

DiJiTAL RESMİN
YENİDEN ŞEKİLLENDİRİLMESİ VE DEĞERLENDİRİLMESİ

Ayşe Uğuzavcı /

Anadolu Üniversitesi
Fen Bilimleri Enstitüsü
Lisansüstü Yönetmeliği Uyarınca
Elektrik ve Elektronik
Mühendisliği Ana Bilim Dalı
Elektronik Bilim Dalında
YÜKSEK LİSANS TEZİ
Olarak Hazırlanmıştır.

Danışman : Dr. Erdener İldiz

Şubat-1989

Ayşe Uğuzavcı'nın YÜKSEK LİSANS tezi olarak hazır-
ladığı " DİJİTAL RESMİN YENİDEN ŞEKİLLENDİRİLMESİ VE
DEĞERLENDİRİLMESİ" başlıklı bu çalışma jürimizce lisansüstü
yönetmeliğinin ilgili maddeleri uyarınca değerlendirilerek
kabul edilmiştir.

30 / 1 / 1989

Başkan: Prof. Dr. Atalay BARKANA

Uye .. Prof. Dr. Atila BARKANA

Uye .. Dr. Erdener İLDİZ

31 Fen Bilimleri Enstitüsü Yönetim Kurulu'nun
OCAK 1989 gün ve 201/8 sayılı kararıyla
onaylanmıştır.

Enstitü Müdürü
Prof. Dr. Rüstem Kaya

TEŞEKKÜR

Çalışmalarım Süresince araştırma ve uygulamalarda benden yardımlarını esirgemeyen danışman hocam,

Dr. Erdener İldiz'a

çalışmanın yazılımı esnasında destek ve yardımcı olan,

Ars.Grv. Selçuk Canbek'e

tezin özellikle halıcılık ile ilgili olan kısımlarında beni aydınlatarak yardımcı olan,

ERA Halıcılık ve Pazarlama A.Ş'e,

tezin yazımı ve çizimi esnasında emeği geçen Mehmet Kahvecioğlu ve tüm arkadaşlarıma teşekkürü bir borç bilirim.

ÖZET

Bu çalışmanın amacı, sayısal bir resmin bilgisayarla yeniden şekillendirilmesi ve değerlendirilmesidir. Geliştirilen sistem değişik kullanım alanlarına sahip olmakla birlikte çalışmalar, halı desenleri üzerinde yoğunlaştırılmıştır. Halı desenlerinin oluşturulması, çoğaltılması ve bu desenleri oluşturan motifler üzerinde bazı değişikliklerin yapılması gibi işlemler bilgisayarla gerçekleştirilmiştir.

Giriş bölümünde, tezdeki bölümlerin geniş bir özeti ve görüntü ile ilgili kısa tanımlar verilmiştir.

İkinci bölümde, özellikle gerçek görüntüler ile çalışıldığında, görüntüyü gözle yoruma daha uygun bir hale getirmek için uygulanan yöntemlerden bazıları anlatılmıştır.

Üçüncü bölümde, siyah-beyaz ya da renkli bir görüntüde görüntü kenarlarının daha belirgin olması için kullanılan yöntemlerden birkaçı tanıtılmıştır.

Dördüncü bölümde, sayısal görüntü işleme için geliştirilmiş özel sistemler hakkında bilgi verilmiştir.

Beşinci bölümde, halı desen ve motiflerinin bilgisayar yardımıyla saklanması, geliştirilmesi ve çoğaltılması amacıyla geliştirilen bir yöntem tanıtılmıştır.

Altıncı bölümde, bir halının dokunmadan önce kalitesine göre ip kalınlıklarının ya da ip kalınlıklarına göre kalitesinin belirlenmesi ve halı boyutunun hesaplanması ile ilgili tezdeki çalışma tanıtılmıştır.

Yedinci bölümde, sayısal görüntü işleme üzerine geliştirilmiş olan tekniklerin ışığı altında sayısal bir resmin (halı deseni) yeniden şekillendirilmesi ve değerlendirilmesi üzerine, tezde geliştirilen sistem tanıtılmış, desen oluşturma, deseni belli bir düzende saklama, motiflerde bazı değişiklikler yapma ve desen çoğaltma gibi konularda yapılan yazılım hakkında bilgi verilmiştir.

Ekler kısmında ise, geliştirilen programın listesi, program kullanım örnekleriyle programın çalışması ve yazıcıdan elde edilen desen çıktıları verilmiştir.

SUMMARY

The purpose of this study is to evaluate and reform a digital picture by computer. The improvised system having different usage fields, has been concentrated on carpet desing. Creating carpet desings, increasing them and making some changes on motives which form these desing have been realised by computer.

Introduction part has a wide summary of the chapters in the thesis and short definitions related to vision.

In the second chapter, the methods, which have been applied in order to have a suitable form for visual interpretation, have been explained especially when they are stuied with real visions.

In the third chapter, some of the methods which have been used in black-white or color vision for clarifying vision frames, have been introduced.

In the fifth chapter, a method, which has been improved as the purpose of increasing, developing and keeping carpet desing and motives by means of computer, has been presented.

In the sixth chapter, it has been told that the carpet size has been estimated according to thick of strings.

In the seventh chapter, it has been studied on evaluating and reforming of a digital picture (carpet desing) in the light of techniques which have been developed on digital visual drawing. In this study, the system mentioned above has been introduced and the software about creating desing, keeping it in a systematic way, making some changes on motives and increasing designs has been given.

In the appendix part, the list of improved programme, samples of the programme usage, how it works and design outputs which have been taken from printer have been added.

İÇİNDEKİLER

Sayfa

ÖZET.....	
SUMMARY.....	
ŞEKİLLER DİZİNİ.....	
1. GİRİŞ.....	1
1.1. Giriş.....	1
1.2. Görüntünün Tanımı.....	7
1.3. Yersel Sayısallaştırma, Piksel, Ağa, Ayrım.....	8
1.4. Sayısal Görüntü.....	10
1.5. Görüntü ve Resim.....	12
2. GÖRÜNTÜ ZENGİNLEŞTİRME YÖNTEMLERİ.....	15
2.1. Giriş.....	15
2.2. Komşusal işlemler.....	16
2.3. Katlama.....	17
2.4. Fourier Dönüşümü.....	20
2.5. Orta Değer Süzmesi.....	21
2.6. Renk Zenginleştirme.....	21
3. GÖRÜNTÜ KENARLARININ BULUNMASI.....	23
3.1. Giriş.....	23
3.2. Yönlü Laplace Değişim Maskeleriyle Görüntü Kenarlarının Bulunması.....	23
3.2.1. Değişim maskeleri.....	23
3.2.2. Kenar değişim resmi.....	26
3.2.3. Eşikleme.....	29

3.2.4. Kenar resmi.....	30
3.2.5. Gürültü eleme.....	31
3.3. Filtrelerle Görüntü Kenarlarının Bulunması.....	32
4. SAYISAL GÖRÜNTÜ İŞLEME SİSTEMLERİ.....	35
4.1. Giriş.....	35
4.2. Donanım.....	35
4.3. Yazılım.....	36
5. TÜRK HALI DESEN VE MOTİFLERİNİN BİLGİSAYAR YARDIMIYLA SAKLANMASI, GELİŞTİRİLMESİ VE ÇÖZÜLTÜLMESİ AMACIYLA GELİŞTİRİLEN BİR YÖNTEM.....	38
5.1. Giriş.....	38
5.2. Halı Deseninin Bilgisayara Yükleneceği...39	39
5.3. Döndürme.....	39
5.4. Renklendirme.....	40
5.5. Bordür Renklendirme.....	40
5.6. Alan Renklendirme.....	40
5.7. Komut Formatı.....	41
6. HALİNİN DOKUNMADAN ÖNCE GERÇEK BOYUTUNUN HESAPLANMASI.....	42
6.1. Giriş.....	42
6.2. İp Kalınlıklarına Göre Halı Boyutunun Hesaplanması.....	42
7. YAZILIM.....	48
7.1. Giriş.....	48
7.2. Ana Program.....	50
7.2.1. Yeni bir çizime başlama.....	52
7.2.2. Mevcut bir çizim üzerinde	

değişiklik yapma.....	53
7.2.3. Resmin sınırlarının bulma.....	53
7.2.4. Sınırlardan orijinal resmi elde etme.....	59
7.2.5. Sınırlar üzerinde değişiklik yapma.....	63
7.2.6. Boyut hesaplama.....	65
8. SONUÇ.....	71

KAYNAKLAR DİZİNİ.....	73
-----------------------	----

EKLER

- A. PROGRAM LİSTESİ
- B. PROGRAM KULLANIM ÖRNEKLERİ
- C. PROGRAM ÇIKTILARI

ŞEKİLLER DİZİNİ

Şekil

Sayfa

1.1. AGA-piksel mesafe ilişkilerinin üç farklı durumu.....	9
1.2. CIE standartlarına göre x,y,z ana renk bileşenleri.....	13
3.1. Değişim maskeleri.....	24
3.1.a Kare değişim maskesi.....	24
3.1.b Çapraz değişim maskesi.....	24
3.2. Kare ya da çapraz pencereler.....	25
3.2.a Kare pencere.....	25
3.2.b Çapraz pencere.....	25
3.3. Görüntü kenarlarının bulunmasında kullanılan filtre.....	32
6.1. Düğüm atma teknikleri.....	43
7.1. Ana program akış şeması.....	51
7.2. Motif sınırlarının bulunmasında kullanılan filtre.....	53
7.3. Deseni oluşturan motiflerin sınırlarını bulma alt programı.....	54
7.4. Bir motife ait bilgilerin kütük içine yerleştirilmesi.....	57
7.5. Sınırları belirlenmiş bir motif örneği....	58
7.6. Örnek motife ait bilgilerin kütük içine yerleşimi.....	59
7.7. Sınırları belirlenmiş desenin orijinaline dönme alt programı.....	60
B.1. Ana listeden 1.nolu kod seçildiğinde ekran düzenlemesi.....	B.2
B.2. Pencere içine alarak silme, kopyalama, taşıma işlemleri.....	B.5
B.3. Ana listeden 2.nolu kod seçildiğinde ekran düzenlemesi.....	B.7
B.4. Ana listeden 3.nolu kod seçildiğinde ekran düzenlemesi.....	B.9

- B.5. Ana listeden 4.nolu kod seçildiğinde ekran düzenlemesi.....B.11
- B.6. Ana listeden 5.nolu kod seçildiğinde ekran düzenlemesi.....B.14
- B.7. Renkler ve renkleri ifade eden şekiller...B.23

1. GİRİŞ

1.1. Giriş

Bilgisayarla görüntü işleme, günümüzde çok hızlı gelişme gösteren ve çok farklı kullanım alanlarına sahip bir daldır.

Sayısal görüntü işleme modern anlamda son onbeş yıldır gelişmekte olan bir teknolojidir. Her ne kadar fotogrametri ve hava fotoğraflarının yorumu çok daha öncelere gitmekteyse de, uydu görüntüleri, çok spektral bantlı tarayıcılar ve yoğun bilgisayar kullanımı ile sivil alanlara yönelik yaygın çalışmaların gelişmesi 1972'den sonralara dayanmaktadır. Özellikle uzaktan algılama teknolojisinin gelişmesi büyük ölçüde bilgisayar alanındaki "sayısal görüntü işleme" (digital image processing), kısaca SiG, tekniklerin gelişmesine bağlı olmuştur. Çok bantlı tarayıcılar ile alınan sayısal resimlerden bilgi çıkarmanın bilinen fotoğrafik yollarla zor, hatta bazen imkansız olduğu gözlenmiş, bu amaçla bilgisayar teknolojisinden yararlanma yollarına gidilmiştir.

Sayısal görüntü işleme, görüntü biçimindeki verilerin bilgisayar kullanarak işlenmesi anlamına geldiği için, literatürde bilgisayarla görüntü işleme (computer image processing) diye de anılmaktadır. Gerçekte görüntülerin sayısal işlenmesi için genel amaçlı bilgisayarlar yanında dizi işlemci (array processor) ve gösterge işlemci (display processor) denilen özel sayısal elektronik cihazlar da geniş ölçüde kullanılmakta hatta gerekli olmaktadır.

Bu tezde, sayısal bir resmin yeniden şekillendirilmesi ve değerlendirilmesi üzerine bir sistem geliştirilmiştir. Türk halıcığında desen oluşturma, desen üzerinde değişiklik yapma ve desen çoğaltma gibi konularda

karşılaşılan emek ve zaman sorunlarına bir ölçüde cevap olması açısından, çalışmada sayısal resim olarak halı desenleri seçilmiştir.

Çalışmada, üzerinde inceleme ve değişiklik yapılacak olan desen dijitalizer ya da dijital kamera yoluyla elde edilemediğinden, programın bir kısmı desen çizme bölümüne ayrılmıştır. Bu bölümde bilgisayardaki fonksiyon tuşları, çizime yardımcı olacak bazı temel işlemlere programlanmıştır. Desen üzerinde, bazı kısımları çerçeve içine alarak, desen üzerinde istenen yerlere kopya etme veya taşıma fonksiyonuyla da, desen çizmede zamandan büyük ölçüde tasarruf edilmektedir. Böylece herhangi bir desen oluşturma kolaylıkla ve kısa zamanda gerçekleştirilmektedir. Bilgisayarda çizim alanı olarak ayrılmış bölgede her nokta bir düğüme karşılık gelmektedir.

Desen çizme işlemi gerçekleştirildikten sonra, desen üzerinde yapılmak istenen değişikliğe göre programın diğer bölümleri seçilebilmektedir. Bu bölümlerin fonksiyonları kısaca şöyledir:

Renkli olarak çizilen bir desen depolandıktan sonra göze çarpan birtakım hatalar ya da yapılmak istenen değişiklikler, tüm desenin yeniden çizilmesini gerektirmeden düzeltilebilmektedir.

Çizilmiş olan bir halı deseninde, deseni oluşturan motiflerin iç kısımları boşaltılarak (siyaha dönüştürülerek) sadece sınırları orijinal renkleriyle belirtilebilmektedir. Bu fonksiyonu gerçekleştirmek için, daha önce aynı konu üzerinde yapılan çalışmaların ışığı altında bir filtre tanımlanmış ve bu filtre yardımıyla deseni oluşturan motiflerin sınırları bulunabilmiştir. Sınırlar halinde belirlenmiş olan desen depolanmak istendiğinde, deseni oluşturan her düğüme ilişkin bilginin saklanması yerine sadece sınırdaki düğümlere ait bilgiler de-

polanmaktadır. Bu amaçla, bir sınır takip etme metodu geliştirilmiştir. Siyahtan farklı bir sınır rengi yakalandığında, sınır takip etme işlemine başlanarak, takip edilen renk sona erdiğinde ya da takip etme işlemine başlanan koordinatlara geri döndüğünde bu işlem sona ermektedir. Sınır takip etme işlemi sırasında, sınırı tanımlayan bilgiler, belli yönlerde belli uzunluklar şeklinde belirlenmektedir. Yön koduyla birlikte, o yöndeki uzunluk, geliştirilen bir formülasyonla tek bir karakter olarak ifade edilip depolanmaktadır. Böylece deseni depolama işlemi sırasında bilgisayar kapasitesinden de büyük ölçüde tasarruf edilmektedir. Halı desenlerini çizen desinatörlerde genellikle farklı renklere sahip motifleri, sadece sınırları çizerek göstermektedirler. Bu nedenle desenin dokuma için yazıcıdan çoğaltılması sırasında yazıcı, desen sınırlar halinde belirlendiği zaman çalıştırılmalıdır.

Deseni oluşturan iç motifler sadece sınırlarla belirlenip bu şekilde bir formül kütüğünde depolandıktan sonra, istenildiği zaman boş alanlar renklendirilerek tekrar orijinal renkli resim elde edilebilmektedir.

Yapılan çalışmanın diğer bir bölümünde ise, desen sınırlar halinde belirlenip depolandıktan sonra, deseni oluşturan motifler üzerinde birtakım değişiklikler yapılabilmektedir. Bu değişiklikler motifin başlangıç koordinatlarını, rengini değiştirme, motifi belli oranlarda büyültüp küçültme ve belli açılarda motifi döndürme şeklinde olmaktadır. Motifler üzerindeki değişiklikler, bir kütük içinde sınırları tanımlayan bazı parametrelerin değiştirilmesiyle gerçekleştirilmektedir. Parametre değiştirme işleminde kolaylık sağlaması açısından, fonksiyon tuşları belirli işlevleri gerçekleştirecek şekilde programlanmaktadır.

Kütüklerle ilgili bazı yardımcı işlemlerin (kütük isimlerinin değiştirilmesi, kütüklerin silinmesi ve desenlerin depolandığı kütük isimlerinin listesi) gerçekleştirildiği -

rilmesi ve desenlerin yazıcıdan çoğaltılması için gerekli parametrenin ayarlanması da yazılımın gerçekleştirdiği fonksiyonlar arasındadır.

Sayısal görüntü işleme üzerindeki çalışmalar, özellikle gerçek görüntüler (uzay fotoğrafları...gibi) çalışıldığında, elde edilen görüntüleri gözle yoruma daha uygun bir hale getirmek için bazı yöntemler uygulamak ihtiyacıyla gelişme göstermiştir. Görüntüleri zenginleştirmek, görüntünün belli bölgelerini daha belirgin hale getirmek, bunun yanısıra bazı bölgeleri de bastırmak amacıyla çok farklı yöntemler uygulanmaktadır. Bu yöntemler sayesinde görüntüye karışmış olan gürültüler de elenmektedir (1).

Görüntü zenginleştirme yöntemlerinin, herbirinin belli alanlarda avantajları ve dezavantajları olmaktadır. Gerçekleştirilmek istenen amaç doğrultusunda bu yöntemlerden uygun olanlar seçilebilmektedir. Daha sonraki bölümlerde tanıtılacak olan görüntü zenginleştirme yöntemlerinden bazıları, komşusal işlemler, katlama, fourier dönüşümü, orta değer süzmesi ve renk zenginleştirmedir.

Bir görüntüde görüntü kenarlarının bulunması, görüntüde göze çarpmayan ayrıntıların ortaya çıkarılması açısından büyük önem taşımaktadır. Görüntü kenarlarının bulunmasında pek çok farklı yöntem kullanılmaktadır. Bunlardan birisi de değişim maskelerinin kullanılmasıdır. Bu yöntemde Laplace değişim maskeleri parçalanarak yönlü Laplace değişim maskeleri elde edilmiş ve sonucun daha gürültüsüz, iyi sonuçlar verdiği gözlemlenmiştir (2). Yönlü Laplace değişim maskeleriyle uygulanan işlem sonucunda görüntü kenarlarının bulunması yöntemi, bu alanda iyi sonuçlar veren Prewitt ve Sobel maskeleri kadar kullanışlı olmaktadır. Özellikle siyah-beyaz resimlerde görüntü kenarlarının bulunması için geliştirilmiş olan bazı filtrelerde bulunmaktadır. Filtre yöntemlerinde ise incelenen nokta ile onun

komsuluğunda olan noktalar arasında bazı işlemler uygulanarak, incelenen noktanın sınır olup olmadığına karar verilmektedir (3).

Bu tezde, görüntü zenginleştirme yöntemlerinden bazılarının ve görüntü kenarlarının bulunmasında uygulanan işlemlerden uygun olanlarının, Türk halıcılığının bazı problemlerine çözüm olup-olmayacağı araştırılmıştır. Daha önce yapılan çalışmalar doğrultusunda, çizilmiş olan bir halı deseninde, motiflerin sınırlarını belirlemek amacıyla bir filtre geliştirilmiştir.

