzaklık dönüşümü ve iskelet gösterilmiştir. Uzaklık dönüşümü işlemi özyinelemeli ve özyinelemeli olmayan şekilde yapan değişik fonksiyonlar mevcuttur. Bu algoritmanın en kötü yanı, Şekil 3c.'de de görüldüğü gibi iskeletin, özellikle birleşim yerlerinde, bağlantısının kopmasıdır.



Şekil 3:Uzaklık dönüşümünün temsili gösterimi. (a) Resim (b) Uzaklık dönüşümü (c) İskelet

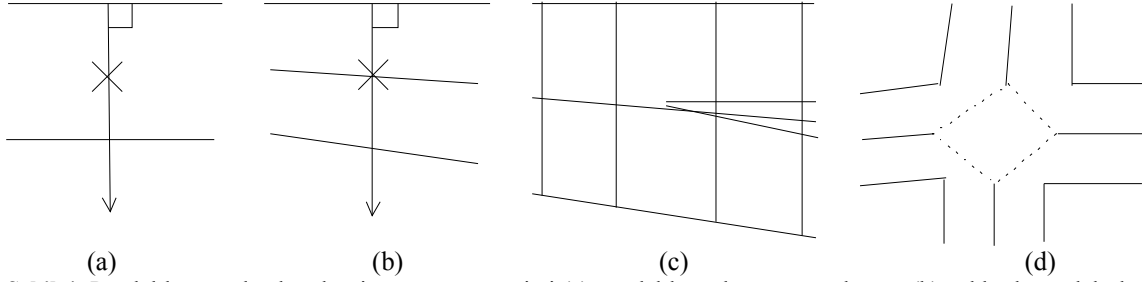
Yukarıdaki iki farklı algoritmayı birleştiren bir üçüncü algoritma ise, birleşik inceltme algoritmasıdır. Birleşim, iskeletin iki piksel kalınlığında olmasına sebep olmaktadır. Daha sonra iskelet, bir piksel kalınlığına indirilmektedir. Bu algoritma, geleneksel inceltme algoritmalarından daha doğru sonuç vermesine karşın, ek işlemler için daha fazla zaman gerektirir.

Genel olarak, inceltme algoritmalarının amacı, veri boyutunu azaltarak, sadece resimdeki şekilleri kullanmayı sağlamaktır. Genel olarak, ek işlemler gereklidir. Birçok inceltme algoritması, şekildeki ilişkileri onarma yeteneğine sahiptir. Fakat yüksek zaman karmaşıklığı, çizgi kalınlığı gibi şekil bilgilerinin kaybolması, kesişimlerde bozulma, yanlış ve sürpriz dallanmalar gibi dezavantajları vardır. Bu algoritmalar, daha çok çizgisel görüntülerin vektörizasyonunda kullanılır. Temel uygulama alanları görüntü alanının küçük ve çizgi kalınlığının önemli olmadığı, Optik Karakter Tanıma (OCR) uygulamalarıdır. Performans bakımından çok iyi olan algoritmalar da mevcuttur. Bu algoritmalar çok hızlı işlem yapabilir, bağlantı noktalarını iyi belirleyebilir fakat iskelet kalitesi çok iyi değildir. Bu tür bir algoritma, bir Optik Karakter Tanıma uygulaması için kullanılabilir. İnceltme işlemi sonucunda elde edilen iskelet, hala bit seviyesindedir ve vektörize edilmesi gerekmektedir. Bir piksel kalınlığında iskelet, kenar izleme alt

işlemi yoluyla bir zincire bağlanır. Bundan sonra çokgenleştirme işlemi, bir önceki adımda bulunan kritik noktaları birleştirip, çokgen oluşturur (Liu ve Dori, 1999).