Türk halıcılığının yaşatılması, geliştirilmesi ve yaygınlaştırılması açısından, desenlerdeki renk zenginleştirilmesinin yapılabilmesi, desenlerin ucuz ve hızlı çoğaltılması amacıyla bir program geliştirilmiştir (4). Yapılmış olan bu çalışmada, Türk halıcılığının, eldeki desenlerin azlığı ve yeni desenlerin geliştirilmesi ile desenlerin çoğaltılmasında karşılaşılan güçlükler gibi iki önemli soruna çözüm aranmıştır. Halı desenlerinin elle hazırlanması ve çoğaltılması çok emek, zaman ve yüksek maliyet gerektirmektedir. Bu nedenle bu tür çalışmaların bilgisayar yardımıyla yapılması büyük bir avantaj sağlamaktadır. Geliştirilmiş olan bu yöntemde, desinatör tarafından serbest formda hazırlanmış olan desen paftası sayısallaştırıcı tablet üzerinde tanımlanmış özel komutlar yardımıyla bilgisayara yüklenmektedir. Desenin bilgisayara yüklenmesi işlemi tamamlandıktan sonra, tanımlanan özel komutlar sayesinde desen üzerinde belli bir alan silinebilmekte, çerçevelerle belirlenen alanlar veya motifler saklanabilmekte, olduğu gibi ya da döndürülerek başka yerlere kopya edilebilmekte, sınırlarla belirlenen alanlar renklendirilebilmektedir.

Tezin bölümlerinde yer alan çalışmalar, aşağıda kısaca verilmiştir.

ikinci bölümde, özellikle gerçek görüntüler dediğimiz uzay fotoğrafları ile yapılan çalışmalarda, alınan görüntüyü gözle yoruma daha uygun bir hale getirmek için geliştirilen bazı yöntemler anlatılmaktadır. Bu yöntemlerin sayısı bir hayli olmakla birlikte bunlardan seçilen beş tanesine yer verilmekte ve belli alanlardaki avantajları gösterilmektedir (1).

Tezin üçüncü bölümünde, renkli ya da siyah-beyaz görüntülerde görüntü kenarlarının bulunmasında, görüntü üzerinde bir filtre doluşturma, yönlü Laplace değişim maskeleriyle Prewitt ve Sobel maskelerinin kullanılmasıyla ilgili yöntemler anlatılmaktadır (2)(3).

Dördüncü bölümde, sayısal görüntü işlemede genel amaçlı bilgisayarların bir çok bakımdan yetersiz kalması sonucu geliştirilen, özel donanım, yazılım ve yan birimler içeren "Sayısal Görüntü İşleme Sistemleri" (GIS) hakkında genel bilgiler verilmektedir (1).

Tezin beşinci bölümünde, Türk halıcılığının, mevcut desenlerin azlığı ve yeni desenlerin geliştirilmesi ile desenlerin çoğaltılmasında karşılaşılan güçlükler gibi iki önemli sorununa belli ölçüde çözüm olması açısından geliştirilmiş olan program tanıtılmaktadır. Bu programla, halıların renk ve desen açısından zenginleştirilmesi ayrıca desenlerin ucuz ve hızlı olarak çoğaltılıp yaygın kullanıma sunulması amaçlanmaktadır (4).

Altıncı bölümde, halı dokunmadan önce gerçek boyutunun hesaplanmasına yönelik olarak bu tezde geliştirilen yaklaşım anlatılmaktadır. Özellikle el dokuması halılar için üretilen iplerin standart olmaması ve insan faktörünün bulunması nedeniyle, dokuma işleminden önce boyut hesaplama oldukça güçtür. Buna rağmen örnek halılar üzerinde yapılan çalışmalar sonucunda pratik uygulama ile teorik boyut hesaplama arasında oldukça yakın sonuçlar elde edilmiştir.

Buna göre kalitesi, çözgü kalınlığı ve sipariş ebatı bilinen bir halıda, enine ve boyuna olması gereken düğüm sayısı, ilmelik ve atkı kalınlığı yazılım tarafından kullanıcıya verilmektedir. Ayrıca çözgü, atkı, ilmelik ip kalınlığı ve sipariş ebatı belli olan bir halının kalitesi, enine ve boyuna tüm halıda olması gereken düğüm sayısının belirlenmesi de yazılımın sağladığı veriler arasındadır.

Tezin yedinci bölümünde, sayısal görüntü işleme ve görüntü kenarlarının bulunması üzerine geliştirilmiş olan birtakım tekniklerin ışığı altında sayısal bir resmin yeniden şekillendirilmesi ve değerlendirilmesi üzerine bu tezde geliştirilen sistem tanıtılmaktadır. Özellikle desen oluşturma, deseni oluşturan motiflerin sınırlarını orijinal renkleriyle belirleme, deseni belli bir düzende saklama, desen üzerinde birtakım değişiklikler yapma ve desen çoğaltma gibi konularda yapılmış olan yazılım çalışmaları hakkında bilgi verilmektedir.

Tezin Ekler kısmında ise, tezin yazılımında oluşturulan programın listesi, kullanıcıya kolaylık sağlaması açısından program kullanım örnekleri detaylı bir şekilde verilmektedir. Ayrıca desenlerin çoğaltılması amacıyla yazıcıdan elde edilen desen çıktıları ilave edilmiştir.

1.2 Görüntünün Tanımı

Günlük yaşamda resim de diyebileceğimiz "görüntü" (image) en genel anlamıyla iki boyutlu bir işaret (signal) kayıdır. Fotoğraf ya da televizyon ekranı gibi gözle görünen bir biçimde olabileceği gibi, manyetik bantta yazılı bir kayıt, ya da bilgisayar belleğinde duran sayısal değerler biçiminde de olabilir.

Önce iki tür görüntü arasındaki önemli bir farkı belirtmek gerekir. Bu fark "sürekli-ayrık", "analog-sayısal" ya da "sürekli-sayısal" olarak ortaya konabilir. Bir fotoğraf üzerindeki parlaklık hem seviye olarak hem de konum olarak sürekli değişen bir değerdir. Buna karşılık bilgisayar içinde değerler ayrık sayılar biçiminde temsil edilirler. Bunun için bir resmin bilgisayara girebilmesi, onun belirli sayıda konum (ya da nokta) ve belirli sayıda parlaklık değeri ile ifade edilmesiyle mümkündür. "Sayısallaştırma" denen bu işlem resmin sürekli değişkenlerini ayrık değişkenlere dönüştürmek olarak da tanımlanabilir (1).

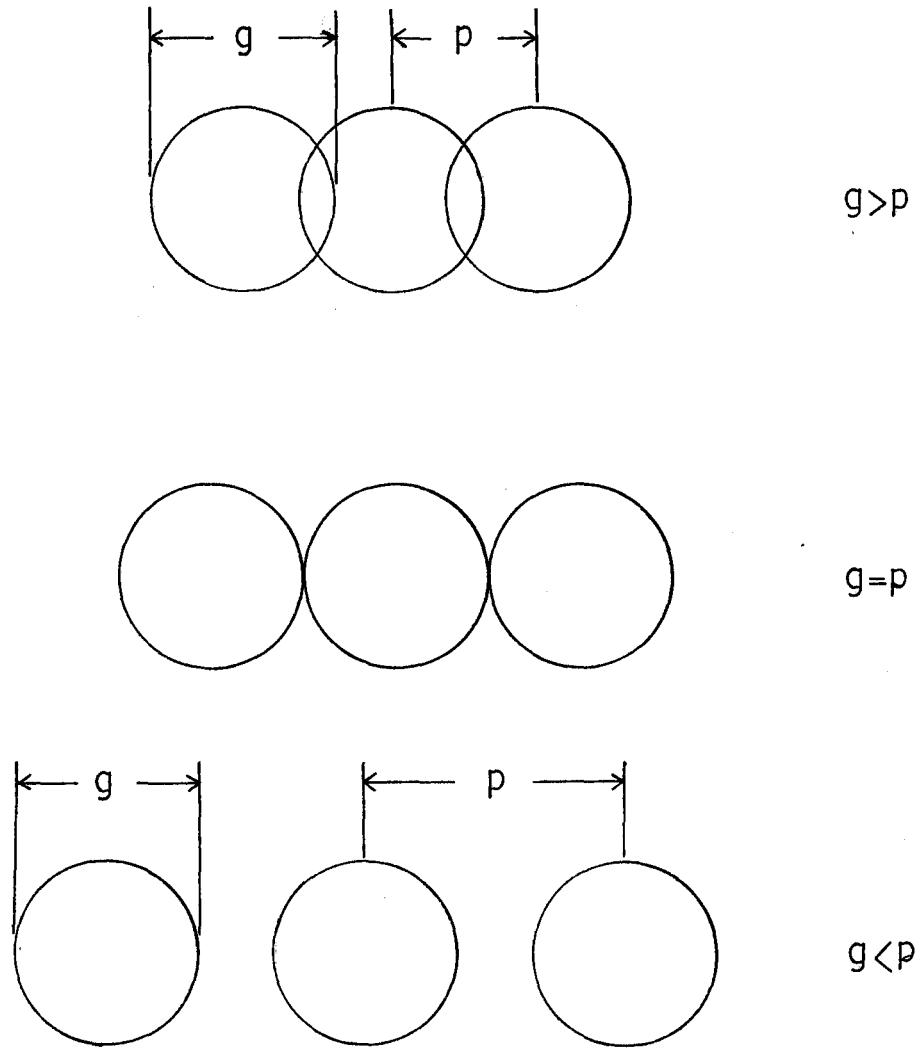
Matematiksel açıdan ve bilgisayar açısından sayısal bir görüntü iki boyutlu bir vektör dizisidir. Diğer bir deyişle her elemanı bir vektör olan bir matristir. Görüntünün iki bağımsız değişkenini "x" ve "y" geometrik boyutları oluşturur. Matris elemanlarının değerleri her (x,y) noktasındaki fiziksel değişkenlerin değerlerini gösterir. Siyah - beyaz bir resim için elemanlar vektör değil skalar değerlerdir.

1.3 Yersel Sayısallaştırma, Piksel, Ağa, Ayırım

Bir resim, görüntü matrisini oluşturmak üzere ızgara biçiminde küçük alanlara bölüldüğü zaman, oluşan her alana bir resim elemanı ya da "piksel" (pixel) denir. Bazı literatürlerde "pel" olarak da geçer. Geometrik ayırım (resolution) kavramı ile yakından ilişkili olan piksel, bir sayısal görüntünün en küçük parçasıdır. Genellikle kare biçimindedir. Yeryüzünde bir piksele karşı düşen alan ne kadar değişik görünüm ve değerlere sahip olursa olsun, pikselin bir spektral banttaki değeri tek bir sayı ile ifade edilir. Bu sayı bir bakıma o alanın ortalama değeri olur. Sayısal bir resim yeterince büyütüldüğünde blok blok

$g = \text{AGA çapı}$

$p = \text{Pikseller arası mesafe}$



ŞEKİL 1.1 AGA-PİKSEL MESAFE İLİŞKİLERİNİN ÜÇ FARKLI DURUMU

görünümleriyle pikseller ortaya çıkar. Buna karşılık sürekli bir resim büyütüldüğünde sadece genel netlik kaybolması olur.

Sayısal bir görüntü, sürekli bir resmin sayısallaştırılmasıyla elde edilebileceği gibi, modern çok bantlı tarayıcılar ve bazı elektronik kameralarla elde edilebilen görüntüler zaten baştan itibaren sayısaldırlar. Sayısal görüntü elde eden bir tarayıcı, yapısı itibariyle bir anda, yeryüzünde, ancak bir küçük alana bakıp oradan spektral değerler okur. Tarayıcı optiğinin açısal ayırım gücünü ifade eden bir kavram "Anlık Görüş Alanı" (AGA; Instantaneous Field of View) dir. Genellikle bir AGA bir piksele karşı düşer. Ancak birkaç AGA birleştirilerek bir piksel oluşturulabileceği gibi, sayısal bir resim yeniden örneklendiğinde ya da geometrik düzeltmeden geçirildiğinde AGA piksel ilişkisi birebir olmaktan çıkar (Şekil 1.1).

AGA bir açı ile ifade edilirken piksel boyutsuzdur. Ancak her ikisinde de yeryüzünde karşı düştüğü bir alan vardır (1).

Parlaklık ya da yayın şiddeti olarak ölçülen değerlerin, bilgisayarda saklanabilmesi için, yine belirli sayıda ayırık değerler biçiminde ifade edilmesi, yani sayısallaştırılması gerekir. Bilgisayarların yapıları dolayısıyla her değer bir baytla ifade edilmesi çok kez uygun görülmektedir. Bir bayt sekiz bitten oluştuğu için ayırdedilebilen parlaklık seviyesi sayısı $2^8 = 256$ olmaktadır. Genellikle en karanlığa ya da sıfır sinyal seviyesine 0, en yüksek sinyal seviyesine de 255 verilir.

1.4 Sayısal Görüntü

Genel anlamıyla bir sayısal görüntü $G(i,j)$ şöyle nitelendirilebilir (1).

$$i=1, \dots, N \quad (1.1)$$

$$j=1, \dots, M \quad (1.2)$$

NxM boyutlarındaki bir görüntü, $G(i,j)$ matrisiyle ve bu görüntünün her pikseli, matrisin bir elemanı ile temsil edilmektedir (1). Matris indeksleri i ve j görüntü piksellerinin yerdeki konum koordinatlarını gösterir. Matrisin her elemanı $g(i,j)$ bir vektördür.

$$g(i,j) = \begin{bmatrix} g_1(i,j) \\ g_2(i,j) \\ \cdot \\ \cdot \\ \cdot \\ g_1(i,j) \end{bmatrix} \quad (1.3)$$

Bu vektörün her elemanı skalar bir değerdir ve i,j koordinatlarındaki belli bir spektral bantta, bir zaman noktasında ölçülen değerini verir. Vektörün diğer elemanları aynı pikselle ilgili öteki ölçümlerden oluşur.

Görüntülerin veri hacmi çok fazla olabileceğinden, işlenmesi de yüksek bilgisayar hızı ve bellek kapasitesi gerektirir. Görüntünün ilk anda depolanması için gereken belleğin yanısıra yapılacak işlemler sonucu ortaya çıkacak yeni yeni görüntüler için ayrıca bu belleğin birkaç katı daha bellek gereklidir.

Bilgisayarla ilgili ikinci bir konu da hızdır. Veri sayısı yüksek olunca, bu kadar veri üzerinde yapılacak işlem de uzun zaman alacaktır.

Bunun için hem kapasite hem de hız açısından, genel amaçlı çok kullanımlı bilgisayar sistemleri, veri sayısı fazla olan çalışmalarda kısıtlayıcı olmakta ve yetersiz

kalmaktadır. Eğitim amacıyla küçük görüntüler kullanarak yapılacak çalışmalarda bu tür sistemler uygun olabilir. Ancak çalışma kapsamı büyüdükçe genel amaçlı sistemlerden, özel yapılmış görüntü işleme sistemlerine bir geçiş gerekmektedir.

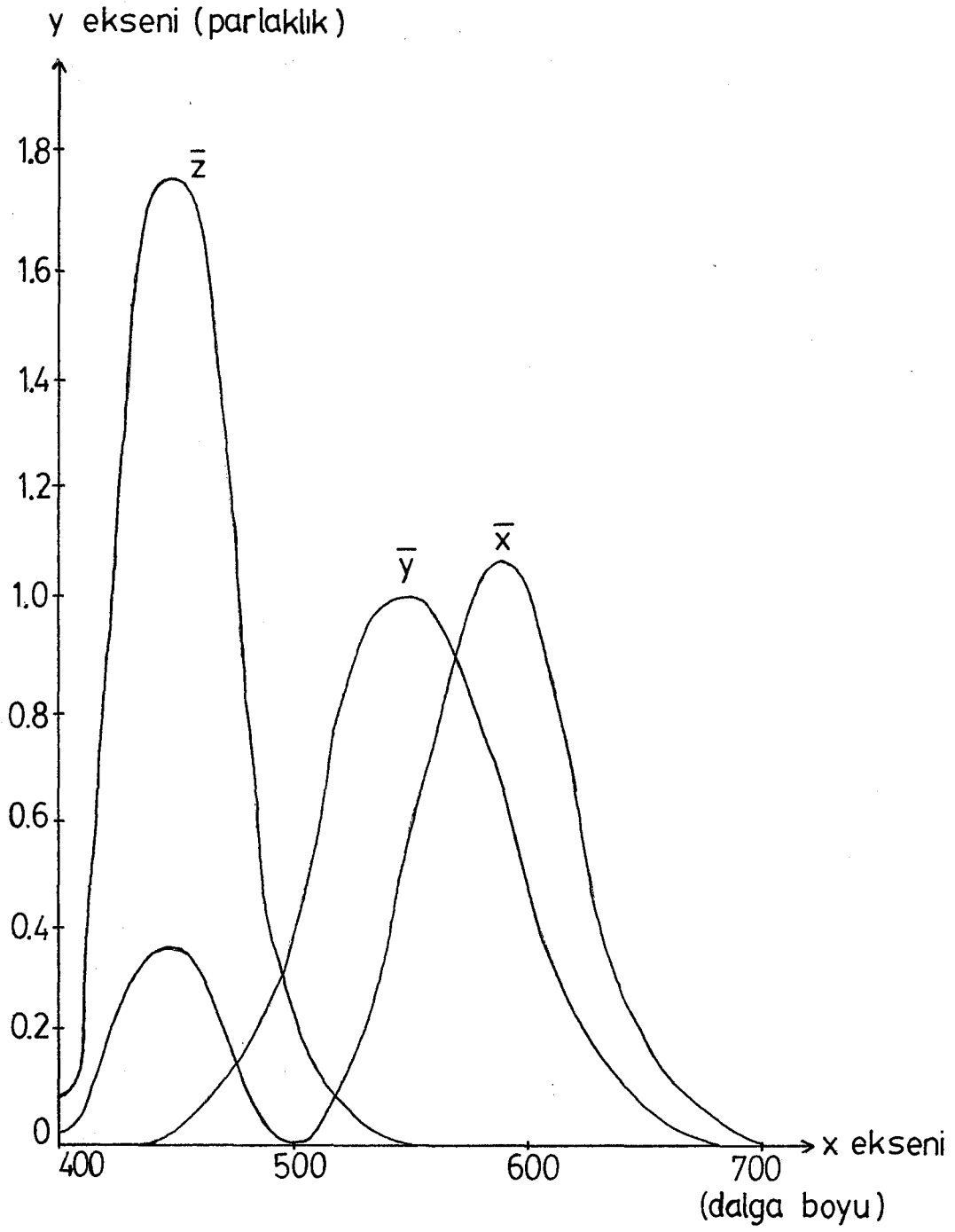
1.5 Görüntü ve Resim

Bir görüntünün bilgisayarla işlenmesi sırasında, mutlaka bazı aşamalarda yorumcu tarafından görülmek üzere bir fotoğrafa, kağıda ya da televizyon ekranına dökülecektir. Resim diyebileceğimiz gözle görünen bu tür görüntülerin taşıdığı bilgi bakımından önemli sınırlamaları vardır. Bilgisayar ortamında saklanan görüntüler için ise bu tür sınırlamalar söz konusu değildir.

Bir resim renkli ya da siyah beyaz olabilir. Siyah beyaz bir resim tek bir spektral banttaki görüntüyü gösterir. Renkli bir resim ise üç spektral bantı içerir. Spektral ayırımı yüksek, çok kanallı tarayıcılar tarafından alınan görüntüler üçten fazla spektral bantı içeriyorsa (bu sayı onbir ya da daha fazla olabilir) bu görüntü renkli resim ortamına sığmaz. Tüm bilgi gözün görebileceği tek bir resim olarak ortaya konamaz.

Siyah-beyaz resimlerde insan gözünün ayırabildiği grilik seviyesi 10-16 arasındadır. Sayısal resimlerde pratik bir sınır olmamakla birlikte genellikle 256 seviye kullanılmaktadır. Üç spektral bantlı bir sayısal görüntü de böylece $256^3 = 16 \times 10^6$ değişik değer gösterilebilir (1).

Renk, görünür tayfdaki bir ışık etkeninin (stimulus) gözde uyandırdığı duygudur. Elektromanyetik tayfın görünür bölgesi yaklaşık 0.38 mikron ile 0.76 mikron arasındadır. Bu iki uç arasında sonsuz sayıda ışık etkeni eğrisi olabilir, ancak iki değişik eğri gözde aynı renk duygusunu da



ŞEKİL 1.2 CIE STANDARTLARINA GÖRE X,Y,Z ANA RENK BİLEŞENLERİ

uyandırabilir. Bu durumda iki eğriye metarık çift denir. Algılanan renk açısından herhangi bir renk üç boyutlu uzayda bir vektör olarak temsil edilebilir. Usayın üç ana rengi olarak hemen hemen her renk üçlüsü (üç değişik renk) temel renkler alınabilir. Seçilen üç rengin değişik miktarlarda toplanmalarıyla öteki renklerden bazıları ya da birçoğu oluşur. Bu üç ana renk kırmızı, yeşil ve mavinin birer tonu olarak seçildiğinde bunlardan toplama (additive) yoluyla oluşacak renklerin sayısı çoğalır ve renklerin büyük çoğunluğu elde edilebilir (1).