2.3 Şekil Tabanlı Metotlar

Şekil tabanlı metotlarda (contour based methods), ilk önce raster görüntülerden şekiller çıkarılmakta ve daha sonra çizgi gibi alanları tanımlamak için eşlenebilir şekiller belirlenmektedir. Çoğunlukla nokta zincirleri ile temsil edilen orta eksenler, bu şekil çiftleri arasından oluşturulmaktadır (Song et al., 2002a). Şekil tabanlı vektörizasyon algoritmaları, aynı anda örnekleme ve orta eksen bulma işlemini gerçekleştirdikleri için inceltme tabanlı metotlardan farklıdır. Bu işlem, inceltme tabanlı metotlarda ilk önce tüm orta eksenlerin bulunması ve daha sonra çizgi izlenmesiyle gerçekleştirilmektedir. Kenarlar, çeşitli kenar çıkartıcı algoritmaları ile kolaylıkla çıkarılabilir. Kenarlar bulunduktan sonra orta eksenlerin nasıl bulunabileceği Şekil 4'de görülmektedir. Burada, kenarlardan birbirine indirilen dik çizgilerin orta noktaları orta eksenlerdir.



Şekil 4: Paralel kenar tabanlı vektörizasyonun gösterimi (a) paralel kenarların orta noktası (b) yaklaşık paralel olan kenarların orta noktası (c) kaçırılmış kesişme (d) karıştırılmış kesişme

Şekil tabanlı algoritmaların en büyük sorunu, kesişme noktalarındadır. Genel olarak iki temel sorundan bahsedilebilir. İlki, Şekil 4c'de görülen küçük açılı kenarlardır. Bunlar, kenardan çok, kenar çıkarma sırasında kaçırılmış çizgi parçacıklarına benzemektedir. İkinci sorun ise, Şekil 4d'de görüldüğü gibi çapraz kesişimin oluşmasıdır. Vektörizasyon algoritmasının, her türlü şekle karşı uygulanabilir olması, kesişimlerde yanlış karar vermemesi ve yanlış iskelet üretmemesi gerekliliği dikkate alındığında, çok kesişimin ve eğik çizgilerin olduğu çizimlerde kullanılması uygun değildir (Liu ve Dori, 1999).

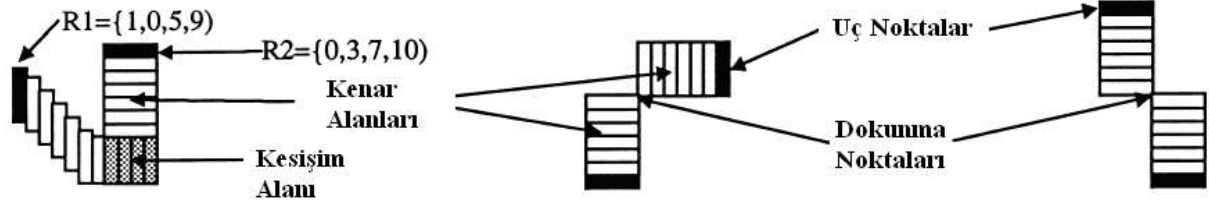
2.4 Grafik Yürütme Tabanlı Metotlar

Grafik yürütme tabanlı metotlar (Run Graph –Based Methods), yürütme uzunluğunu (run-length encoding) hesaplamak için raster görüntüleri ya satır ya da sütun boyunca taramaktadır. Daha sonra yürütmeler, grafik yapıları oluşturmak için analiz edilmektedir. Çizgi gibi alanlardaki yürütmelerin orta noktası, bir nokta zinciri biçimine çokgenleştirilmektedir (polygonized). Çizgi biçimi olmayan bir alan, bitişik kenarları birleştiren bir düğüm olmaktadır (Song et al., 2002a). Grafik yürütme tabanlı metotlar, yapısal gösterim için yeterli, çizgi çıkarmada ve bilgi elde etmede verimli olup, işlemesi de kolaydır. Şekil 5'de bir yürütme grafiğinin pek çok durumu gösterilmiştir. Bir yürütme, yatay veya dikey bir yöne sahiptir. Sahip olduğu yönde siyah piksel dizileri mevcuttur. Aşağıdaki şekilde ifade edilebilir.

$$R = \{d, c_d, b_d, e_d\}, b_d \leq e_d$$

Burada d, yürütme yönüdür yatay için 0 dikey için 1 değeridir. c_d , dik koordinattır. Eğer yürütme, yatay ise satır numarası, dikey ise sütun numarasıdır. b_d yürütmeye ait ilk pikselin koordinatı, e_d yürütmeye ait son pikselin koordinatıdır. Şekil 5'de dikey bir yürütme R1 ile yatay bir yürütme R2 ile gösterilmiştir. Bir yürütme aynı zamanda iki sonlandırma noktası ile de gösterilebilir. Bu durumda yön bu koordinatlardan anlaşılır. Eğer bu noktaların x koordinatı aynı ise yön dikey, y koordinatları aynı ise yön yataydır.

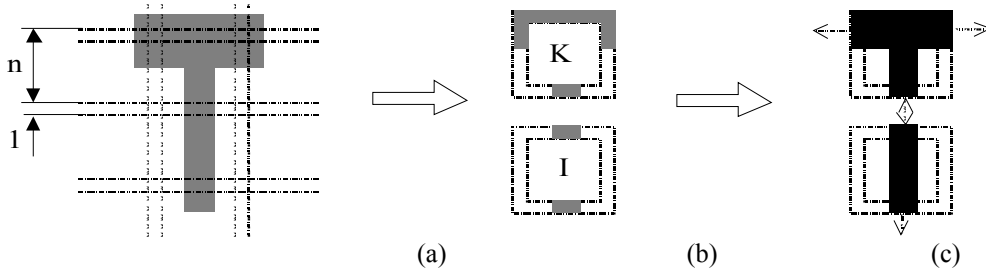
Bir resme ait yürütme grafiğini oluşturmak şu şekildedir: İlk adım, sırasıyla, sadece yatay ve sadece dikey yürütmelerden oluşan, basit dikey ve yatay yürütme grafikleri oluşturmaktır. İkinci olarak kenarlar, bitişik ve düzenli kısa yürütmelerle inşa edilir. Resmin geri kalanı, düğüm noktalarında, dikey yürütme ve kısa yürütmelerle kodlanır. Çizgi çıkarma işlemi, girdi olarak bir yürütme grafiği alır. Daha sonra düğümler sezgisel bir yöntemle, bağlı kenarların uzunluğu en çok olacak, alanları en az olacak şekilde rafine edilirler. Kenar alanlarındaki kısa yürütmelerin orta noktaları iskelet olarak alınır. Bundan sonra ise elde edilen iskelet çokgenleştirme işlemine gönderilir (Liu ve Dori, 1999). Bu metod, gürültüye duyarlıdır ve kesişimlerde bozulmalara neden olabilmektedir (Song et al., 2002a).



Şekil 5: Piksel olarak ifade edilen resimde, yürütme grafiklerinin temsili gösterimi

2.4 Ağ Desen Tabanlı Metotlar

Ağ desen tabanlı metotların (mesh based methods) temel fikri, tüm görüntüyü belirli bir ağı bölmek ve ağın sınırları içerisindeki siyah piksellerin dağılımına bakarak karakteristik desenleri belirlemektir. Bu desenler kullanılarak görüntüye ait bir kontrol haritası oluşturulmakta ve bu haritalar yardımıyla da uzun, düz çizgilerin çıkarılması gerçekleştirilmektedir. Şekil 6a'da, görüntü eşdeğer n büyüklüğündeki ağlara bölünmektedir. Her bir birim ağ, sadece sınırlardaki bir piksel genişliğindeki piksellere göre analiz edilmektedir. Elde edilen ağın özellikleri, veritabanındaki ile karşılaştırılmakta ve eşdeğeri ile etiketlenmektedir. Daha sonra görüntüdeki her ağ biriminin yerine gerçek görüntüde etiketi kullanılarak kontrol haritası oluşturulur.