Bir sayısal görüntü ile (gözle görülebilen) resim arasındaki önemli farklar şöyle özetlenebilir:

1. Bir resim ancak üç spektral bandı içerebilir, sayısal görüntüde böyle bir sınır yoktur.
2. Üç spektral bantın iki değişik karışımı aynı rengi verebilir.
3. Renk oluşturulmasında kullanılan üç ana renk ile renklerin çoğu oluşturulmasına rağmen, elde edilemeyen renkler de vardır.
4. Sayısal görüntü ile (bilgisayar ortamında) insan gözünün ayırabileceğinden çok daha fazla değer saklanabilir, izlenebilir.

2. GÖRÜNTÜ ZENGİNLEŞTİRME YÖNTEMLERİ

2.1 Giriş

Görüntü zenginleştirme (image enhancement) görüntüyü gözle yoruma daha uygun bir duruma dönüştürmek için yapılır. Amaç bozulmaları düzeltmek değildir, ama çok kez bozulma etkileri kendiliğinden giderilir. Zenginleştirmeyle görüntüde aranan bazı özellikler daha belirgin hale gelir, ancak bunun yanında diğer bazı özellikler bastırılıyor olabilir. Fakat genelinde zenginleştirilmiş görüntü orijinalinden göze daha "hoş", daha "kaliteli" görünür (1).

Özellikle gerçek görüntüler (uzay fotoğrafları ...gibi) ile çalışıldığında çok farklı görüntü zenginleştirme yöntemleri kullanılır. Görüntü zenginleştirme yöntemleri, genelde tek bantlı ve çok bantlı olmak üzere ikiye ayrılabilir. Tek bantta yapılan işlemler noktasal, komşusal ve frekans domeni (point operations, neighborhood operations, frequency domain operations) olarak üç grupta ele alınabilir. Çok bantlı işlemlerde renkli (3 bant) ve daha çok bantlı (>3) biçiminde iki grupta toplanabilir. Görüntü zenginleştirmede kullanılan bazı işlemler şunlardır.

- Komşusal işlemler
- Katlama
- Fourier dönüşümü
- Orta değer süzmesi
- Renk zenginleştirme

2.2 Komşusal işlemler

Özellikle gerçek görüntüler (uzay fotoğrafları ...gibi) üzerinde çalışıldığında, bir pikselin parlaklık değeri, yalnız ona karşı düşen yeryüzü parçasına değil, komşu alanlara da bağlıdır. Bunun sonucunda genellikle görüntü netliğinden kaybeder, keskin hatlar yumuşar, hızlı değişimler (yüksek frekanslar) zayıflar. Elektrik mühendisliği dilinde alçak geçiren bir süzgeç, matematiksel olarak bir yerel entegrasyon uygulanmış olur. Görüntüyü netleştirmek, keskinlik kazandırmak için bunların ters işlemi olan yüksek geçiren bir süzgeç, ya da türev (differentiation) uygulamak gerekir. Bu işlemler pikselin hem kendi değerine hem de komşularının değerine bağlıdır.

Sayısal görüntülerde türev şu şekilde tanımlanır (1).

$i-1, j-1$	$i, j-1$	$i+1, j-1$	(2.1)
$i-1, j$	i, j	$i+1, j$	
$i-1, j+1$	$i, j+1$	$i+1, j+1$	

$$\Delta_x f(i, j) = f(i, j) - f(i-1, j) \quad (2.2)$$

$$\Delta_y f(i, j) = f(i, j) - f(i, j-1) \quad (2.3)$$

Ust mertebeden türevler benzer şekilde hesaplanabilir:

$$\begin{aligned} \Delta_x^2 f(i, j) &= \Delta_x f(i+1, j) - \Delta_x f(i, j) \\ &= f(i+1, j) + f(i-1, j) - 2f(i, j) \end{aligned} \quad (2.4)$$

$$\Delta_y^2 f(i, j) = f(i, j+1) + f(i, j-1) - 2f(i, j) \quad (2.5)$$

Bir görüntünün türevi alındığında elde edilen de bir görüntüdür.

Laplasyanda ise, ikinci türevler bulunmakta olup, sayısal görüntüler için şu şekilde hesaplanır:

$$\begin{aligned}\nabla^2 f(i, j) &= \Delta_x^2 f(i, j) + \Delta_y^2 f(i, j) \\ &= f(i+1, j) + f(i-1, j) + f(i, j+1) + f(i, j-1) - 4f(i, j)\end{aligned}\quad (2.6)$$

Değişik tanımlara göre değişik laplasyan formülasyonları bulunabilir. Fakat genelinde laplasyon bir noktadaki fonksiyon değerinin komşusal bir ortalaması ile olan farkına orantılıdır.

2.3 Katlama (Convolution)

Katlama (convolution), $M \times N$ boyutlu iki sayısal fonksiyon arasında şöyle gösterilir ve tanımlanır.

$$g(i, j) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(i-m, j-n) \cdot d(m, n) \quad (2.7)$$

Görüntü işlemede fonksiyonlardan biri orijinal görüntü diğeri de süzgeç fonksiyonudur. ikisinin katlanmasıyla süzölmüş görüntü ortaya çıkar (1).

Katlama pratikte verilen süzgeç fonksiyonunun resim üzerinde gezdirilmesinden ibarettir.

Örnek:

$$\text{Görüntü } d = \begin{bmatrix} 0 & 0 & 0 & 5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 0 & 0 \\ 5 & 5 & 5 & 5 & 5 & 5 & 5 \\ 0 & 0 & 0 & 5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 0 & 0 \end{bmatrix}$$

$$\text{Süzgeç } f = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

$$f^t = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} = f$$

Önce f 'nin $(1,1)$ 'lik elemanı d 'nin $(1,1)$ 'lik elemanının üstüne çakışacak gibi oturtulur. Üstüste gelen elemanlar çarpılıp toplanır. Bunun sonucu $0x1+0x1+0x1+0x1=0$ dır. Çıktı görüntüsünün $(1,1)$ 'lik elemanının değeri 0 olur. Sonra f,d üzerinde bir adım sağa kaydırılır ve yeniden çakışan elemanlar çarpılıp, toplanır, $d(1,2)$ için sonuç 0 çıkar. $d(1,3)$ ve $d(2,3)$ için ise,

$$d(1,3)=0x1+0x1+5x1+5x1=10$$

$$d(2,3)=5x1+5x1+0x1+0x1=10$$

elde edilir.

Bu şekilde ilerlendiğinde sonuç $d' = d * f =$

$$\begin{bmatrix} 0 & 0 & 10 & 10 & 0 & 0 \\ 0 & 0 & 10 & 10 & 0 & 0 \\ 10 & 10 & 15 & 15 & 10 & 10 \\ 10 & 10 & 15 & 15 & 10 & 10 \\ 0 & 0 & 10 & 10 & 0 & 0 \\ 0 & 0 & 10 & 10 & 0 & 0 \end{bmatrix}$$

olarak çıkar (4).

Çıktı görüntünün boyutları 6x6 dır. Orijinal görüntü ise 7x7 idi. Genelde görüntüler çok büyük, süzgeçler çok küçük olduğundan, meydana gelen küçülme farkedilmez veya önemsenmez. Kullanılan süzgeç dört elemanın toplamlarını alan bir işlem olarak, alçak geçiren bir süzgeçtir.

Görüntü işleme dilinde süzgeç matrislerine, çekirdek (kernel) de denir. Değişik görüntü zenginleştirme etkileri ve matematiksel işlemler için değişik çekirdekler kullanılabilir.

Katlama işlemi türev amaçlı ise çekirdekler:

$$\Delta_x^2 f \text{ için } \begin{bmatrix} 1 & -1 \end{bmatrix}$$

$$\Delta_y^2 f \text{ için } \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

olacaktır.

Katlama işlemi ile laplasyan (laplacian) almak için kullanılan bazı çekirdek matrisler şöyledir:

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 4 & 1 \\ 4 & -20 & 4 \\ 1 & 4 & 1 \end{bmatrix}$$

Görüntü laplasyan kullanılarak zenginleştirilmek istendiğinde, orijinal görüntü ile laplasyanın belirli bir oranı toplanır. Bu iki işlem tek bir katlama işleminde birleştirilebilir. Yukarıdaki üç çekirdek ile $k_2 = 1/2$ oranını kullanarak yapılacak uygulamada şu üç çekirdek kullanılır.

$$\begin{bmatrix} 0 & -1/2 & 0 \\ -1/2 & 3 & -1/2 \\ 0 & -1/2 & 0 \end{bmatrix}, \begin{bmatrix} -1/2 & -1/2 & -1/2 \\ -1/2 & 5 & -1/2 \\ -1/2 & -1/2 & -1/2 \end{bmatrix}, \begin{bmatrix} -1/2 & -2 & -1/2 \\ -1/2 & 11 & -2 \\ -1/2 & -2 & -1/2 \end{bmatrix}$$

Katlama kullanıldığında başka bir amaç da düzgünleştirmedir. Düzgünleştirme (smoothing) işlemleri genellikle görüntüde bulunan gürültüyü azaltmak için kullanılır. Bu işlemler netliği azaltıcı olduklarından mümkünse ancak gürültünün bulunduğu yerlerde uygulanması uygundur. Bu tür işlemlerin ikinci bir kullanım alanı da dolaylı olarak netliği arttırmaktır. Bu fonksiyonda düzgünleşmiş görüntünün bir kesimini orijinal görüntüden çıkararak elde edilebilir (1).

$$f^i = f - kf_D \quad (2.8)$$

Burada f_D düzgünleştirilmiş görüntüyü ifade eder.

2.4 Fourier Dönüşümü

Bir görüntünün Fourier dönüşümü onun içindeki yersel (spatial) frekansların varlığını gösterir (1).

$M \times N$ boyutlarında bir görüntünün ayrık Fourier dönüşümü şöyle verilir:

$$F(u, v) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) e^{-2\pi j \left(\frac{mu}{M} + \frac{nv}{N} \right)} \quad \begin{array}{l} u=0, 1, 2, \dots, M-1 \\ v=0, 1, 2, \dots, N-1 \end{array} \quad (2.9)$$

$$j = \sqrt{-1}$$

Ters Fourier dönüşümü ile tekrar $f(m, n)$ elde edilir.

$$f(m, n) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) e^{2\pi j \left(\frac{mu}{M} + \frac{nv}{N} \right)} \quad (2.10)$$

Fourier dönüşümü bir görüntüyü netleştirmek ya da onda mevcut bazı periyodik özellikleri ortaya çıkarmak için kullanılabilir. Fourier dönüşüm görüntüsü uygun bir süzgeç

fonksiyonu ile çarpılıp ters dönüşüm alındığında, orijinal görüntüdeki bazı yersel frekanslar vurgulanmış bazıları da bastırılmış olarak ortaya çıkar. Görüntüde elde edilmek istenen etkiye göre uygun süzgeç görüntüleri bulunarak, yüksek geçiren bir süzgeç ile netleştirme elde edilebilir.

2.5 Orta Değer Süzmesi (Median Filtering)

Orta Değer Süzmesi yöntemi kullanılarak, nokta türü gürültüler tamamen giderilirken, kenarlar olduğu gibi korunur. Bu yöntem gereğince, önceden tanımlanan 3x3 ya da 5x5 gibi bir yakın çevre içindeki piksel değerleri, büyüklüklerine göre sıralanır. Ortadaki değer, yeni pikselin değeri olur. Orta değer süzmesi, çok kullanışlı bir yöntem olmasına karşılık yine de birtakım sorunlar bulunmaktadır. Görüntüdeki tek piksel kalınlığındaki çizgiler ve köşeler kaybolmaktadır. Amaca göre kullanılan çekirdeğin biçimi değiştirilerek bazı sorunlar giderilebilir. Örneğin, kare yerine + biçiminde bir çekirdekle yatay ve dikey ince çizgiler ve keskin köşeler korunmuş olur (1).

Herhangi bir yakın çevre içindeki pikseller büyüklük sırasına dizilince, orta değer yerine başka bir değer de alınabilir. Yakın çevrede enaza yakın bir değer almak parlak nesnelere küçültür. Ençoka yakın bir değer almak ise parlak nesnelere büyültür. Karanlık bir fon üzerinde kaybolabilecek parlak bir yol en büyük değer alınarak daha görünür hale getirilebilir.

2.6 Renk Zenginleştirme (Hue Enhancement)

Bir renkli kısım kırmızı (K), yeşil (Y) ve mavi (M) olmak üzere üç bileşenden oluşur. Renk tonlarının daha doygun ve ayırık biçime getirilmesi için şu yöntem uygulan-

bilir. (i,j) konumundaki bir pikselin kırmızı, yeşil ve mavi bileşenleri $P_K(i,j)$, $P_Y(i,j)$ ve $P_M(i,j)$ ile gösterilsin:

$$P_S(i,j) = P_K(i,j) + P_Y(i,j) + P_M(i,j) \quad (2.11)$$

$$P'_K(i,j) = \left[\frac{P_K(i,j)}{P_S(i,j)} \right] \quad (2.12)$$

$$P'_Y(i,j) = \left[\frac{P_Y(i,j)}{P_S(i,j)} \right] \quad (2.13)$$

$$P'_M(i,j) = \left[\frac{P_M(i,j)}{P_S(i,j)} \right] \quad (2.14)$$

P'_K , P'_Y ve P'_M den oluşan resim tamamen doyuma ulaşmış renklerden oluşmaktadır (4). Bunu daha doğal bir biçimde göstermek için belli bir oranı orijinal görüntüye ekleyerek yeni görüntü ortaya çıkar. Bu oran %10 ile %50 arasında değişebilir (4).

$$P''_K(i,j) = P_K(i,j) + kP'_K(i,j) \quad (2.15)$$

$$P''_Y(i,j) = P_Y(i,j) + kP'_Y(i,j) \quad (2.16)$$

$$P''_M(i,j) = P_M(i,j) + kP'_M(i,j) \quad k \approx 0.25 \quad (2.17)$$

3. GÖRÜNTÜ KENARLARININ BULUNMASI

3.1 Giriş

Bir görüntünün kenarları, görüntü siyah-beyaz ise gri seviyelerinin, renkli ise renk değerlerinin ani ya da yavaş yavaş değişiklik gösterdiği yerlerdir. Görüntü kenarları, görüntüde göze çarpmayan ayrıntıların ortaya çıkarılması bakımından önem taşır. Görüntü kenarlarının bulunmasında pek çok farklı yöntem kullanılmaktadır (2).

3.2. Yönlü Laplace Değişim Maskeleriyle Görüntü Kenarlarının Bulunması

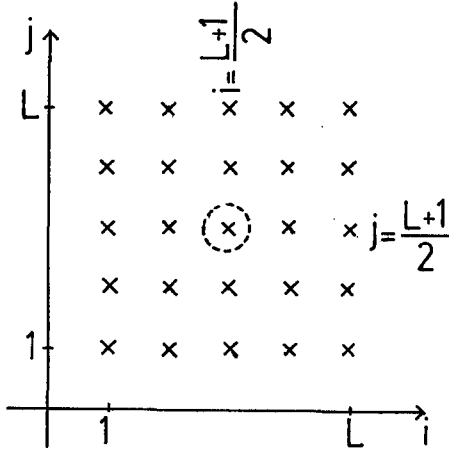
Görüntü kenarlarının bulunmasında kullanılan yöntemlerden biri de değişim maskelerinin kullanılmasıdır. Bu yöntem gereğince, görüntü önce değişim maskeleriyle çarpılarak kenar değişim resmi elde edilir. Sonra bu resme eşikleme uygulanarak görüntü kenar resmi bulunur (2).

Bu yöntemde, Laplace değişim maskeleri parçalanarak yönlü Laplace değişim maskeleri elde edilmiştir. Bu maskelerin Laplace değişim maskelerine göre gürültüsüz ve çok iyi sonuçlar verdiği örnekler üzerinden gözlemlenmiştir. Bu maskeler görüntü kenarlarının bulunmasında iyi sonuçlar veren Prewitt ve Sobel maskeriyle karşılaştırılmış ve bunların en az adı geçen maskeler kadar iyi sonuçlar verdiği gözlemlenmiş, öte yandan da icra sürelerinin (execution time) birbirlerine ± 0.25 zaman birimi yakınlıkta olduğu anlaşılmıştır.

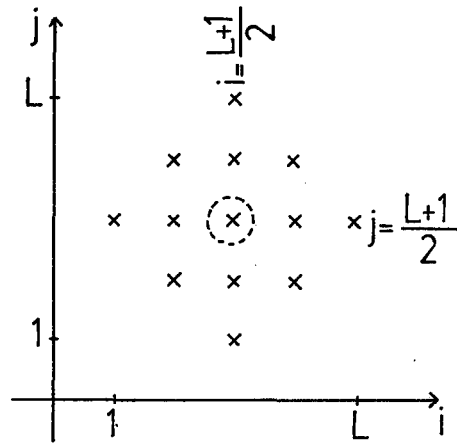
3.2.1 Değişim maskeleri

Değişim maskeleri eni ya da köşegeni L olan kare ya da çapraz biçiminde, Şekil 3.1'de gösterilen noktalar dizisi görünümündedir.

LxL boyutlarında L^2 ya da $(L+1)/2$ elemana bölünmüş bir Z değişim maskesi gözönüne alalım (2).



Şekil 3.1.a Kare
değişim maskesi
(L=5)



Şekil 3.1.b Çapraz
değişim maskesi
(L=5)

Şekil 3.1 Değişim maskeleri

Z kare biçiminde ise denklem (3.1), çapraz biçiminde ise denklem (3.2) ile tanımlanan bir matristir.

$$Z^k = [Z_{ij}^k] \quad 1 \leq i \leq L, \quad 1 \leq j \leq L \quad (3.1)$$

$$Z^c = [Z_{ij}^c] \quad 1 \leq i \leq L, \quad 1 + |i - (L+1)/2| \leq j \leq L - |i - (L+1)/2| \quad (3.2)$$

Görüntü üzerinde herhangi bir nokta X olsun. Bu noktanın etrafında eni ya da köşegeni L olan, Şekil 3.2'de gösterilen kare ya da çapraz biçiminde P penceresi tasarlansın (2).

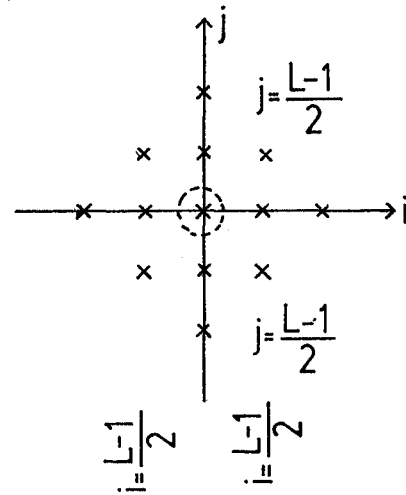
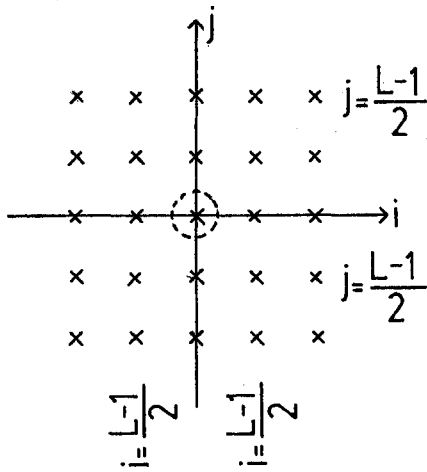
$$P^k = (1, n | |1|, |n| \leq (L-1)/2) \quad (3.3)$$

$$P^c = (1, n | |1| \leq (L-1)/2, |n| \leq (L-1)/2 - 1) \quad (3.4)$$

Çapraz pencere $L > 3$ için Şekil 3.2.b'de görüldüğü gibi 45° döndürülmüş bir kare görünümündedir.