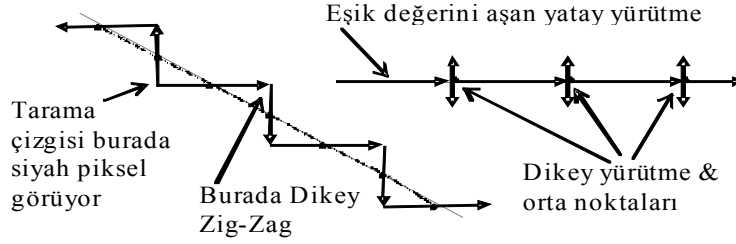
Şekil 6: Göz desen tabanlı çizgi çıkarma metodu. (a) görüntü ve gözler. (b) görüntünün iki merkezi gözünün, göz desen etiketleri ve kontrol haritası, (c) çizgiler kontrol haritasının analiz edilmesiyle çıkarılır.

Şekil 6b'de, Şekil 6a'da görülen, resmin ana bölümleri, sırasıyla K ve I olarak etiketlenmiş iki ağdan oluşan kontrol haritası ile temsil edilmiştir. Çizgiler, ağdan ağa, kontrol haritasındaki ağların karakteristikleri kullanılarak, Şekil 6c'de görüntü çıkarılmıştır. Eğer bilinmeyen, karmaşık desenler var ise soru işareti ile etiketlenirler. Bu tür alanlar, başka bir işlemle, kontrol haritası analizi sırasında, daha detaylı işlenir. Bu işlem, bu alanlardaki her pikseli tarar ve her siyah pikseli bir özelliğe veya çizgiye ait olup olmamasına göre etiketler.

Bu yöntemde ağ büyüklüğü en önemli konu olarak karşımıza çıkmaktadır. Büyük olursa daha fazla işlem zamanı harcanmaktadır. Küçük olursa erişim oranı artarak işlemi zorlaştırmaktadır. Ağ boyu, çizimdeki en büyük çizgi kalınlığından daha büyük iki çizgi arasındaki mesafeden de daha küçük olmalıdır. Çizgi tespiti gerçekleştirilmesi için, ağ boyu çizimdeki en küçük çizgi boyundan da küçük olmalıdır. Noktalar çizgi izleme aşamasında gözden kaçabilir. Bu durum noktaların görüntüde olması durumunda avantaj olabileceken aksi halde dezavantaj oluşturacaktır (Liu ve Dori, 1999).

2.5 Ayrık Piksel Tabanlı Metotlar

Ayrık piksel tabanlı metotlar (sparse pixel based methods), Dori tarafından geliştirilen dikey zig-zag (OZZ) metodundan esinlenerek yine Dori tarafından gerçekleştirilmiştir (Dori ve Liu, 1999). OZZ algoritmasının temel fikri, tek piksel kalınlığında bir ışık ışınının çizgileri takip ederek herhangi bir kenara çarpması durumunda dikey olarak dönmesi prensibi ile çalışır. Her yürütmenin, ince ışın ile kesişen orta noktası kaydedilmektedir. Bu durum Şekil 7'de gösterilmektedir. Eğer bir yürütme belirlenen bir eşik değerinden daha uzunsa, yürütme burada kesilmekte ve dikey bir yürütme yapılarak bunun orta noktası kaydedilmektedir. Böyle bir durumla takip edilen çizginin yatay veya dikey olması durumunda karşılaşılmaktadır. Şekil 7' de yatay durum gösterilmektedir.



Şekil 7: Orthogonal Zig-Zag (OZZ) vektörizasyon algoritmasının prensibi.

OZZ, zaman bakımından verimlidir. Çünkü seyrek piksel erişimi mantığıyla çalışmaktadır. Şekil 7’de görüldüğü gibi, OZZ’nin ziyaret ettiği piksel sayısı, resmin yükseklik ve genişliğinin toplamı ile doğru orantılıdır. Ayrıca gürültüye karşı duyarlıdır. Bundan başka, OZZ algoritması çubuk nesnelere elde etmek için tasarlanmıştır. Bu yüzden, eğik çizgiler ve yaylar bulunurken, bunları, sonları üst üste binmiş çubuklar olarak veya yanlış sonlandırılmış çubuklar olarak bulmaktadır. OZZ tanımı genişletilerek, Ayrık Piksel Tabanlı algoritma geliştirilmiştir. Ayrık Piksel Tabanlı algoritmada, OZZ aşağıdaki ilavelerle geliştirilmiştir:

- Genel izleme işlemi, daha önceden çalışan ve her siyah piksel alanına uygulanan başka bir işlemle tespit edilmiş güvenilir bir orta eksen noktasından başlayacaktır.
- Genel izleme işlemi, OZZ’de görülebilecek yatay, dikey ve eğik durumları kontrol edecek şekilde gerçekleştirilecektir. Bu yüzden sadece bir ekran geçişi gerekir ve iki geçişin birleşimi önlenir. Bu, Ayrık Piksel Tabanlı algoritmayı OZZ’den daha hızlı bir hale getirmektedir.
- Son olarak, çizgi izleme işlemi sırasında, nerede bir kesişim olursa, bir kesişim iyileştirme işlemi gerçekleştirilmektedir.

Metot hakkında ayrıntılı bilgi için “Liu and Dori, 1999” ve “Dori and Liu, 1999” kaynaklarından yararlanılabilir.