Kare ya da çapraz pencere içindeki görüntü parçasının yönsüz ya da belirli bir yöndeki z değişim maskesiyle çarpımı, x noktasındaki d kenar değişim miktarını belirler.

$$d_{ij} = \sum_l \sum_n x_{i+l, j+n} z_{l+\frac{L+1}{2}, n+\frac{L+1}{2}} \quad (3.5)$$



Şekil 3.2.a Kare pencere
(L=5)

Şekil 3.2.b Çapraz pencere
(L=5)

Şekil 3.2 Kare ya da çapraz pencereler

Denklem (3.5)'deki x , n , l ve z parametreleri (3.1-4) bağıntılarıyla tanımlanmıştır. Bu değişim maskeleri aşağıdaki biçimlerde tanımlanabilirler:

Laplace değişim maskelerinin parçalanarak yönlü bileşenlere ayrılması Tablo 3.1'de gösterilmiştir.

Tablo 3.1'de görüldüğü gibi, 1 nolu maskenin parçalanmasında kuzey, doğu, güney ve batı yönlerine ağırlık verilmiş (bu yönlerdeki farkın iki misli alınmış), 2 nolu maskenin parçalanmasında ise her yöndeki farka eşit davranılmıştır. 3 nolu maske çaprazdır ve sadece kuzey,

doğu, güney ve batı yönlerine ayrılmıştır. Yönlü Laplace maskeleri yardımıyla, kenar değişim miktarı her yöndeki mutlak farkın toplanmasıyla belirlenmektedir.

Laplace değişim maskeleriyle çarpım için (3.2) bağıntısı, yönlü Laplace değişim maskeleriyle çarpım için ise, maske 1 için (3.6), maske 2 için (3.7) ve maske 3 için (3.8) bağıntısı verilebilir.

Tablo 3.1 Yönlü Laplace değişim maskeleri

	MASKE 1	MASKE 2	MASKE 3
	$\begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$
MASKE			
BİLEŞEN	MASKE	MASKE	MASKE
YÖNLERİ	BİLEŞENLERİ	BİLEŞENLERİ	BİLEŞENLERİ
KUZEY	$\begin{bmatrix} 0 & -2 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & -1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & -1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$
KUZEY DOĞU	$\begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
DOĞU	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & -2 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$
GÜNEY DOĞU	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$	
GÜNEY	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & -2 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 0 \end{bmatrix}$

$$\begin{array}{l}
\text{GÜNEY} \\
\text{BATI} \\
\text{BATI} \\
\text{KUZUY} \\
\text{BATI}
\end{array}
\begin{array}{c}
\begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \\
\begin{bmatrix} 0 & 0 & 0 \\ -2 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\
\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\
\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\
\begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}
\end{array}
\begin{array}{c}
\begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}
\end{array}$$

$$d_{ij} = \sum_{l=-1}^1 \left(\sum_{n=|l|}^{|l|} |x_{ij} - x_{i+l, j+n}| + \sum_{n=-1}^1 |x_{ij} - x_{i+l, j+n}| \right) \quad (3.6)$$

$$d_{ij} = \sum_{l=-1}^1 \sum_{n=-1}^1 |x_{ij} - x_{i+l, j+n}| \quad (3.7)$$

$$d_{ij} = \sum_{l=-1}^1 \sum_{n=|l|-1}^{|l|} |x_{ij} - x_{i+l, j+n}| \quad (3.8)$$

Prewitt ve Sobel maskeleri Tablo 3.2'de verilmiştir. Bu maskeler, üç ve beş seviyeli yönlü basit maskelerin kuzey (güney) ve batı (doğu) yönlerinden oluşmaktadır. Karşı yöndeki maskeler, o yöndeki maskelerden -1 çarpanı ile ayrılmaktadır. Kenar değişim miktarını hesaplamak için mutlak değişim miktarı göz önüne alındığı için dört maske yerine iki maske kullanılır.

Tablo 3.2 Prewitt ve Sobel maskeleri

MASKE	MASKE	MASKE
YÖNLERİ	4	5
KUZUY (GÜNEY)	$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$	$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$
BATI (DOĞU)	$\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$

Bu maskelerle kenar deęişim miktarını hesaplarken,

$$\sqrt{x+y} \ll x+y \quad (3.10)$$

eşitsizliğinden faydalanılmıştır. Kenar deęişim miktarı (K:KUZEY, B:BATI) (3.11) baęıntısıyla belirlenmiştir.

$$d_{ij} = \left| \begin{array}{cc} 1 & 1 \\ \sum_{l=1}^K & \sum_{n=1}^B x_{i+l, j+n} \end{array} \right| + \left| \begin{array}{cc} 1 & 1 \\ \sum_{l=1}^K & \sum_{n=1}^B x_{i+l, j+n} \end{array} \right| \quad (3.11)$$

3.2.2 Kenar deęişim resmi

Bir görüntünün her noktasında hesaplanan kenar deęişim miktarları, kenar deęişim resminin elemanlarını oluşturur. Kenar deęişim resminde kenarlar orijinal resme göre daha belirgin hale geldięi için bu işleme kenar zenginleştirme adı da verilir.

X görüntüsü NxM boyutlarında olsun ve NM görüntü elemanına bölünsün. X görüntüsü NxM'lik bir matrisle tanımlanabilir.

$$X=[x_{ij}] \quad 1 \leq i \leq M, \quad 1 \leq j \leq N \quad (3.12)$$

D kenar deęişim resminin her d elemanı, X görüntü matrisinin her x $L-2 \leq i \leq M$, $L-2 \leq j \leq N$ noktası civarında oluşturulan kare ya da çapraz pencerenin, Z kare ya da çapraz deęişim matrisiyle çarpılmasıyla hesaplanır. D, böylece (3.13)'de verilen $M-L+1 \times N-L+1$ boyutlarında bir matrisle tanımlanır.

$$D=[d_{ij}] \quad L-2 \leq i \leq M, \quad L-2 \leq j \leq N \quad (3.13)$$

3.2.3 Eşikleme

Kenar deęişim resminden yararlanarak kenar resmini elde etmek için yapılan işleme eşikleme denilir. Eşikleme ya bütün ya da yerel bir bölge içindeki, kenar deęişim resmi elemanlarını gözönüne alarak yapılır. Eşikleme de, buna göre bütünsel ya da yerel eşikleme adını alır (2).

Bütünsel eşikleme, kenar deęişim resminin en çok deęişen elemanlarının kenar olarak seçilmesini sağlar. Eşik olarak belirlenen kenar deęişim miktarına eşit ya da daha çok deęişen noktalar görüntünün kenarlarını oluşturur. Eşiğin altında kalan elemanların toplamı, kenar deęişim resmi elemanlarının belirli bir ϵ yüzdesine eşittir.

Bütünsel eşikleme, kenar deęişiminin az olduęu bölgelerdeki kenarları eler, çok olduęu bölgelerde ise kenarların kümeleşmesine neden olur.

Yerel en çok nokta, yerel bölgenin konumuna göre deęiştii için yerel eşikleme kenar deęişiminin az olduęu yerlerdeki kenarları ortaya çıkartır. Kenar deęişiminin çok olduęu yerlerde ise, kenar deęişim resminin bütününe göre çok fakat yerel pencere içinde az deęişen kenarları eler.

Yerel eşik, $L \times L$ boyutlarında bir kare ya da çapraz pencere içindeki kenar deęişim resmi elemanlarının m 'inci en çok deęişen noktasına eşitlenir.

Bütünsel ve yerel eşiklemenin bir arada kenar deęişim resmine uygulanmasıyla her iki eşiklemenin iyi yönlerinden yararlanılır. Şöyle ki, kenar deęişiminin az olduęu yerlerdeki kenarlar silinir. Böylece bu bölgelerin kenar deęişiminin çok olduęu bölgelere eşit ölçüde kenar resmine yansması önlenir. Bütünsel eşiğin üzerinde fakat yerel

eşiğin altındaki kenarların elenmesiyle kenar değişiminin çok olduğu bölgelere kenar yığılması önlenir, hatlarda incelik sağlanır.

3.2.4 Kenar resmi

Kenar değişim resminin eşik üzerinde kalan elemanları kenar resmini oluşturur (2).

D, kenar değişim resmi $N \times M$ boyutlarında bir matrisle tanımlansın.

$$D = [d_{ij}] \quad 1 \leq i \leq M, \quad 1 \leq j \leq N \quad (3.14)$$

Kenar resmi, bütünsel eşikleme için (3.15)'deki boyutlarda, yerel (ve bütünsel) eşikleme için $M-L+1 \times N-L+1$ boyutlarında bir K matrisiyle tanımlanır. Bu matrisde kenar bulunan noktalar 1, diğer noktalar 0 sayısıyla gösterilmektedir.

$$K = [k_{ij}] \quad L-2 \leq i \leq M, \quad L-2 \leq j \leq N \quad (3.15)$$

Burada, L yerel eşikleme yapılan pencerenin boyutudur. e_{ij} yerel eşik ise,

$$k = \begin{cases} 1 & \text{eğer } d_{ij} \geq e_{ij}, \quad L-2 \leq i \leq M, \quad L-2 \leq j \leq N \\ 0 & \text{ters halde} \end{cases} \quad (3.16)$$

$e_{ij} = e$ bütünsel eşik ise,

$$k = \begin{cases} 1 & \text{eğer } d_{ij} \geq e, \quad 1 \leq i \leq M, \quad 1 \leq j \leq N \\ 0 & \text{ters halde} \end{cases} \quad (3.17)$$

e_{ij} yerel eşik ise, e bütünsel eşik ise,

$$k = \begin{cases} 1 & \text{eğer } d_{ij} > e_{ij} \text{ ve } e_{ij} \\ 0 & \text{ters halde} \end{cases}, \quad 1 \leq i \leq M, \quad 1 \leq j \leq N \quad (3.18)$$

3.2.5 Gürültü eleme

Kenar resminde bulunan bazı kenarlar ötekilerinden yalıtılmıştır. Çevrelerinde hiçbir kenar bulunmamaktadır. Bunlar muhtemelen gürültü kenarlarıdır ve elenmesi gerekmektedir.

K , kenar resminin $N \times M$ boyutlarında bir matrisle tanımlandığını varsayalım.

$$K = [k_{ij}] \quad 1 \leq i \leq M, \quad 1 \leq j \leq N \quad (3.19)$$

3×3 boyutlarında bir kare pencerenin merkezini muhtemel kenarlarla karşılaştırarak, komşu kenarları sayalım. Eğer komşu kenar yoksa, o kenar saymada hesaba katılmayacaktır.

Gürültüsüz kenar resmi, G , $N-2 \times M-2$ boyutlarında bir matrisle tanımlanabilir.

$$G = [g_{ij}] \quad 1 \leq i \leq M-1, \quad 1 \leq j \leq N-1 \quad (3.20)$$

$$\text{Burada, } g_{ij} = \begin{cases} 1 & \text{eğer } \sum_{x=-1}^1 \sum_{y=-1}^1 k_{i+x, j+y} > 1 \\ 0 & \text{ters halde} \end{cases} \quad (3.21)$$

3.3 Filtreler Kullanarak Görüntü Kenarlarının Bulunması

Bazı durumlarda, bir nesnenin sadece kenarlarının belirlenmesi ya da bir resmi oluşturan çeşitli şekillerin renklerinin (ya da gri seviyelerinin) aynı taşınamaksızın sadece resim üzerindeki orijinal renklerinin belirlenmesi istenebilir. (3.22)'de gösterilen sınır bulma yöntemi, noktaların (pixel) komşuluklarındaki noktalarla aralarındaki renk farklılıklarını hesaplayarak yeni bir resim oluşturur (3).

A	B	C
D	*	E
F	G	H

*=incelenen nokta
bölgesi

Şekil 3.3 Görüntü kenarlarının bulunmasında kullanılan filtre

$$\text{yeni nokta değeri} = \frac{|A-H| + |C-F| + |B-G| + |D-E|}{4} \quad (3.22)$$

4

Buna göre bir görüntüde parlak ve karanlık iki alan arasında keskin bir geçiş olarak parlak bir çizgi oluşacaktır. Renklerde ya da gri seviyelerinde herhangi bir değişiklik olmayan, yani komşuluklarında farklı renkler bulunmayan alanlar ise siyaha dönüştürülür. Orijinal görüntüde siyah olan yerler ise önemli değildir. Sadece farklı renkler arasındaki geçişler dikkate alınmaktadır.

Bu yöntemle görüntü kenarlarını bulmak, çeşitli kıyaslamalarla bir biti daha karmaşık bir şekilde elde etmektir. incelenen noktanın etrafında, dört yöndeki komşuluk değerleri arasındaki farkların mutlak değerleri bulunur. Yani yatay, dikey ve iki çapraz yöndeki sekiz

nokta(pixel) arasındaki fark değerleri hesaplanır. Aslında çok karmaşık olan yöntemin basit bir uygulaması olan bu işlem "convolution" olarak adlandırılır (3).

Yukarıda belirtilen işlemde sonuca ulaşma zamanını azaltmak için, üzerinde işlem yapılan nokta sayısı altıya indirilebilir. Buna göre 3'e 3'lük iki nokta grubu alınarak bunların karşılıklı gelen değerleri arasında bir karşılaştırma işlemi uygulanır. Bu yöntemle şeklin kenarlarının belirlenmesi işlemi de iyi sonuç vermektedir. Kenarları daha hassas ve dikkatli belirlemek amacıyla üzerinde işlem yapılan nokta grupları, 45° derece çapraz köşegenlerdeki değerler de katılarak 5'e 5'lik iki nokta grubuna dönüştürebilir. Ayrıca uygulanan işlemde mutlak değer fonksiyonu kaldırılarak sınırların yönleri de belirlenebilir. Buna örnek olarak 3.23'deki formülasyon verilebilir.

$$\text{Kenar} = \frac{(A-F) + (B-G) + (C-H)}{3} \quad (3.23)$$

Buna göre incelemeye alınan 3'e 3'lük iki nokta grubundan üst noktaları parlak, alt noktaları karanlık farklı iki nesne arasında yatay bir sınır belirtecek ve işlem sonucu da pozitif bir değer olacaktır. Bunun tersi bir pozisyonda iki nesne düşünüldüğünde ise elde edilecek işlem sonucu negatiftir. Kenar bulma işleminde, sonuç incelenen nokta değerlerine göre değiştiğinden sonuçta yeni nokta değerini elde etmek için sabit bir ofset değerini işlem sonucuyla toplamak gerekmektedir. Bütün sınırların ekranda görülebilmesi için, sınır olarak hesaplanan değer genellikle iki ya da üçe bölünür. Ayrıca bu sayede bütün yumuşak kenarlar bastırılabilir.

(3.24)'de verilen formülle ise Şekil 3.3'deki gibi, incelenen noktanın komşuluklarındaki dört noktanın ortalamasını alan bir filtre kullanılarak çıkış resmi elde edilir.

$$\text{Yeni nokta değeri} = \frac{B+D+E+G}{4} \quad (3.24)$$

(3.24)'deki işleme göre çıkış resminde sınırlar büyük ölçüde kaldırıldığı için, daha çok detayları iyi bir şekilde belli olan nesnelere uygulanabilir.

4. SAYISAL GÖRÜNTÜ İŞLEME SİSTEMLERİ

4.1 Giriş

Sayısal görüntü işleme önceleri genel amaçlı bilgisayar sistemlerinde başlamış, ancak bunların birçok bakımdan yetersiz kalması sonucu özel donanım, yazılım ve yan birimler içeren "Sayısal Görüntü İşleme Sistemleri" (GIS) ortaya çıkmıştır (1).

4.2 Donanım

Bir GIS, giriş, çıkış, işlem ve saklama birimlerinden oluşur. Bir kullanıcı açısından sistemin yerine getirmesi gereken işlemler de bu dört başlık altında ele alınabilir. Bunlara girmeden önce görüntü işlemenin iki özelliğini ortaya koymak gerekir.

1. Veri hacmi yüksektir. Milyonlarca pikselle çalışmak olağandır.

2. İşlemler genellikle az sayıda basit işlemin tekrar tekrar her piksele uygulanması biçiminde ele alınabilir.

Kullanıcı, giriş birimleri kanalıyla sisteme alfanümerik bilgi ve komut verir, görüntüdeki belli piksel ve alanları işaret eder ve sisteme görüntü verir. Bu amaçlar için klavye (keyboard), izleme topu (tracball), tablet, sayısallaştırıcı (digitizer), kamera, film okuyucu ve manyetik teyp birimi gibi cihazlar kullanılır.

Çıkış birimlerinin amacı alfanümerik bilgi vermek ve görüntüyü göstermektir. Birinci amaç için standard bilgisayar birimleri olan terminal ve satır yazıcılar kullanılır. Görüntü ortamları da renkli ortam ve fotoğraf

olarak ikiye ayrılır. Renkli ekran görüntüyü anında gösteren esnek, hızlı fakat kalıcı olmayan bir ortamdır. Buna karşılık fotoğraf üretilmesi uzun süren fakat kalıcı bir ortamdır.

işlem birimleri açısından kullanıcının istediği hızdır. istediği bir işlemin sonucunu mümkünse anında (bir iki saniye içinde) görmek ister. Anlık cevap alınmazsa bazı işlemlerin etkinliği ve anlamı kalmaz, bilgi kaybolur. Bu nedenle genel amaçlı bilgisayarlar hız açısından yetersiz kalmaktadır.

Görüntü işlemede hız sağlamak için gösterge işlemci (display processor) ve dizi işlemci (array processor) denen özel bilgisayar yan birimleri kullanılmaktadır. Gösterge işlemcilerin önemli bir görevi görüntüyü renkli ekranda göstermektir. Bunun yanında görüntü ile etkileşimli çalışmayı gerektiren, işlemleri hızlı yapan özel elektronik devreleri de vardır. Dizi işlemciler daha genel amaçlı cihazlar olup birçok bilimsel uygulamalarda kullanılırlar.

4.3 Yazılım

Diğer alanlarda olduğu gibi görüntü işlemede de bir bilgisayar sisteminin etkinliği ve yararı, en az donanım kadar yazılımın da kalitesi ve gücüne bağlıdır. Yazılım geniş kapsamlı, büyümeye açık, esnek, kullanımı kolay, iyi belgelenmiş, zengin hata mesajlı, hızlı çalışan, optimal kodlanmış ve derlenmiş,... olmalıdır.

Bir GİS yazılım paketinin yapması gereken işlemler şöyle gruplanabilir:

1. Temel işlevler: Sisteme giriş-çıkış, kontrol, dosya yönetimi veri türü dönüşümleri gibi, daha çok bilgisayar işletim sistemi işlevleri.

2. Gösterge işlevleri: Ekran üzerinde değişik bant karışımları, büyültme (zoom), dolaşma (pan), kursör kontrol, renk kontrol v.b.

3. Görüntü işleme: Matematiksel formülasyona bağlı, zenginleştirme, sınıflandırma gibi her tür görüntü işlem işlevleri.

4. Görüntü saklama: Görüntü aktarma, saklanan görüntüler için arşiv tutma, v.b. işlevler.

5. Görüntü dışı veri dosyaları işlevleri: Görüntü geçmişi dosyaları, istatistik dosyaları, grafik dosyaları, tablo-liste dosyaları, çalışma dosyaları (session files), v.b. tutma işlevleri.

5. TÜRK HALI DESEN VE MOTİFLERİNİN BİLGİSAYAR YARDIMIYLA SAKLANMASI, GELİŞTİRİLMESİ VE ÇOĞALTILMASI AMACIYLA GELİŞTİRİLMİŞ BİR YÖNTEM

5.1 Giriş

Bu bölümde, halılarımızın desen ve renk açısından zenginleştirilmesini, desenlerin ucuz ve hızlı olarak çoğaltılıp yaygın kullanıma sunulmasını ve Türk el halıcılığı arşivinin düzenlenmesini sağlamak amacıyla geliştirilmiş olan bir program anlatılacaktır (4). Bu program, Türk halıcılığının yaşatılması, geliştirilmesi ve yaygınlaştırılması açısından da önemlidir.

Yapılan araştırmaya göre Türk halıcılığının iki önemli sorunu vardır:

1. Eldeki desenlerin azlığı ve yeni desenlerin geliştirilmesinde karşılaşılan güçlükler,
2. Desenlerin çoğaltılmasında karşılaşılan güçlükler.

Ülkemizin çeşitli yörelerine yayılmış halı desenlerinin bir merkezde toplanması, zenginleştirilmesi ucuz ve hızlı olarak çoğaltılıp yaygın kullanıma sunulması ve gelecek nesillere sağlıklı olarak aktarılabilmesi için arşivlenmesi gerekmektedir. Bu tür çalışmalar, ancak güçlü bir bilgisayar yardımıyla gerçekleştirilebilir.

Yapılan çalışmada her düğüme ilişkin veri bir sekizlide (byte) saklanmaktadır. Her sekizlide dört bit renk bilgisi için, dört bit de bordür ve motif bilgileri için kullanılmaktadır. Renk bilgisi için ayrılan dört bit ile bir desende 16 ayrı renk tanımlanabilmektedir. Diğer dört bit ile de bir düğümün komşu düğümlere olan bağıntısı yani yön bilgisi saklanmaktadır. Bir düğüme ilişkin yön bilgisi

sadece kendisinde değil, aynı zamanda komşu düğümlerde de saklanmaktadır, yani bir düğümün taşıdığı bilgi tek başına fazla bir anlam taşımamaktadır. Yön bilgisinin saklanması ile, bir bordür veya motif üzerindeki herhangi bir düğüm ile programa tanıtılabilmektedir. Böylece, bir kapalı şekil (örneğin bir çiçek motifi) ne kadar karmaşık şekillerin biraraya gelmesinden oluşursa oluşsun program (örneğin bordürün silinmesi veya renginin değiştirilmesi için) hiç bir düğümü kaçırmadan şeklin tamamı üzerinde dolaşabilmektedir. Ayrıca kapalı alanların renklendirilmesinde, rengin alan dışına taşması yön bilgisiyle önlenmektedir (4).

5.2 Halı Deseninin Bilgisayara Yüklenmesi

Desinatör tarafından serbest formda hazırlanmış desen paftası sayısallaştırıcı üzerine yerleştirilmekte ve sayısallaştırıcı tablet üzerinde tanımlanmış özel komutlar yardımıyla desen bilgisayara yüklenmektedir. Bu komutlar desen üzerinde belli bir alanın silinmesini, çerçevelerle belirlenen alanların veya motiflerin saklanmasını ve olduğu gibi ya da döndürülerek başka bir yere kopya edilmesini, bordürlerin ve bordürlerin sınırladığı bölgelerin renklendirilmesini sağlamaktır (4).

5.3 Döndürme

Desen üzerinde tanımlanan her şekil veya belli bir çerçeve ile sınırlanmış bölgede bulunan tüm şekiller, bu şekil veya bölge üzerinde tanımlanmış bir nokta (orijin) etrafında 90° , 180° ve 270° derece döndürülebilmekte, ayna görüntüsü alınabilmekte ve bir başka noktaya taşınabilmektedir (4).

5.4 Renklendirme

Program, kullanıcının belirlediği her renge karşı bir sayısal kod tanımlamaktadır. Böylece bir halı deseninin birbirinden farklı renk bileşimlerinde kolaylıkla izlenmesi sağlanmaktadır (4)

5.5 Bordür Renklendirme

Halılarda kapalı şekillerin genellikle, tek bir kapalı şekilden ziyade birçok kapalı şekillerin biraraya gelmesiyle oluştuğu görülmektedir (örneğin bir çiçek motifi). Veri yapısının özelliği nedeniyle karmaşık bir şeklin herhangi bir düğümünden başlanarak tüm düğümlere erişilebilmektedir. Böylece herhangi bir motife ait sakin-tiler veya bordürler değişik renklerle renklendirilebilmektedir. Bu işlemin gerçekleştirilebilmesi için her düğüme ilişkin yön bilgisi kullanılmaktadır. Sözkonusu düğüme geliş yön bilgisi ve düğüme ilişkin bilgidен yararlanılarak ilerlenmesi muhtemel yönler bulunmaktadır. Birden fazla yöne ilerlenmesi muhtemel olan durumlarda en öncelikli olan yön kontrol edilmekte, diğer yönler ise bir yığında saklanmaktadır. Algoritma, seçilen yönde bir bordür düğüme veya bordür olmayan bir düğüme rastlandığında durmakta, yığından bir değer okuyarak işlemlerine devam etmektedir. Yığın boşaldığında, ilgili tüm düğümlere erişildiği anlaşılakta ve işlem durdurulmaktadır (4).

5.6 Alan Renklendirme

Bir bordür veya motif ile sınırlandırılmış herhangi bir alanın içindeki tüm kapalı şekiller dışında kalan kısmın renklendirilmesi de geliştirilen bir algoritma ile sağlanmaktadır (4).

Kullanıcı, renklendirmek istediği alanı, içindeki herhangi bir nokta ile programa tanıtmakta, algoritma bu noktadan başlayarak düşey yönde (aşağı ve yukarı), seçilen renkte bir düğüme veya bir bordüre rastlayana kadar ilerlenmektedir. ilerleme yönüne bağlı olarak, bulunan bu doğrunun sağında (pozitif) ve/veya solunda (negatif) ilerlenmesi muhtemel olan yönlerdeki düşey aralıklar özel olarak tanımlanmış yığınlarda saklanmaktadır. Yığınlardan okunan her değer için yukarıdaki işlemler tekrarlanmakta ve yığınlar boşaldığında tüm düğümlere erişildiği anlaşılıp işlem durdurulmaktadır.

5.7 Komut Formatı

Kullanıcı, gerçekleşmesini istediği işlemleri, programa bir komut dizisiyle tanıtmaktadır. Her işlemin belirli bir komut formatı vardır (4).

Program komutları sayısallaştırıcı tablet ve sayısallaştırıcı üzerinde tanımlanan özel komut tablosu üzerinden okunarak girilmekte, sık yapılan işlemlere ilişkin komutlar ise kullanıcıya kolaylık sağlaması için sayısallaştırıcı tablet üzerinde tanımlanmaktadır.

6. HALININ DOKUNMADAN ÖNCE GERÇEK BOYUTUNUN HESAPLANMASI

6.1 Giriş

Günümüzde halıcılık sektörünün içinde bulunduğu en önemli problemlerden birisi de, halı dokunmadan önce ip kalınlıklarına göre gerçek boyutunun hesaplanabilmesidir. Özellikle el dokuması halılarda, insan faktörü ve ip kalınlığı halı boyutu üzerinde doğrudan etkili olmaktadır.

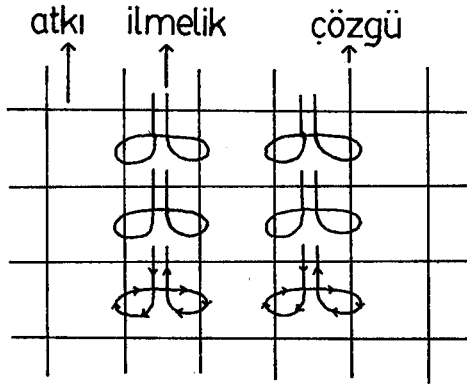
Halı dokumacılığında, belli yöre halılarında aynı ip kalınlıkları kullanılsada, üretilen iplerin kalınlıklarında belli bir standart sağlanamamaktadır. Ayrıca dokuma işlemi sırasında, bir sıra düğüm atıldıktan sonra bir sırada ara atkısı geçirilip vurularak düğümler sıkıştırılmaktadır. Sıkıştırma miktarıda tamamen dokuyucunun vuruş kuvvetiyle orantılıdır. Bu nedenle, halı boyutunun, dokuma işlemi gerçekleşmeden önce ip kalınlıklarına göre hesaplanmasında teorik ve pratik uygulama arasında sabit bir oran sağlanması son derece güçtür. Böylece, öncedenbelirlenen bir kalite ve ebatta halının dokunması çoğu kez mümkün olmamaktadır.

Bu tez de yapılan çalışmada, bir düğüm kalınlığını etkileyen faktörler gözönüne alınarak, teorik hesaplama ile pratik uygulaması arasında belli bir yakınlık sağlanabilmiştir. Hata oranı, halı eninde yaklaşık $\% -0.42$, boyunda ise $\% -0.24$ gibi çok küçük bir değerde olmaktadır.

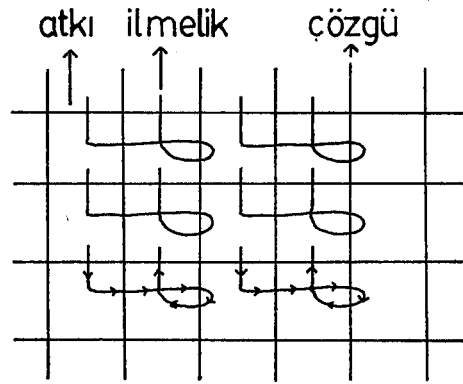
6.2 İp Kalınlıklarına Göre Halı Boyutunun Hesaplanması

Bir halının dokunması işleminde, kalınlıkları ve düğüm içindeki fonksiyonları farklı olan üç tür ip kullanılmaktadır. Bu ip kalınlıkları bir düğümün büyüklüğünü

doğrudan etkileyen faktörlerdir. Şekil 6.1'de, bir düğümde kullanılan iplerin konumları ve iki farklı tür düğüm atma tekniği gösterilmektedir.



Şekil 6.1.a Çift düğüm tekniği



Şekil 6.1.b Tek düğüm tekniği

Şekil 6.1 Düğüm atma teknikleri

Halıcılıkta kullanılan bazı terimlerin anlamları kısaca şöyledir:

Kalite: Bir halıda 10cm^2 'sindeki düğüm sayısıdır. Örneğin; 28×32 olarak belirlenen bir kaliteye göre, dokunacak halının 10cm^2 'sinde enine 28, boyuna 32 düğüm olmalıdır.

Çözü: Bir halıda boyuna çekilen ipliklerdir. İki çözü arasında bir düğüm atıldığından halının enini belirleyen en önemli faktörlerden birisidir. Genellikle ilmelik ipinden daha kalın ve 2 ya da 3 bükümlüdür.

Atkı: Bir halıda, bir sıra düğüm atıldıktan sonra enine, çözü iplerinin bir altından bir üstünden geçirilerek çekilen ipliklerdir. Böylece çözü iplerinin birbirinden ayrılması önlenmektedir. İsteğe göre iki düğüm sırası arasında iki ya da daha fazla atkı ipi çekilebilir. Daha sonra bu atkı iplikleri dokuyucu tarafından vurularak, atkı

ve ilmelik iplerinin iyice sıkıştırılıp aradaki boşlukların giderilmesi sağlanmaktadır. Genellikle ilmelik ipten daha kalın 2 ya da 3 bükümlüdür.

ilmelik: Halının düğümlerinin atıldığı ipliklerdir. Atkı ve çözgü ipliklerine oranla daha ince ve bükümlü olabilmektedir.

Bir halı dokunmaya başlamadan önce, halının ebatları, hangi kalitede olması istendiği ve kullanılacak ipler belirlenmektedir. Belli yörelerde dokunan halılar için atkı, çözgü ve ilmelik ipleri genellikle aynı olmaktadır. Fakat iplerin üretimi sırasındaki bazı etkenlerden dolayı ip kalınlıkları standardından daha ince ya da kalın olabilmektedir. İp kalınlıklarındaki bu değişmeler, bir düğüm kalınlığı üzerinde doğrudan etkili olduğundan, önceden belirlenen halı kalitesinin de değişmesine sebep olmaktadır. Dolayısıyla, halı verilen sipariş ebatından daha büyük ya da küçük olabilmektedir. Bu durum ise ancak halı dokunup bittikten sonra görülebilmektedir. Böyle bir sorunda, belirlenen kaliteyi sağlamak için ip kalınlıklarının ne olması gerektiğinin ya da belirlenen iplerle halı dokunduğunda kalitenin ne olacağının önceden belirlenmesi çok büyük bir problemdir. Özellikle el dokuması halılarda, atkı ipinin çekilmesinden sonraki sıkıştırma tamamen dokuyucunun kuvveti ile orantılıdır. Bu durum sipariş ebatına göre halı boyunda önemli değişikliklere sebep olmaktadır. Standart bir sıkıştırma ve ip kalınlığı olmaması sebebiyle, halı boyutunun dokumaya başlanmadan önce hesaplanması da mümkün değildir.

Bu tezde yapılan çalışmada, halı dokumacılığının içinde bulunduğu bu gibi sorunlara cevap olabilecek yaklaşımlar geliştirilmiştir. Verilen ip kalınlıklarına göre halının dokunmadan önce kalitesinin ne olacağı ya da verilen kalitenin sağlanabilmesi için ip kalınlıklarının ne olması gerektiği hesaplanabilmektedir. Ayrıca desen çizimi

sirasında belirlenen bir kalite ve ebatta halının dokunabilmesi için halının eninde ve boyunda kaç düğüm olması gerektiği de yazılım tarafından kullanıcıya verilmektedir.

Bir halının dokunmadan önce bazı parametrelerinin belirlenmesi işlemi sırasında, pratik uygulaması yapılmış olan halıların bir takım özelliklerinden faydalanılmıştır. Belli yörelerde dokunan halıların çözgü, atkı ve ilmelik iplerinden örnekler alınarak bunların kalınlıkları ölçülmüştür. Bu iplerle dokunan, kalitesi ve ebatları belli halılar üzerinde yapılan çalışmalarla ip kalınlıkları ile kalite arasında bir bağıntı bulunmuştur.

Bir halının boyutu, enine ve boyuna düğüm sayısı ile düğüm kalınlıklarının çarpımı şeklinde belirlenmektedir. Kalitesi ve sipariş ebatı belli olan bir halıda enine ve boyuna olması gereken düğüm sayısı basit bir orantı ile hesaplanabilmektedir. Şekil 6.1' de de görüldüğü gibi enine bir düğüm kalınlığını etkileyen çözgü ve ilmelik, boyuna bir düğüm kalınlığını etkileyen de atkı ve ilmelik iplerinin kalınlığıdır. Daha önce de belirtildiği gibi bir halıdaki kalite, halının 10 cm^2 'sindeki düğüm sayısını göstermektedir. Bir düğüm iki çözgü ipi arasından atıldığı için, 10 cm. ' deki çözgü sayısı düğüm sayısının iki katı kadardır. Böylece 10 cm. ' deki çözgü sayısı halı enini belirleyen en önemli faktörlerden birisidir. Şekil 6.1.a' daki düğüm atma şekli göz önüne alındığında, teorik olarak enine bir düğüm kalınlığının;

$$\text{Enine bir düğüm kalınlığı} = 2 \times \text{çözgü kal.} + 4 \times \text{ilmelik kal.} \quad (6.1)$$

Boyuna bir düğüm kalınlığının ise;

$$\text{Boyuna bir düğüm kalınlığı} = 2 \times \text{atkı kal.} + 1 \times \text{ilmelik kal.} \quad (6.2)$$

olması gerektiği görülmektedir. Bu şekilde, belirlenen ip kalınlıkları ile bir düğümün enine ve boyuna kalınlığı he -

saplandığında, pratik uygulamadaki düğüm kalınlığı ile arasında çok büyük farklar bulunmaktadır. Bunun üzerine, halının dokunması sırasında bir düğüm içindeki iplerin konumları gözlenerek belli yaklaşımlar yapılmıştır.

Çift düğüm tekniğinde, iki çözü arasında birleşerek çıkan ilmelik iplikleri yanyana değil de üstüste durmaktadır. Ayrıca çözülere dolanan ilmelik iplikleri ise düğüm atma sırasında iplerin çekilmesinden dolayı enine düğüm kalınlığında etkili olmamaktadır. Bu yaklaşıma göre enine bir düğüm kalınlığının;

Enine bir düğüm kalınlığı = $2 \times \text{çözü kal.} + 1 \times \text{ilmelik kal.}$ (6.3)

olması gerektiği bulunmuştur. Örnek olarak alınan halılarda 10 cm.'deki enine düğüm sayısı belli olduğundan orantı yoluyla bir düğüm kalınlığı hesaplanmış ve bu değer teorik yaklaşımla bulunan büyüklük ile karşılaştırılmıştır. Sonuçun birbirine oldukça yakın ve hata oranının -0.42% gibi çok küçük bir değerde olması yaklaşımın doğru olduğunu kanıtlamaktadır.

Boyuna bir düğüm kalınlığında ise sıkıştırma kuvveti doğrudan etkili olduğundan, örnek halılara göre bir yaklaşım yapmak oldukça güç olmaktadır. Atkı kalınlığı ve 10 cm.'deki boyuna düğüm sayısı belli olan halılarda orantı yoluyla boyuna bir düğüm kalınlığı hesaplandığında, bu değer iki atkı kalınlığı kadar olduğu görülmüştür. Buna göre genellikle bir halıda ilmelik kalınlığını yok edecek kadar boyuna bir sıkıştırma olmaktadır. Fakat bu sıkıştırma miktarı standart olmayıp tamamen dokuyucunun kuvveti ile orantılı olmaktadır.

Şekil 6.1.b' deki tek düğüm tekniğinde ise, ilmelik kalınlığının enine bir düğüm kalınlığı üzerindeki etkisi

yok denecek kadar azdır. Düğüm atma şeklinden ve düğüm atma sırasında iplerin çekilmesinden dolayı enine bir düğüm kalınlığı;

Enine bir düğüm kalınlığı = $2 \times \text{çözgü kal.}$ (6.4)

yaklaşık iki çözgü kalınlığı kadar olmaktadır.

Boyuna bir düğüm kalınlığı ise, çift düğüm atma tekniğindeki hesaplama şekli ile aynıdır

Pratikteki uygulamalara göre getirilen bu yaklaşımlarla hesaplanan ip kalınlıkları ya da verilen ip kalınlıklarına göre olabilecek kalite teoride olması gerekendir. Fakat özellikle el halıcılığında insan faktörünün olması ve ip kalınlıklarında belli bir standardın bulunmaması nedeni ile teoride olması gereken ile pratik uygulama arasında farklar doğabilmektedir.

7. YAZILIM

7.1 Giriş

Bu bölümde, daha önceki bölümlerde, sayısal görüntü işleme üzerine geliştirilmiş olan birtakım tekniklerin ışığı altında sayısal bir resmin yeniden şekillendirilmesi ve değerlendirilmesi üzerine geliştirilen sistem anlatılmaktadır. Üzerinde işleme yapılacak olan görüntünün dijitalizer ya da dijital bir kameradan alınması imkanı gerçekleştirilemediğinden, programa resim çizmek için gerekli fonksiyonları sağlayan bir bölüm eklenmiştir.

Günümüzde halıcılık sektörünün içinde bulunduğu bazı problemlere belli bir ölçüde çözüm sağlanması amacıyla, üzerinde işleme yapılacak olan resimler halı desenlerinden seçilmiştir. Yapılan çalışma, halı desenlerinin geliştirilmesi, üzerinde değişiklik yapılması ve kolaylıkla çoğaltılabilmesi amacı doğrultusunda geliştirilmiştir. Ülkemizde, halı deseni üzerinde yapılan çalışmalar, çok emek ve yüksek maliyet gerektirdiğinden yetersiz kalmaktadır. Bir halı deseninin oluşturulması şu safhaları kapsamaktadır.

1. Desinatör hayalinde canlandırdığı deseni serbest formda kağıda aktarır.
2. Bordür ve motifler kareli kağıda geçirilir.
3. Desen renklendirilir.
4. Desenin tamamı incelenir, desen veya renklendirme konusundaki eleştirilerin ışığı altında gerekli düzenlemeler ve değişiklikler yapılır.
5. Desen son halini aldığı anda temiz bir kopya çıkarılır.

5. Çoğaltma ya elle ya da offset tekniğiyle gerçekleştirilir.

Görüldüğü gibi yukarıda belirtilen işlemlerin hemen hemen hepsi elle yapılmaktadır. Bu işlemler sırasında da birtakım problemlerle karşılaşılmaktadır. Bu problemler şunlardır:

(1) Genellikle desinatörler, kareleyicilerin kendine özgü yorumları nedeniyle karelenmiş desenlerin istedikleri gibi olmamasından yakınmaktadırlar. Bu nedenle, bazen desinatörler kareleme işlemlerini de kendileri yapmakta ve bu son derece zor ve emek isteyen işe uzun bir süre ayırmaktadırlar.