3. JEODEZİ VE FOTOGRAMETRİ MÜHENDİSLİĞİ AÇISINDAN VEKTÖRİZASYONUN ÖNEMİ

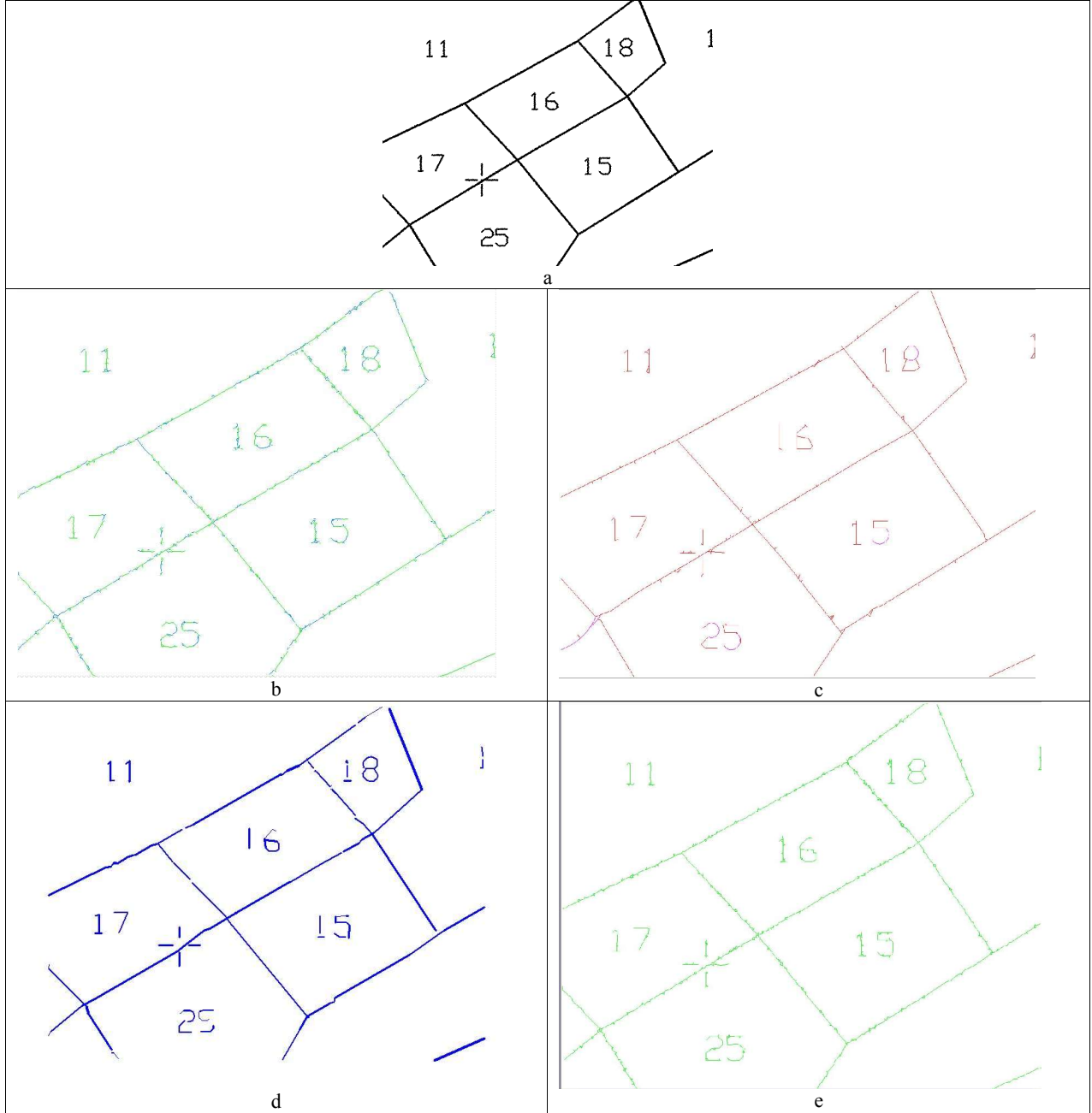
Gelişen bilgisayar teknolojisi ve buna paralel olarak gelişen ve yaygınlaşan Coğrafi Bilgi Sistemleri (CBS), coğrafi veriye olan ihtiyacı artırmaktadır. Coğrafi Bilgi Sisteminin en önemli bileşeni coğrafi veridir. Coğrafi veri kazanımı yollarından biri analog haritaların sayısallaştırılması, diğerleri ise hava fotoğraflarının ve uydu görüntülerinin değerlendirilmesidir. Analog haritalardan veri kazanımı amacıyla CBS teknolojisinin başlangıcında sayısallaştırıcı tablet kullanımı yaygın iken, günümüzde bilgisayarların grafik kapasitelerinin gelişmesi ve büyük formatlı tarayıcıların yaygınlaşması ile birlikte sayısallaştırıcı kullanımı terk edilmiştir. Analog haritalar, büyük formatlı tarayıcılarla görüntü dosyaları haline getirilerek manuel sayısallaştırma, otomatik ya da yarı otomatik vektörizasyon yollarıyla vektör formatında coğrafi veri kazanılmaktadır. Benzer durum, fotogrametri ve uzaktan algılama alanlarında da geçerlidir. Bu alanlarda da resimlerde ön işlemler (düşeye çevirme v.b) uygulandıktan sonra benzer şekilde veri kazanılmaktadır. Manuel sayısallaştırma yönteminin zaman ve işgücü kaybına yol açtığı bilinen bir gerçektir. Bu nedenle raster formatındaki görüntülerden otomatik olarak vektör formatında veri kazanımı yönündeki çalışmalar gerek akademik gerekse ticari anlamda yoğunlaşmaktadır. Vektörizasyon sadece CBS veya Jeodezi ve Fotogrametri Mühendisliğini ilgilendiren bir konu değildir. Hemen hemen tüm mühendislik alanlarındaki kağıt v.b altlıklardaki çizimlerin vektörizasyonu konusunda pek çok çalışma vardır (Song et al, 2002b). Buna paralel olarak görüntülerden obje tanıma yönelik (iris tanıma, yüz tanıma v.b) pek çok çalışma yapılmıştır ve yapılmaktadır. Bu gelişmelerin doğal olarak Jeodezi ve Fotogrametri Mühendisliği alanına yansımaları olmaktadır. Özellikle haritalardan ve hava-uydu görüntülerinden vektör veri elde etmeye yönelik uygulamalar vardır. Eş yükseklik eğrilerinin otomatik vektörizasyonu ile sayısal yükseklik modeli oluşturulmasını hızlandırmaya yönelik uygulamalardan uydu görüntülerindeki yol objelerinin vektörizasyonuna (road extraction) kadar pek çok uygulama alanı vardır.