(2) Kağıt üzerinde elle renklendirme olanakları son derece sınırlıdır. Desinatörler kendi tecrübelerinden yararlanarak uyumlu renkleri biraraya getirmekte ve deseni renklendirmektedirler. Kağıt üzerinde silme suretiyle renk değişikliği kağıdın yıpranması nedeniyle ancak bir iki kez yapılabilmektedir.

(3) Desenin offset baskıyla çoğaltılması pahalı olmakta, elle yapıldığında ise göz yanılması gibi nedenlerle desenlerdeki yanlışlar da artmaktadır. Ayrıca elle çoğaltmanın güçlüğü nedeniyle, simetrik desenler için yanlış simetriyi oluşturan en küçük bölümler çizilmekte, böylece dokuyucunun simetriyi zihninde canlandırmasından kaynaklanan dokuma yanlışları oluşmaktadır.

(4) Belirlenen ip kalınlığı ve kalitede dokuma işlemi gerçekleştirildiğinde halının gerçek boyutunun ne olacağına önceden hesaplanması çok güç olduğundan, genellikle sipariş ebatını tutturmak büyük bir problem olmaktadır.

Halı deseni üzerindeki çalışmalarda emek, zaman ve maliyeti azaltmak, ayrıca bazı problemlere cevap vermek için geliştirilen program QBASIC programlama dili kullanılarak yazılmıştır. Böylece desen üzerinde uygulanan işlemlerde yeterli bir hıza ulaşılmıştır. Ayrıca kullanıcıya kolaylık sağlanması açısından büyük ölçüde kolay ve anlaşılır biçimde bir yazılım gerçekleştirilmiştir.

Aşağıda programın oluşturulmasında izlenen yol ve alt programlar akış şemalarıyla birlikte anlatılmıştır.

7.2 Ana Program

Şekil 7.1'de gösterildiği gibi ana program farklı işlevleri seçmek için olanak sağlayan 7 ana bölümden oluşmaktadır.

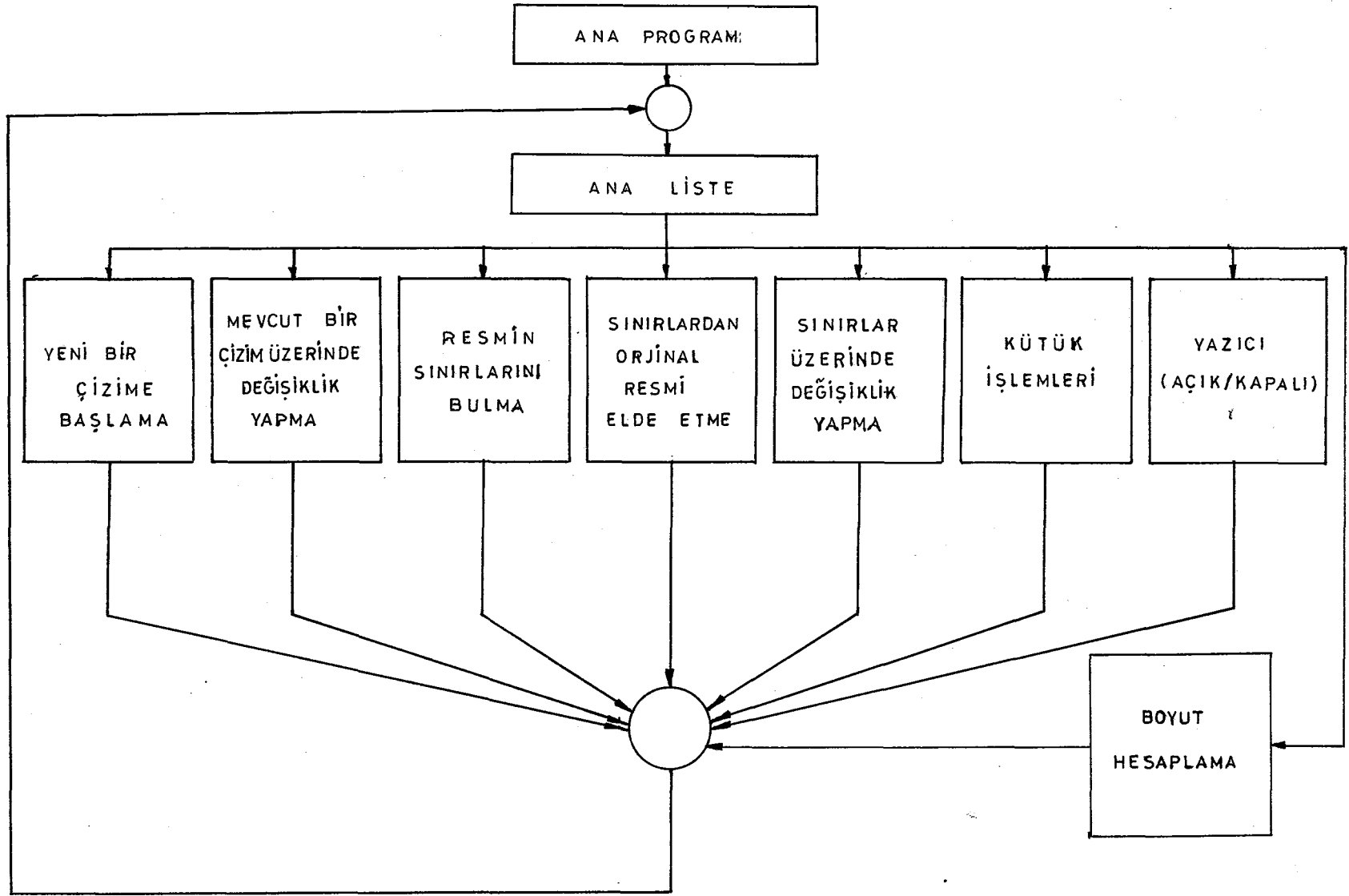
YENİ BİR ÇİZİME BAŞLAMA : Yeni çizim yapma (desen oluşturma) için gerekli olan ekran düzenlemesine geçilir.

MEVCUT BİR ÇİZİM ÜZERİNDE DEĞİŞİKLİK YAPMA : Ekran düzenlemesi, yeni bir çizim yapma modundakiyle aynı olmakla birlikte işlev olarak farklıdır. Belli bir kütük isminde depolanmış olan desen üzerindeki birtakım çizim hataları düzeltilebilir ya da isteğe göre bazı değişiklikler yapılabilir.

RESMİN SINIRLARINI BULMA : Çizilmiş renkli bir desende, deseni oluşturan motiflerin sadece sınırları, orijinal renkleriyle belirlenir. Geliştirilen bir formülasyonla sınırlar depolanır.

SINIRLARDAN ORJİNAL RESMİ ELDE ETME : Bir önceki modda, deseni oluşturan motiflerin sınırları belirlenip belli bir kütük isminde depolandıktan sonra, bu modda sınır renkleri referans alınarak orijinal renkli desen tekrar elde edilir.

ŞEKİL 7.1 ANA PROGRAM AKIŞ ŞEMASI



SINIRLAR ÜZERİNDE DEĞİŞİKLİK YAPMA : Renkli deseni oluşturan motifler üzerinde birtakım değişiklikler (motifin başlangıç koordinatlarını ve rengini değiştirme, büyültme, küçültme, döndürme), motifi ifade eden formülasyonda bazı parametreleri değiştirmekle gerçekleştirilir.

KUTUK İŞLEMLERİ : Kütüklerle ilgili birtakım yardımcı işlemler geliştirilmiştir.

YAZICI (AÇIK/KAPALI) : İstenen herhangi bir deseni yazıcıdan elde etmek için geliştirilen bir bölümdür.

BOYUT HESAPLAMA : Verilen kalite ve ebatta bir halı dokunması için ip kalınlığının ne olması gerektiği ya da verilen ip kalınlıklarına göre kalitenin ne olacağı halı dokunmadan önce hesaplanabilir.

7.2.1 Yeni bir çizime başlama

Bu program, esasen var olan sayısal bir resim üzerinde inceleme yapma, resmin yeniden şekillendirilmesi ve değerlendirilmesi amacı doğrultusunda geliştirilmiştir. Fakat sayısal bir resmin bir dijitalizer ya da dijital kamera kullanılarak elde edilmesi imkanı sağlanamadığından, desen oluşturma işlemi yazılımın bu bölümünde gerçekleştirilmektedir. Desen çizimi sırasında kullanıcıya kolaylık sağlaması açısından, fonksiyon tuşları bazı yardımcı komutlara (çizgi çizme, çizgi silme, çizme ya da silmeyi durdurma, renk seçme, ekran silme, pencereleme (silme, kopyalama, taşıma), çıkış) programlanır. 128x128 noktadan oluşan bir çizim alanı içinde hareketli bir kursör ve belirtilen yardımcı komutlar sayesinde oluşturulan desen, ana programa dönme tuşuna (ÇIKIŞ) basıldığında isteğe göre otomatik olarak depolanabilir. Depolama işlemi sırasında kütük ismi olarak, bu mod seçildiğinde girilen kütük ismi alınır.

7.2.2 Mevcut bir çizim üzerinde değişiklik yapma

Bu bölümde, daha önce çizilip belli bir kütük isminde depolanmış olan desen tekrar ekrana çizilir. Birinci bölümde desen çizilip depolandıktan sonra göze çarpan birtakım hataların düzeltilmesi ya da desen üzerinde bazı değişikliklerin yapılması bu bölümün fonksiyonları arasındadır. Ekran düzenlemesi ve fonksiyon tuşlarının işlevleri, yeni bir çizim yapma modundakiyle aynıdır. Ana programa dönme tuşuna basıldığında, resim aynı kütük ismi altında, istenirse tekrar depolanabilir.

7.2.3 Resmin sınırlarını bulma

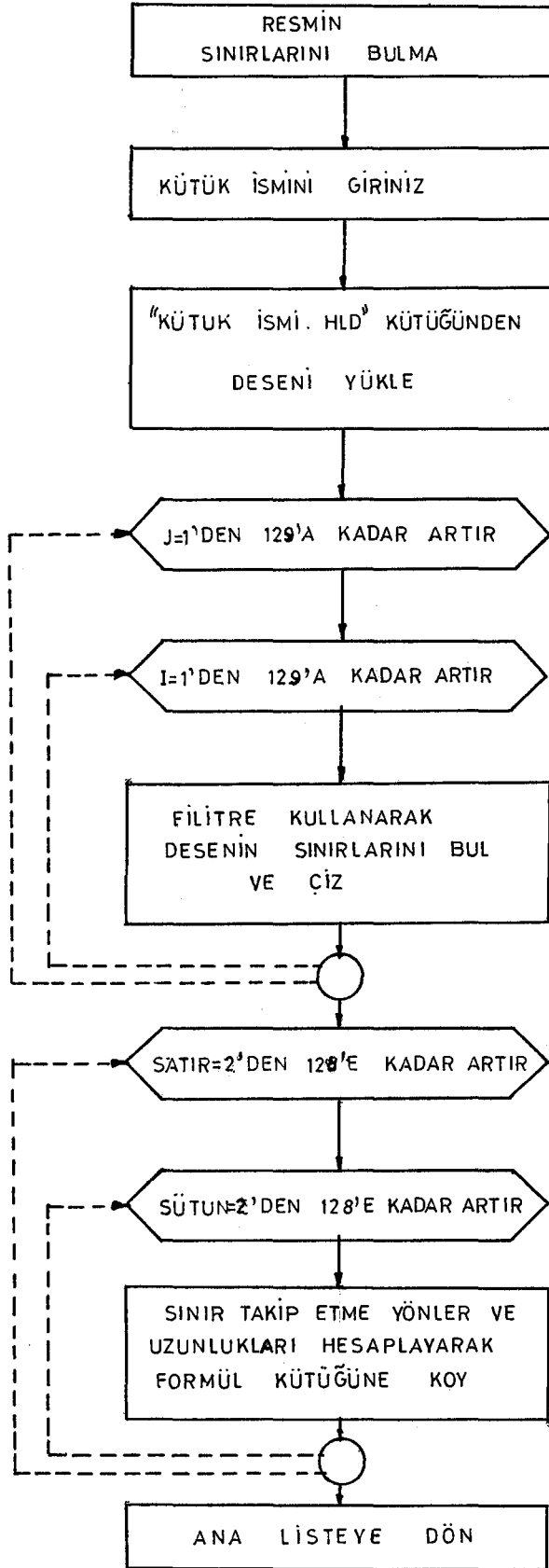
Program bu modda, renkli olarak çizilmiş olan bir desende, deseni oluşturan motiflerin sınırlarını orijinal renkleriyle belirleyerek, sınır takip etme fonksiyonuyla, her sınırı bir formülasyonla ifade ederek depolamaktadır.

Bir deseni oluşturan nesnelerin sınırlarını bulmak amacıyla geliştirilen yöntemler daha önceki bölümlerde açıklanmıştır. Bu programda, özellikle renkli bir desende, deseni oluşturan motiflerin sınırlarının belirlenip içlerinin boşaltılması bir filtre yardımıyla gerçekleştirilmektedir. 3x3'lük bir matris tüm desen üzerinde dolaştırılıp sınır olarak belirlenen noktalar orijinal renk değerini almakta, diğer noktalar ise siyah olarak kalmaktadır.

	E	
F	*	H
	G	

*= sınır olup olmadığı incelenen pikselin konumunu gösterir.

Şekil 7.3. Motif sınırlarının bulunmasında kullanılan filtre



ŞEKİL 7.2 DESENİ OLUŞTURAN MOTİFLERİN SINIRLARINI BULMA
ALT PROGRAMI

E, F, G, H incelenen noktanın sađ-sol-alt-üst komşuluđundaki noktaların renk kodlarını belirtmektedir.

$$B = \text{ABS} (F-H)$$

$$D = \text{ABS} (E-G)$$

$$A = \text{ABS} (F-E)$$

$$\text{Yeni nokta deęeri} = A+B+D$$

Program bu modda çalışmaya başladığında ekran düzenlemesi, ana listeden ilk iki mod seçildiđi zamankinden farklıdır. Çizim alanı olarak 128x128 noktadan oluşan iki bölge ayrılır. Ekranın ilk kısmında, belirlenen kütük isminde, sınırlarının bulunması istenen desen çizilmektedir. Deseni kütükten çekip çizme işlemi tamamlandıktan sonra, desen en sol-üst köşeden itibaren nokta, nokta taranmaya başlar. Ele alınan her noktanın sınır olup olmadığını belirlemek amacıyla, incelenen noktanın renk deęeri ile komşuluđundaki noktaların renk deęerleri arasında yukarıda açıklanan filtre işlemi uygulanır. Bu işleme göre yeni nokta deęeri olarak hesaplanan renk deęerinin sıfır olması, incelenen noktanın sınır olmadığını belirtir. Böylece ikinci 128x128'lik alan içinde, bu noktanın konumuna karşılık gelen nokta siyah olarak kalır. Yeni nokta deęeri olarak hesaplanan renk deęerinin sıfırdan farklı bir deęerde olması ise incelenen noktanın sınır olduđu anlamını ifade eder. Yani sözkonusu noktanın, ele alınan komşuluklarında farklı renk grupları var demektir. Sonuçta, sınır olarak belirlenen nokta, ikinci çizim alanına karşı düşen konumunda kendi rengini koruyarak yer alır. Bu işlem birinci çizim alanındaki tüm desen nokta, nokta taranmaya kadar devam eder.

Bir deseni oluşturan motiflerin, orijinal renkleriyle, kapalı alanlar şeklinde sadece sınırları belirlendikten sonra, desenin bu şekliyle kütükde saklanması belli bir formülasyonla sağlanmaktadır. Böylece her noktaya ait veri için bir sekizli (byte) ayrılmaktansa, sadece sınırdaki olan noktalara ait veriler saklanmaktadır. Bu sayede desenin depolanması için gerekli olan hafıza kapasitesinden de büyük ölçüde tasarruf edilmektedir.

Sınır bilgilerinin saklanması için bir yöntem geliştirilmiştir. Sınırları belirlenen motiflerin çoğu kapalı alanlar şeklinde olmasına karşın bazıları da kapalı değildir. Oluşturulan kütük yapısı gereğince, her kapalı (ya da kapalı olmayan) alan için byte'lar ayrılır. Bunun ilk iki byte, sınırı takip edilecek olan motifin, takibe başlanacağı başlangıç noktasının koordinatları, üçüncü byte sınırı takip edilen motifin renk kodu, dördüncü byte ölçek faktörü, beşinci byte döndürme açısı, diğer byte'lar ise yönler ve uzunluk bilgileri için ayrılmıştır. Sınırların bir formül kütüğünde saklanması sırasında dördüncü byte ölçek faktörünü "1" yapan "4" değerine, beşinci byte ise, herhangi bir döndürme olmadığı anlamında "0" 'a otomatik olarak ayarlanır.

Deseni oluşturan tüm motiflerin sınırları belirlendikten sonra, sınır takip ederek sadece sınırdaki noktaları ifade eden bilgilerin kütükde saklanması işleminde, ilk önce sıfırdan farklı bir renk değerine ulaşmaya kadar desen taranır. Siyah rengi belirten "0" 'dan farklı bir renk koduyla karşılaşıldığında, bu noktanın koordinatları başlangıç değerleri olarak saklanır. Belirlenen rengin takibi için, bu noktanın sekiz komşuluğu taranır. Bu tarama işlemine, incelenen noktanın ilk önce sağ (right) komşuluğuna bakılarak başlanır. Eğer burada takip edilen sınır rengi bulunamazsa, izlenecek sıra, sağ-alt (right-down), alt (down), sol-alt (left-down), sol (left), sol-üst (left-up), üst (up) ve sağ-üst (right-up) komşuluk-

lar şeklindedir. Bu şekilde devam eden tarama işlemi, motif kapalı bir alan ise sınır takibine başlanan noktaya ulaşınca kadar, motif kapalı bir alan değil ise, komşuluklardan herhangi birinde takip edilen renk bulunamayınca kadar sürer.

Sınır takip etme işlemi sırasında, deseni oluşturan motiflerin sınırlarının hangi yönde kaç nokta gidilerek bulunduğu hesaplanır ve kütük içinde ayrılan bölgeye yazılır. Herhangi bir motife ait sınır bilgilerinin sona erdiği ise, sözkonusu motif için kütük içinde ayrılan yerin sonuna sıfır konularak belirlenir. Bir motifin sınırlarını tanımlayan bilgiler (başlangıç koordinatları, renk kodu, ölçek faktörü, döndürme açısı, yönler ve uzunluklar) kütük içine depolanır.

1.byte	2.byte	3.byte	4.byte	5.byte
X BAŞ.	Y BAŞ.	renk	ölç.	açı	yön ve uzunluk

Şekil 7.4. Bir motife ait bilgilerin kütük içine yerleştirilmesi.

Sınırları ifade eden yönleri ve bu yönlerdeki uzunlukları kütük içine depolarken, ilk üç bit yönleri (program içinde DDR değişkeni ile gösterilir) belirtmek, diğer beş bit ise belirtilen yöndeki uzunluğu belirtmek için kullanılır. Bu bilgiler için bir sekizli yetmezse sekizli sayısı arttırılmaktadır.

0 0 0 → sağ (right)

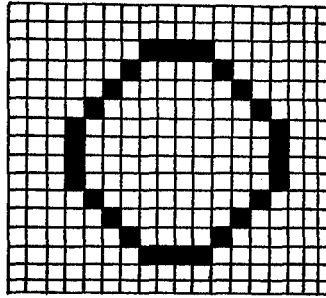
0 0 1 → alt (down)

0 1 0 → sol (left)

0 1 1 → yukarı (up)
 1 0 0 → sağ-alt (right-down)
 1 0 1 → sol-alt (left-down)
 1 1 0 → sol-üst (left-up)
 1 1 1 → sağ-üst (right-up)

Bir deseni oluşturan tüm motiflerin sınırları yukarıda açıklanan kütük yapısı içine (formül file), her motif formülize edilerek depolanır.

Örnek : Şekil 7.5'deki gibi sınırları belirlenmiş olan bir motifi ifade eden bilgilerin kütüğe yerleştirilmesi şu şekildedir.



Şekil 7.5. Sınırları belirlenmiş bir motif örneği.

X-BASLANGIÇ=25 Y-BASLANGIÇ=50 RENK=4 (KIRMIZI)

ÖLÇEK=4 (ölçek faktörünü "1" yapan değer)

AÇI=0 (Herhangi bir döndürme yok)

(X-BASLANGIÇ, Y-BASLANGIÇ) başlangıç noktasından başlayarak belirli yönler ve bu yönlerdeki uzunluklar şu formülle hesaplanır:

$$FFF(FC) = (DDR-1) * 32 + CNT \quad (7.1)$$

FC, bir motife ait bilgilerin yerleştirdiği byte'ların sayısını gösterdiği için yukarıdaki formülasyon FC'nin 6 ve daha büyük değerleri için geçerlidir. DDR, sekiz yönden herhangi birisini, CNT, ise aynı yönde gidilen nokta sayısını göstermektedir.

$$FFF(1) = 25 \quad FFF(2) = 50 \quad FFF(3) = 4 \quad FFF(4) = 4 \\ FFF(5) = 0$$

Başlangıç noktasından taramaya başlanırsa:

$$\begin{array}{llll} DDR = 1 \text{ (right)} & CNT = 3 & FFF(6) = (1-1) * 32 + 3 = 3 \\ DDR = 6 \text{ (left-down)} & CNT = 4 & FFF(7) = (6-1) * 32 + 4 = 164 \\ DDR = 2 \text{ (down)} & CNT = 3 & FFF(8) = (2-1) * 32 + 3 = 35 \\ DDR = 5 \text{ (right-down)} & CNT = 4 & FFF(9) = (5-1) * 32 + 4 = 132 \\ DDR = 3 \text{ (left)} & CNT = 3 & FFF(10) = (3-1) * 32 + 3 = 67 \\ DDR = 8 \text{ (right-up)} & CNT = 4 & FFF(11) = (8-1) * 32 + 4 = 228 \\ DDR = 4 \text{ (up)} & CNT = 3 & FFF(12) = (4-1) * 32 + 3 = 99 \\ DDR = 7 \text{ (left-up)} & CNT = 4 & FFF(13) = (7-1) * 32 + 4 = 196 \end{array}$$

Yönlere bağlı olarak uzunluklar hesaplandıktan sonra kütük içine depolanır (Şekil 7.6).

1.byte	2.byte	3.byte
25	50	4	4	0	3	164	35	132	67	228	99	196

Şekil 7.6 Örnek motife ait bilgilerin kütük içine yerleşimi.

7.2.4 Sınırlardan orijinal resmi elde etme

Programın bu bölümünde, bir önceki modda sınırları bulunarak (7.1)'deki bağıntı ile formülize edilmiş ve

belli bir kütük ismi ile formül kütüğüne depolanmış olan desen tekrar çizdirilerek, orijinal renkli haline dönüştürülebilmektedir.

(7.1)'deki formülasyonla kütüğe atılan, motiflerin sınırlarını belirleyen değerlerin, kütükten çekilerek tüm desenin sadece sınırlarla çizilmesinde, basic programlama dilinde kullanılan "DRAW" komutundan yararlanılmaktadır.

"DRAW" komutunun kullanımı:

"DRAW" komutu, bir grafik tanımlama dili ile çizim yapmak için kullanılmaktadır. "DRAW" komutu yanında bir takım açıklamalarla çizim gerçekleştirilmektedir. Komutun yürütülmesi sırasında, bazı değerler ve harflere göre nesneler çizilmektedir.

Aşağıda verilen hareket komutları, girilen son nokta referans alınarak belirli yönlerde çizim yapmayı sağlamaktadır.

U n	Yukarı hareket
D n	Aşağı hareket
L n	Sola hareket
R n	Sağa hareket
E n	Ust-sağ köşegende hareket
F n	Alt-sağ köşegende hareket
G n	Alt-sol köşegende hareket
H n	Ust-sol köşegende hareket

n , önündeki komutun gösterdiği yönde hareket edilecek uzunluğu göstermektedir. n , ile belirlenen sayı, ölçek faktörüyle belirlenen değer kadar çarpılmaktadır.

Programda, çizim yapmak için "DRAW" komutu içinde kullanılan diğer yardımcı komutlarda şunlardır:

TA n : Bu komutta, n döndürme açısını göstermektedir. n 'nin sınırları -360° 'den $+360^\circ$ 'ye kadardır. n pozitif ise dönüş yönü saatin tersi yönü, aksi takdirde saatin dönüş yönündedir.

C n : Ayarlanmak istenen renk değerini belirleyen komuttur. n , seçilmek istenen rengin renk kodunu göstermektedir.

S n : Çizim için bir ölçek faktörü belirleyen komuttur. n 'nin sınırı "1" ile "255" arasındadır. n ile verilen sayı "4"'e bölünerek ölçek faktörü belirlenmektedir. Örneğin ; " $n=1$ " ise, o çizimdeki ölçek " $1/4$ "'dür. Böylece U, D, L, R, E, F, G, H komutlarıyla verilmiş olan uzunluklar ölçek faktörüne bölünerek, çizilecek uzunluklar hesaplanmaktadır. Herhangi bir ölçek faktörü belirlenmediğinde, otomatik olarak alınan değer "4"'dür. Böylece ölçek "1" olmaktadır.

Sınırları belirlenerek (7.1)'deki formülasyonla kütüğe depolanan nesnelere tanımlayan yönler ve uzunluk değerlerinin kütükten çekilerek, gösterdikleri yön koduna göre karşılık gelen harf ve bu yöndeki uzunluğun hesaplanması yine bir formülasyonla gerçekleştirilmektedir.

Orijinal desenin sadece sınırlarını belirleme modunda, motiflerin sınırlarını ifade eden yönler ve uzunluklar karakterler olarak kütükte saklanmaktaydı. Bu bilgilerin

kütükten çekilerek sınır bilgisini içeren yönlerin ve tesbit edilen yöndeki uzunlukların bulunması aşağıdaki formlerle gerçekleştirilmektedir.

$$D = \text{INT} (F(I)/32) \quad (7.2)$$

$$UZ = \text{INT} ((F(I)/32-D)*32+0.5) \quad (7.3)$$

D : Yön kodu (direction)

UZ : Uzunluk (length)

F(I) : Kütükten çekilen yön koduyla birlikte uzunluğu içeren değer.

Hesaplanan bu değerler "DRAW" komutu içinde kullanılacak şekilde yerleştirilmektedir. Böylece sınır takip etme metoduyla kapalı ya da açık alanlar şeklinde kütüğe depolanmış olan motiflerin sınırları hızlı bir şekilde çizilmektedir. Ayrıca istenildiği takdirde bu sınırların içleri belirtilen renk değerleri ile taranarak orijinal renkli desen tekrar elde edilmektedir.

7.2.5 Sınırlar üzerinde değişiklik yapma

Programın bu bölümünde, sınırları belirlenerek, belirli bir sıra ile kütüğe depolanmış olan motifler tek tek tekrar çizdirilerek, motifler üzerinde birtakım değişiklikler gerçekleştirilmektedir. Bu işlemler motifin orijinal desen üzerindeki başlangıç koordinatlarını, rengini, ölçek faktörünü, döndürme açısını değiştirmektir.

Program bu modda çalışmaya başladığında, ilk aşama olarak, her motifin sınırlarını belirleyen değerler kütükten çekilirken aynı zamanda bir vektör içine de atılmaktadır.

Ekran düzenlemesi 128x128 noktadan oluşan iki çizim

alanından oluşmaktadır. İlk kısımda, girilen kütük isminde formül kütüğünde depolanmış olan desen çizilmektedir. İkinci 128x128 noktalık alanda ise vektöre atılmış olan motifler klavye'den kontrol edilerek, kütüğe atılma sırasına göre tek tek çizdirilmektedir. Üzerinde değişiklik yapılacak olan motif bu şekilde seçildikten sonra fonksiyon tuşları bazı yardımcı komutlara programlanarak, motifi tanımlayan birtakım parametreler değiştirilebilmektedir.

Seçilen motif üzerinde istenen değişiklikler (başlangıç koordinatlarını, rengini, ölçek faktörünü, döndürme açısını değiştirmek) yapıldıktan sonra, orijinal desen üzerinde eski motif silinerek, yeni şekliyle yer almaktadır. Böylece yapılan değişikliklerden sonra, düzeltilmiş motifin desenin bütünü içindeki konumu görülebilmektedir. Sonuçta bu değişikliklerle birlikte desen bir formül kütüğünde depolanmazsa yapılan değişikliklerin hiç birisi gözönüne alınmamaktadır.

Bir deseni oluşturan tüm motifler, belli yönlerde belli uzunluklar şeklinde ifade edilip, "DRAW" komutunun avantajlarından yararlanılarak çizimin gerçekleştirilmesi, desen üzerindeki işlemlerde büyük kolaylıklar sağlamaktadır. Bu komut içindeki, başlangıç koordinatları, renk, ölçek faktörü ya da döndürme açısından herhangi birinde değişiklik yapılması, bir "DRAW" komutu içindeki tüm bilgilere de yansdığından işlemlerin yürütülmesinde büyük bir hız sağlanmaktadır.

7.2.6 Boyut hesaplama

Program bu modda, özellikle el dokuması halıcılığının çok önemli bir problemine belli ölçüde bir çözüm getirmektedir. Bu problem tezin altıncı bölümünde geniş bir şekilde anlatılan, bir halının dokunmadan önce gerçek boyutunun hesaplanabilmesidir. Dokuma işlemine başlamadan halının hangi ebatlarda olması istendiği, kalitesi ve kullanılacak ipler belirlenmektedir. Fakat seçilen ip kalınlıkları, dokuma işlemi gerçekleştirildiğinde istenen kalite ve ebatı sağlamak için gerekenden ince veya kalın gelebilmektedir. Bu sorunun dokuma işleminden önce çözülmesi ise çok güçtür. Çünkü ip kalınlıkları ve atkı ipinden sonraki sıkıştırmalar çoğu kez standart olmamaktadır.

Yazılım içinde belli değerlerin hesaplanmasında pratik uygulaması yapılmış olan halılardan alınan ölçümlerden yararlanılmıştır. Kalitesi, sipariş ebatı ve çözgü kalınlığı belli olan bir halıda bu verileri sağlayacak ip ve atkı kalınlığının ne olması gerektiğinin hesabında izlenen yöntem şudur:

Öncelikle tüm özellikleri belli olan halılar üzerinde yapılan uzun bir çalışma sonucunda, bir düğüm kalınlığında çözgü, atkı ve ilmelik iplerinin etkisi bulunmuştur. Kalitesi (halının 10cm^2 'sindeki enine ve boyuna düğüm sayısı) bilinen bir halıda, basit bir orantı ile enine ve boyuna bir düğüm kalınlığı hesaplanmıştır. Ayrıca halının dokunduğu ipler elde edilerek bunların kalınlıklarında ölçülmüştür. Halılarda iki çözgü arasından bir düğüm atıldığı için, halının 10cm 'sinde düğüm sayısının iki katı kadar çözgü ipi vardır. Çözgü ipleri bir düğüm kalınlığına doğrudan etkili olmaktadır. Bu nedenle, 10cm 'deki çözgü sayısı ile çözgü ipi kalınlığının çarpımı 10cm 'den çıkarılarak, 10cm 'deki ilmelik iplerinin toplam kalınlığı bulunmaktadır. Bu büyüklük düğüm sayısına bölünerek iki çözgü ipi arasındaki ilmelik kalınlığı hesaplanmaktadır.

Çift düğüm tekniğinde, Şekil 6.1.a'da görüldüğü gibi iki çözümlü ipi arasından iki ilmelik ipi ucu çıkmakta ise de, hesaplama sonucunda bunun tek ilmelik ipi kalınlığı kadar olduğu görülmektedir.

Tek düğüm tekniğinde ise iki çözümlü ipi arasından tek bir ilmelik ipi çıkmaktadır. Fakat düğüm sırasında ilmelik ipinin çekilmesinden ve çözümlü ipi ilmelik ipinden kalın olduğundan, 10cm'de ilmelik ipi kalınlığının etkisi yok denecek kadar azdır. Bu nedenle halının kalitesinde etkili olan doğrudan çözümlü ve atkı ipinin kalınlığıdır.

ip kalınlıklarının bir halının boyu üzerindeki etkisini bulmak için, Şekil 6.1'deki düğüm atma şekilleri incelendiğinde sadece atkı ve ilmelik ip kalınlıklarına bağlı olduğu görülmektedir. Aynı şekilde kalitesi ve ip kalınlıkları belli olan bir halıda, kalite faktörünü kullanarak halının boyuna 10cm'sindeki bir düğüm kalınlığı hesaplanmaktadır. Diğer taraftan halının dokunması sırasında kullanılan atkı ipinin kalınlığı ölçülüp, hesaplanan bir düğüm kalınlığı ile karşılaştırıldığında bunun iki düğüm sırası arasına geçirilen atkı ipinin sayısı ve atkı ipi kalınlığı ile ilgili olduğu görülmektedir. Örneğin; iki sıra ara atkısı geçirildiğinde boyuna bir düğüm kalınlığı iki atkı ipi kalınlığı kadardır. Bunun anlamı ilmelik ipi kalınlığının halının boyu üzerinde bir etkisinin olmadığıdır. Her ara atkısının çekilmesinden sonra dokuyucu tarafından vurularak düğümler sıkıştırılmaktadır. Bu sıkıştırma miktarıda genellikle bir ilmelik kalınlığını yok edecek kadar olmaktadır. Bu yaklaşım tek ve çift düğüm tekniklerinin her ikisi içinde geçerlidir. Fakat bu sıkıştırma tamamen dokuyucunun vuruş kuvveti ile ilgili olduğundan sonuç her zaman hesaplanan değer gibi olmayabilir.

Kalite faktörü ve sipariş ebatı bilinen bir halıda, desen çizimi sırasında halının eninde ve boyunda kaç düğüm olması gerektiğinde orantı yolu ile hesaplanıp yazılım

içinde kullanıcıya verilmektedir.

Aşağıdaki örneklerde, belirlenen bazı özelliklerde bir halinin dokunabilmesi için gerekli olan ilmelik ve atkı ipi kalınlığının hesaplanmasında, (6.3-4)'deki yaklaşımların çıkarılmasını göstermektedir. Bu hesaplamalarda çözgü ipi kalınlığının bilindiği kabul edilmektedir. Karşılaştırma yapmak amacı ile dokuma iplerinden örnekler alınarak bunların kalınlıkları ölçülmüştür.

Örnek 1 : Örnek olarak alınan 1. halı çift düğüm tekniği dokunmuştur. Halinin dokunmasında kullanılan iplerin kalınlığı ölçün sonucu şöyle bulunmuştur:

Çözgü=1.3mm Atkı=1.5mm ilmelik=1mm-1.6mm arasında

Veriler:

10cm^2 'deki toplam düğüm sayısı=896

10cm^2 'deki enine düğüm sayısı=28

10cm^2 'deki boyuna düğüm sayısı=32

10cm^2 'deki atkı sayısı=64

10cm^2 'deki çözgü sayısı=56

iki düğüm sırası arasına çift sıra ara atkısı geçilmiştir.

Sipariş ebatı=262x194cm

işlem:

10cm 'deki toplam çözgü iplerinin kalınlığı=56 telx0.13cm
=7.28cm

10cm 'deki toplam ilmelik iplerinin kalınlığı= $10\text{cm}-7.28\text{cm}$
=2.72

iki çözgü arasındaki ilmelik ipi kalınlığı= $2.77\text{cm}/28$ düğüm
=0.097cm
=0.97mm

Atkı ipi kalınlığı= $10\text{cm}/64$ tel
=0.15cm
=1.5mm

örnek 2: örnek olarak alınan 2. halı çift düğüm tekniği ile dokunmuştur. Halının dokunduğu iplerin kalınlığı ölçüm sonucu şöyle bulunmuştur:

Çözgü=1.3mm Atkı=1.5mm ilmelik=1mm-1.6mm arasında

Veriler:

10cm² 'deki toplam düğüm sayısı =652.5

10cm² 'deki enine düğüm sayısı =22.5

10cm² 'deki boyuna düğüm sayısı =29

10cm² 'deki atkı sayısı =58

10cm² 'deki çözgü sayısı =45

iki düğüm sırası arasına çift sıra ara atkısı geçilmiştir.

Sipariş ebatı=234x161cm

işlem:

10cm'deki toplam çözgü iplerinin kalınlığı=45 telx0.13cm
=5.85cm

10cm'deki toplam ilmelik iplerinin kalınlığı=10cm-5.85cm
=4.15cm

iki çözgü arasındaki ilmelik kalınlığı=4.15cm/22.5 düğüm
=0.18cm
=1.8mm

Atkı ipe kalınlığı=10cm/58 tel

=0.17cm

=1.7mm

örnek 3: örnek olarak alınan 3. halı çift düğüm tekniği ile dokunmuştur. Halının dokunduğu iplerin kalınlığı ölçüm sonucu şöyle bulunmuştur:

Çözgü=1.3mm Atkı=1.5mm ilmelik=1mm-1.6mm arasında

Veriler:

10cm² 'deki toplam düğüm sayısı =851

10cm² 'deki enine düğüm sayısı =23

10cm² 'deki boyuna düğüm sayısı =37

10cm²'deki atkı sayısı =74

10cm²'deki çözgü sayısı =46

iki düğüm sırası arasına çift sıra ara atkısı geçilmiştir.

Sipariş ebatı=210x149cm

işlem:

10cm'deki toplam çözgü iplerinin kalınlığı=46 telx0.13cm
=5.98cm

10cm'deki toplam ilmelik iplerinin kalınlığı=10cm-5.98cm
=4.02cm

iki çözgü arasındaki ilmelik kalınlığı=4.02cm/23 düğüm
=0.17cm
=1.7mm

Atkı ipi kalınlığı=10cm/74 tel
=0.13cm
=1.3mm

örnek 4: örnek olarak alınan 4. halı tek düğüm tekniği ile dokunmuştur. Halının dokunduğu iplerin kalınlığı ölçüm sonucunda şöyle bulunmuştur:

Çözgü=1.6mm Atkı=1.4mm ilmelik=1.3mm

Veriler:

10cm²'deki toplam düğüm sayısı =957

10cm²'deki enine düğüm sayısı =29

10cm²'deki boyuna düğüm sayısı =33

10cm²'deki atkı sayısı =66

10cm²'deki çözgü sayısı =58

iki düğüm sırası arasına çift sıra ara atkısı geçilmiştir.

Sipariş ebatı=182x121cm

işlem:

Çözgü ipi kalınlığı=10cm/58 tel
=0.17cm
=1.7mm

Atkı ipi kalınlığı=10cm/66 tel
 =0.15cm
 =1.5mm

örnek 5: örnek olarak alınan 5. halı tek düğüm tekniği ile dokunmuştur. Halının dokunduğu iplerin kalınlığı ölçüm sonucu şöyle bulunmuştur:

Çözgü=1.6mm Atkı=1.4mm filmelik=1.3mm

Veriler:

10cm² 'deki toplam düğüm sayısı =945

10cm² 'deki enine düğüm sayısı =27

10cm² 'deki boyuna düğüm sayısı =35

10cm² 'deki atkı sayısı =70

10cm² 'deki çözgü sayısı =54

iki düğüm sırası arasına çift sıra ara atkısı geçilmiştir.

Sipariş ebatı=164x112cm

işlem:

Çözgü ipi kalınlığı=10cm/54 tel

=0.18cm

=1.8mm

Atkı ipi kalınlığı=10cm/70 tel

=0.14cm

=1.4mm

örneklerden de görüldüğü gibi, teorik olarak hesaplanan ip kalınlıkları ölçüm sonucu bulunan ip kalınlıkları ile oldukça yakın değerlerde olmaktadır.

8. SONUÇ

Bu tezde, sayısal bir resmin yeniden şekillendirilmesi ve değerlendirilmesi üzerine bir program geliştirilmiştir. Çalışmada, üzerinde inceleme ve değişiklik yapılacak olan desen dijitalizer ya da dijital kamera ile elde edilemediğinden, desen çizme işlemi de program içinde gerçekleştirilmektedir. Türk halıcılığında desen oluşturmak, desen üzerinde değişiklik yapmak ve desen çoğaltmak gibi konularda karşılaşılan emek ve zaman sorunlarına belli ölçülerde cevap olması açısından çalışmalar halı desenleri üzerinde yoğunlaştırılmaktadır.