4. TİCARİ AMAÇLI VEKTÖRİZASYON YAZILIMLARI

Vektörizasyon işlemi gerçekleştirilmek için birçok ticari yazılım geliştirilmiştir. Bunlardan bazıları, WinTopo, Scan2CAD, Acme TraceART, R2V ve NETCAD olarak sıralanabilir. Aşağıda bu yazılımlar hakkında kısa bir değerlendirme yer almaktadır.

WinTopo, SoftSoft Şirketi tarafından geliştirilmiştir. Vektörizasyon iki adıma gerçekleştirilmektedir. Birinci adımda; inceltme işlemi, ikinci adımda; piksel çizgilerinden gerçek vektörlerin çıkarılması işlemi gerçekleştirilir. 5 adet inceltme metodu kullanılır. Bunlar; Stentiford inceltme, Zhang Suen inceltme, Best Combination, Simple Edge Detection, Canny Edge Detection’dur. İkinci adımda ise; ilk önce kayıpsız olarak vektörizasyon yapılmakta, daha sonra yumuşatma veya çoklu çizgi azaltma ya da her ikisi uygulanmaktadır (URL1).

Scan2Cad yazılımı, mimari, elektrik makine, CNC profil, topoğrafik harita çizimlerini ve şehir planlarını vektöre dönüştürebilmektedir (URL2). Acme TraceART yazılımında ilk önce kenarlar çıkarılmakta sonra vektörizasyon işlemi gerçekleştirilmektedir (URL3). R2V yazılımı, Able yazılım şirketi tarafından geliştirilmiştir. Hem haritaların hem de diğer çizgisel görüntülerin vektöre dönüştürülmesi gerçekleştirilebilir. Siyah-beyaz, gri değerli ve renkli görüntüye uygulanabilmektedir (URL4). Tüm yazılımlarda, otomatik vektöre dönüşüm tamamlandıktan sonra şekilleri düzeltme olanağı sağlanmıştır. Şekil 8’de bu yazılımlarla vektöre dönüştürülmüş şekiller görülmektedir.



Şekil 8: a) Orijinal raster görüntü b)Win Topo yazılımı ile elde edilen vektör görüntü c) Scan2CAD yazılımı ile elde edilen vektör görüntü d)Acme TraceART yazılımı ile elde edilen vektör görüntü e) R2V yazılımı ile elde edilen vektör görüntü

NETCAD yazılımının Vectory modülünde ise yarı otomatik bir yaklaşım uygulanmıştır. Burada kullanıcı sayısalıştırmaya başlayacağı çizgiyi seçer daha sonra seçilen çizgi otomatik olarak bitene ya da ikiye (üç) ayrılıncaya kadar izlenir. Bu yaklaşımla otomatik vektörizasyonda oluşan gereksiz çizgilerden ve kopukluklardan sakınılmaktadır. Ancak interaktif olarak çalışıldığından daha fazla zaman ve işgücü gerektirmektedir.

5. SONUÇ VE ÖNERİLER

Bu çalışmada analog haritalardan mekansal veri kazanımında elle sayısallaştırmaya alternatif olan vektörizasyon işlemi ele alınmıştır. Vektörizasyon yazılımları bilgisayar ve tarayıcı teknolojilerine paralel olarak gelişmekte, vektörizasyon algoritmaları konusunda çok sayıda yayına rastlanmaktadır. Genel anlamda vektörizasyon görüntü işleme konusu altında ele alınmaktadır. Bu çalışmada konu ile ilgili kaynaklar taranarak kullanım alanı yaygın olan yöntemler tanıtılmış, konunun Jeodezi ve Fotogrametri Mühendisliği disiplini açısından önemi irdelenmiştir. Vektörizasyon değişik disiplinlerdeki çizimler için çizimlerin yapıları dikkate alınarak yapılmalıdır. Bu bağlamda her yazılım ya da her algoritma haritalardan veri kazanımı için uygun olmayabilir. Hangi algoritma uygulanırsa uygulansın, vektörizasyon sonrası elde edilen verilerin vektör yapıları içinde test edilmesi ve gerekli düzeltmelerin yapılmasının zorunludur. Bu işlem İngilizce kaynaklarda “post-processing” olarak adlandırılmaktadır. Vektöre dönüştürülen çizimlerin haritalar açısından kullanılabilir olup olmaması da post-processing işleminin amaca uygun yapılmasına bağlıdır. Daha açık bir ifade ile vektörizasyon sonrası uygulanacak post-processing işlemi ile ham nitelikteki vektör verilere ilgili disipline kullanılabilirlik özelliği kazandırılmaktadır.

Konuyu analog haritalardan mekansal veri kazanımı açısından ele aldığımızda, vektörizasyon hızlı ve güvenilir veri kazanımında manuel sayısallaştırmaya alternatif bir yöntemdir. Vektörizasyonun hızlı olduğu kesindir, ancak güvenilir sonuçlar verip vermemesi görüntünün kalitesine ve uygulanacak otomatik post-processing işlemine bağlıdır. Post-processing yapılmamış verilerin manuel olarak düzeltilmesi çoğu kez elle sayısallaştırmadan daha uzun zaman alır. Geliştirilen yazılımların görüntü kalitesine çok fazla bağımlı olması nedeniyle taranmış görüntülerden elle sayısallaştırma halen yaygın olarak kullanılmaktadır. Post-processing işlemine gerek bırakmayacak ve görüntüdeki gürültünün etkisini ortadan kaldıracak bir çözüm yolu da yarı otomatik vektörizasyondur. Bu yaklaşımda vektörizasyon interaktif olarak operatör kontrolünde gerçekleşir.

İncelenen yazılımlar görünüm olarak aynı sonuçları vermekte, ancak nokta yoğunluğu, çoklu doğruların bulunması, kesişimlerde ortaya çıkan durumlar açısından farklılıklar göstermektedir. WinTopo diğerleri arasında konuya daha bilimsel yaklaşmakta, değişik algoritmalar arasından seçme olanağı tanımaktadır. NETCAD yazılımında ise yarı otomatik yaklaşım benimsenmiştir.

KAYNAKLAR

Dori D. And Liu W., 1999. *Sparse Pixel Vectorization: An Algorithm and Its Performance Evaluation*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21, No. 3, 202-215

Liu W. and Dori D., 1999. *From Raster to Vectors: Extracting Visual Information From Line Drawing*, Pattern Analysis & Applications (1999) 2, 10-21.

Song J., Su F., Tai C.T. and Cai S., 2002a. *An Object-Oriented Progressive-Simplification-Based Vectorization System for Engineering Drawings: Model, Algorithm, and Performance*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24, No. 8, 1048-1060.

Song J., Su F., Li H. And Cai, 2002b. *Raster to Vector Conversion of Construction Engineering Drawings*, Automation in Construction 11(2002) 597-605

Tombre K., 1998. *Analysis of Engineering Drawings: State of the Art and Challenges*, Proceedings of the Graphics Recognition-Algorithms and Systems, 257-264

URL1: <http://wintopo.com> 11.03.2005

URL 2: <http://www.softcover.com> 11.03.2005

URL 3: www.freefirestudio.com 11.03.2005

URL 4: <http://www.ablesw.com/r2v> 11.03.2005