Bilgisayardaki fonksiyon tuşları çizime yardımcı olacak bazı temel işlemlere programlanarak, desen üzerinde bazı kısımları çerçeve içine alarak istenen yerlere kopya etme veya taşıma işlemi gerçekleştirilmektedir. Böylece desen çizme işleminde kolaylık ve zaman tasarrufu sağlanmaktadır. Bilgisayarda çizim alanı olarak ayrılmış bölgede her nokta bir düğüme karşılık gelmektedir.

Renkli olarak çizilmiş bir halı deseninde, deseni oluşturan motiflerin iç kısımları boşaltılarak (siyaha dönüştürülerek) sadece motif sınırları bulunmaktadır. Sınırlar halinde belirlenen desen depolanırken de her düğüme ilişkin verinin saklanması yerine sadece sınırdaki düğümlere ait veriler depolanarak bilgisayar hafızasından da büyük ölçüde tasarruf edilmektedir.

Kütük içinde bir motifi tanımlayan bilgiler motif sınırının, başlangıç koordinatları, rengi, ölçek faktörü, döndürme açısı ve motifin sınırlarını belirleyen yönler uzunluklar şeklinde olmaktadır. Böylece bir motif üzerindeki değişiklikler, kütük içinde o motife ait bazı parametreleri değiştirerek kolaylıkla gerçekleştirilmektedir. Ayrıca programın her çizim aşamasında kullanıcı

yazıcıdan desen çıktısı alabilmektedir. Çalışma sırasında renkli yazıcı kullanılmadığından, her renk yazıcıdan farklı bir 8x8'lik nokta matrisi şeklinde ifade edilmektedir.

Programın diğer bir bölümünde de, özellikle el dokuması halılar için dokuma işleminden önce gerçek halı boyutunun ne olması gerektiğinin hesaplanması problemine de belli ölçüde bir çözüm getirmektedir. Tüm özellikleri belli (ip kalınlıkları, kalitesi, sipariş ebatı ve ara atkısı sayısı) ve pratikte uygulanmış olan halılar üzerinde yapılan çalışmalar sonucu, ip kalınlıklarının bir düğüm kalınlığı üzerindeki etkisi konusuna bir yaklaşım getirilmiştir. Bu yaklaşımların ışığı altında, çözgü kalınlığı, kalitesi ve sipariş ebatı verilen bir halının dokunduktan sonrada aynı kalite ve ebatı sağlayabilmesi için ilmelik ve atkı kalınlıklarının ne olması gerektiği program tarafından hesaplanmaktadır. Bulunan değerler olması gereken ip kalınlıklarıdır. Fakat üretilen ip kalınlıkları her zaman standart olmamaktadır. Ayrıca atkı ipinin çekilmesinden sonra dokuyucunun vurarak düğümleri sıkıştırma miktarı her zaman sabit olmamaktadır. Bu nedenle, boyuna düğüm kalınlığını etkileyen atkı ipinin kalınlığının hesaplanmasında da standart bir sıkıştırma olduğu varsayımı yapılmaktadır. Diğer taraftan tüm ip kalınlıkları programa verildiğinde, bu ip kalınlıklarına göre halı kalitesinin ne olacağı ve verilen sipariş ebatını gerçekleştirmek için halının eninde ve boyunda kaç düğüm olması gerektiğinin hesaplanması da programın gerçekleştirdiği fonksiyonlar arasındadır.

KAYNAKLAR

1. Uzaktan Algılamada Sayısal Görüntü İşleme
Fuat İnce, TÜBİTAK Elektronik Araştırma Ünitesi
Ünite İçi Rapor, 1986-Gebze
2. Yönlü Laplace Değişim Maskeleriyle Görüntü Kenarlarının
Bulunması
Erhan Alpaslan, TÜBİTAK Elektronik Araştırma Ünitesi
Ünite İçi Rapor, 1979-Gebze
3. Using The image Wise Video Digitizer
Steve Ciarcia, BYTE, July 1987-Tolland
4. Türk Halı Desen ve Motiflerinin Bilgisayar Yardımıyla
Saklanması, Geliştirilmesi ve Çoğaltılması
Figen Geçkinli, TÜBİTAK Gebze Araştırma Merkezi Elek-
trik Mühendisliği Ulusal Kongresi 1985
5. Waevers, Merchants And Kings (Halı Desen Örnekleri Ki-
tapçığı)

EK-A PROGRAMIN LİSTESİ

```

20 ' Program Name : TEZ1.BAS
800 'DIMENSIONS
810 DIM B$(15), RENK$(15), C$(8), FFF(300), F(300), F$(500)
900 PD$ = "KAPALI"
1000 ' main menu
1010 SCREEN 0, 0, 0: KEY OFF: FOR I = 1 TO 14: KEY(I) OFF: NEXT I
1020 WIDTH 80: CLS
1030 LOCATE 2, 30: PRINT "ANA LİSTE"
1035 LOCATE 4, 25: PRINT "0. PROGRAMDAN ÇIKIŞ"
1040 LOCATE 6, 25: PRINT "1. YENİ BİR ÇİZİME BAŞLAMA"
1050 LOCATE 8, 25: PRINT "2. MEVCUT BİR ÇİZİMDE DEĞİŞİKLİK YAPMA"
1060 LOCATE 10, 25: PRINT "3. RESMİN SINIRLARINI BULMA"
1070 LOCATE 12, 25: PRINT "4. SINIRLARDAN ORJİNAL RESMİ ELDE ETME"
1080 LOCATE 14, 25: PRINT "5. SINIRLAR ÜZERİNDE DEĞİŞİKLİK YAPMA"
1090 LOCATE 16, 25: PRINT "6. KUTUK İŞLEMLERİ"
1095 LOCATE 18, 25: PRINT "7. YAZICI (ACIK\KAPALI)"
1097 LOCATE 20, 25: PRINT "8. BOYUT HESAPLAMA"
1100 LOCATE 22, 28: INPUT "SEÇİMİNİZİ GİRİNİZ: ", SEL$
1110 IF SEL$ = "" THEN 1000
1120 SEL$ = LEFT$(SEL$, 1): IF SEL$ < "0" OR SEL$ > "8" THEN 1000
1130 ON VAL(SEL$) GOTO 10000, 12000, 20000, 30000, 35000, 50000, 51000,
1140 CLS : END
10000 LC = 0: PRINT : INPUT "KUTUK İSMİ :", FIL$: SCREEN 7, 0: CLS : KEY
10010 LOCATE 24, 1: PRINT "F1:Çizme F2:Silme F3:Normal F4:Renk F5:Ek
10020 LINE (1, 1)-(130, 130), 15, B
10030 GOSUB 11200
10040 'enable function keys 1 to 10
10050 FOR A = 1 TO 14
10060 KEY(A) OFF: KEY(A) ON
10070 NEXT A
10080 'DEFINE KEYS
10090 ON KEY(1) GOSUB 10330 'ÇİZME
10100 ON KEY(2) GOSUB 10630 'SİLME
10110 ON KEY(3) GOSUB 10380 'NORMAL
10120 ON KEY(4) GOSUB 10420 'RENK
10130 ON KEY(5) GOSUB 10720 'EKİRAN SİL
10140 ON KEY(6) GOSUB 10750 'PENCERE

```

```
10150 ON KEY(7) GOSUB 11640 'EDIT MODUNDAN CIKIE
10160 'DN KEY(8) GOSUB 4000
10170 'DN KEY(9) GOSUB 4500
10180 'DN KEY(10) GOSUB 5000
10190 ON KEY(11) GOSUB 11360 'YUKARI
10200 ON KEY(12) GOSUB 11430 'SOLA
10210 ON KEY(13) GOSUB 11500 'SAGA
10220 ON KEY(14) GOSUB 11570 'ASABI
10230 'SETTING
10240 COL = 15: X = 64: y = 64: EX = 0
10250 DMODE = 0: CMODE = 0
10260 LOCATE COL + 3, 19: PRINT "*": IF LC = 1 THEN GOSUB 12100
10270 'MAIN ROUTINE
10280 DC = POINT(X + 1, y + 1)
10290 PSET (X + 1, y + 1): FOR DLY = 1 TO 10: NEXT DLY
10300 PRESET (X + 1, y + 1): FOR DLY = 1 TO 10: NEXT DLY
10310 LOCATE 18, 1: PRINT USING "X:### Y:###": X: y
10320 IF EX = 1 THEN 1000 ELSE GOTO 10290
10330 'DRAW
10340 DMODE = 1
10350 CMODE = 0: DC = COL
10360 LOCATE 20, 1: PRINT "CIZGI MODU"
10370 RETURN
10380 'SDRAW
10390 DMODE = 0: CMODE = 0
10400 LOCATE 20, 1: PRINT SPACE$(20)
10410 RETURN
10420 'SELECT COLOR
10430 LOCATE COL + 3, 19: PRINT " "
10440 LOCATE 20, 20: INPUT "COLOR:", COL: LOCATE 20, 20: PRINT SPACE$(10)
10450 IF COL < 0 OR COL > 15 THEN 10440
10460 LOCATE COL + 3, 19: PRINT "*"
10470 RETURN
10480 '
10490 WIDTH 80: SCREEN 0: END
10500 RETURN
10510 'SAVE PICTURE
```

```
10515 PSET (X + 1, y + 1), DC
10520 LOCATE 20,1:INPUT"KUTUK ISMI:",FIL#
10530 LOCATE 20, 1: PRINT "SAVING          "
10540 OPEN "R", #1, FIL# + ".HLD", 1
10550 FIELD #1, 1 AS FF#
10560 I = 0: FOR X = 2 TO 128 STEP 2: FOR y = 2 TO 129: I = I + 1
10570 FF = POINT(X, y) * 16 + POINT(X + 1, y)
10580 LSET FF# = CHR$(FF)
10590 PUT #1, I
10600 NEXT y: NEXT X
10610 X = 64: y = 64: CLOSE #1: LOCATE 20, 1: PRINT SPACE$(20)
10620 RETURN
10630 'CLEAR LINE
10640 CMODE = 1: DMODE = 0
10650 DC = 0
10660 LOCATE 20, 1: PRINT "SILME MODU "
10670 RETURN
10680 'STOP CLEAR
10690 CMODE = 0: DMODE = 0
10700 LOCATE 20, 1: PRINT SPACE$(20)
10710 RETURN
10720 'CLEAR PICTURE
10730 LINE (2, 2)-(129, 129), 0, BF
10740 RETURN
10750 'WINDOW
10760 PSET (X + 1, y + 1), DC
10770 LOCATE 20, 1: PRINT "PENCERE          ": PCOPY 0, 1
10780 X1 = X: Y1 = y
10790 CMODE = 0: DMODE = 0
10800 PCOPY 1, 0
10810 LINE (X1 + 1, Y1 + 1)-(X + 1, y + 1), 15, B
10820 FOR DLYW = 1 TO 30: NEXT DLYW
10830 R# = INKEY#: IF R# = CHR$(13) THEN 10850
10840 GOTO 10800
10850 X2 = X: Y2 = y
10860 PCOPY 1, 0: LOCATE 20, 1: PRINT "SILME KOPYA TASIMA BOYUT"
10870 DC = POINT(X + 1, y + 1)
```

```
10880 PSET (X + 1, y + 1): FOR DL = 1 TO 10: NEXT DL
10890 PRESET (X + 1, y + 1): FOR DL = 1 TO 10: NEXT DL
10900 LOCATE 18, 1: PRINT USING "X:### Y:###": X; y
10910 A$ = INKEY$: IF A$ = "" THEN 10880
10920 IF X1 > X2 THEN SWAP X1, X2
10930 IF Y1 > Y2 THEN SWAP Y1, Y2
10940 IF A$ = "S" OR A$ = "s" THEN 10980
10950 IF A$ = "K" OR A$ = "k" THEN 11040
10960 IF A$ = "T" OR A$ = "t" THEN 11110
10965 IF A$ = "B" OR A$ = "b" THEN 55000
10970 GOTO 10880
10980 'ERASE
10990 FOR I = X1 TO X2: FOR J = Y1 TO Y2
11000 PSET (I + 1, J + 1), 0
11010 NEXT J: NEXT I
11020 LOCATE 20, 1: PRINT SPACE$(25)
11030 RETURN
11040 'COPY
11050 FOR I = X1 TO X2: FOR J = Y1 TO Y2
11060 I1 = X + I - X1 + 1: J1 = y + J - Y1 + 1: IF I1 > 128 OR J1 > 128
11070 PSET (I1, J1), POINT(I + 1, J + 1)
11080 NEXT J: NEXT I
11090 LOCATE 20, 1: PRINT SPACE$(25)
11100 RETURN
11110 'MOVE
11120 FOR I = X1 TO X2: FOR J = Y1 TO Y2
11130 I1 = X + I - X1 + 1: J1 = y + J - Y1 + 1: IF I1 > 128 OR J1 > 128
11140 PSET (I1, J1), POINT(I + 1, J + 1)
11150 PSET (I + 1, J + 1), 0
11160 NEXT J: NEXT I
11170 LOCATE 20, 1: PRINT SPACE$(25)
11180 RETURN
11190 RETURN
11200 'COLOR FUNCTION
11210 LOCATE 1, 20
11220 PRINT "NO": PRINT
11230 FOR NUM = 0 TO 15
```

```
11240 LOCATE NUM + 3, 20: PRINT NUM
11250 NEXT NUM
11260 LOCATE 1, 25: PRINT "RENK"
11270 LOCATE 3, 25: PRINT "SIYAH"
11280 /
11285 RESTORE 11340
11290 FOR RENK = 1 TO 15
11300 READ B$(RENK)
11310 COLOR RENK, 0
11320 LOCATE RENK + 3, 25: PRINT B$(RENK)
11330 NEXT RENK
11340 DATA "MAVI", "YESIL", "CAM.G.MAVI", "KIRMIZI", "MOR", "KAHVERENGI", "BEYA"
11350 RETURN
11360 /UP
11370 IF DMODE = 1 THEN DC = COL
11380 IF CMODE = 1 THEN DC = 0
11390 PSET (X + 1, y + 1), DC
11400 y = y - 1: IF y <= 0 THEN y = 128
11410 DC = POINT(X + 1, y + 1)
11420 RETURN
11430 /LEFT
11440 IF DMODE = 1 THEN DC = COL
11450 IF CMODE = 1 THEN DC = 0
11460 PSET (X + 1, y + 1), DC
11470 X = X - 1: IF X <= 0 THEN X = 128
11480 DC = POINT(X + 1, y + 1)
11490 RETURN
11500 /RIGHT
11510 IF DMODE = 1 THEN DC = COL
11520 IF CMODE = 1 THEN DC = 0
11530 PSET (X + 1, y + 1), DC
11540 X = X + 1: IF X > 128 THEN X = 1
11550 DC = POINT(X + 1, y + 1)
11560 RETURN
11570 /DOWN
11580 IF DMODE = 1 THEN DC = COL
11590 IF CMODE = 1 THEN DC = 0
```



```
11600 PSET (X + 1, y + 1), DC
11610 y = y + 1: IF y > 128 THEN y = 1
11620 DC = POINT(X + 1, y + 1)
11630 RETURN
11640 'exit
11650 EX = 1
11660 LOCATE 20, 1: INPUT "DEPOLAMAK ISTIYORMUSUNUZ (E\H) :", y#
11670 IF y# = "E" OR y# = "e" THEN GOSUB 10510
11680 RETURN
12000 'EDIT AN EXISTING DRAWING
12010 PRINT : INPUT "KUTUK ISMI:", FIL#
12020 LC = 1
12030 SCREEN 7, 0: CLS : KEY OFF
12040 GOTO 10010
12100 'LOAD FILE
12110 LOCATE 20, 1: PRINT "RESIM YUKLENIYOR"
12120 OPEN "R", #1, FIL# + ".HLD", 1
12130 FIELD #1, 1 AS FF#
12140 I = 0: FOR X = 2 TO 128 STEP 2
12150 FOR y = 2 TO 129
12160 I = I + 1
12170 GET #1, I
12180 FF = ASC(FF#)
12190 A = INT(FF / 16): B = FF - A * 16
12200 PSET (X, y), A
12210 PSET (X + 1, y), B
12220 NEXT y
12230 NEXT X
12240 X = 64: y = 64
12250 CLOSE #1
12260 LOCATE 20, 1: PRINT SPACE$(20)
12270 RETURN
20000 'CONVERT PICTURE TO LINE FORMAT
20005 PRINT : INPUT "KUTUK ISMI:", FIL#
20010 SCREEN 7, 0: CLS : KEY OFF: C$(1) = "r": C$(2) = "d": C$(3) = "l":
20020 LINE (1, 1)-(130, 130), 15, B: LINE (132, 1)-(261, 130), 15, B
20030 COLOR 4, 0: LOCATE 18, 9: PRINT "A" B": COLOR 15, 0
```

```
20040 GOSUB 20060
20050 GOSUB 12100: GOTO 20130
20060 'COLOR FUNCTION
20070 LOCATE 1, 34: PRINT "RENK"
20080 FOR RENK = 1 TO 15
20090 COLOR RENK, 0
20100 LOCATE RENK + 2, 35: PRINT RENK
20110 NEXT RENK
20120 RETURN
20130 'edge finding
20140 FOR J = 2 TO 129
20150 FOR I = 2 TO 129
20160 A = ABS(POINT(I, J - 1) - POINT(I - 1, J))
20170 B = ABS(POINT(I, J - 1) - POINT(I, J + 1))
20180 C = 0: 'C=ABS(POINT(I+1,J-1)-POINT(I-1,J+1))
20190 d = ABS(POINT(I - 1, J) - POINT(I + 1, J))
20200 NEWPV = A + B + C + d
20210 IF NEWPV = 0 THEN PSET (I + 131, J), 0 ELSE PSET (I + 131, J), POIN
20220 NEXT I: NEXT J
20230 LINE (2, 2)-(129, 129), 0, BF
20240 FOR I = 2 TO 129
20250 FOR J = 2 TO 129
20260 PSET (I, J), POINT(I + 131, J)
20270 PSET (I + 131, J), 0
20280 NEXT J: NEXT I
20300 'edge following
20310 GOSUB 20310
20320 FOR ROW = 2 TO 128
20330 FOR COL = 2 TO 128: DDR = 0
20340 IF POINT(COL, ROW) <> 0 THEN GOSUB 20370
20350 NEXT COL: NEXT ROW: CLOSE #1
20360 GOTO 1000
20370 'edge describe
20380 XST = COL: YST = ROW
20390 BC = POINT(XST, YST): PSET (XST + 131, YST), BC
20400 '
20410 X = COL: y = ROW
```

```

20420 RIGHT = POINT(X + 1, y): RD = POINT(X + 1, y + 1)
20430 DOWN = POINT(X, y + 1): DL = POINT(X - 1, y + 1)
20440 LEFT = POINT(X - 1, y): LU = POINT(X - 1, y - 1)
20450 UP = POINT(X, y - 1): UR = POINT(X + 1, y - 1)
20460 IF RIGHT = BC THEN DRC = 1: GOSUB 20740: GOTO 20550
20470 IF DOWN = BC THEN DRC = 2: GOSUB 20740: GOTO 20580
20480 IF LEFT = BC THEN DRC = 3: GOSUB 20740: GOTO 20610
20490 IF UP = BC THEN DRC = 4: GOSUB 20740: GOTO 20640
20500 IF RD = BC THEN DRC = 5: GOSUB 20740: GOTO 20670
20510 IF DL = BC THEN DRC = 6: GOSUB 20740: GOTO 20680
20520 IF LU = BC THEN DRC = 7: GOSUB 20740: GOTO 20690
20530 IF UR = BC THEN DRC = 8: GOSUB 20740: GOTO 20700
20540 PSET (COL, ROW), 0: DRC = 0: GOTO 20740
20550 'right
20560 X = X + 1: PSET (X, y), 0: GOSUB 20710
20570 GOTO 20720
20580 'down
20590 y = y + 1: PSET (X, y), 0: GOSUB 20710
20600 GOTO 20720
20610 'left
20620 X = X - 1: PSET (X, y), 0: GOSUB 20710
20630 GOTO 20720
20640 'up
20650 y = y - 1: PSET (X, y), 0: GOSUB 20710
20660 GOTO 20720
20670 X = X + 1: y = y + 1: PSET (X, y), 0: GOSUB 20710: GOTO 20720
20680 X = X - 1: y = y + 1: PSET (X, y), 0: GOSUB 20710: GOTO 20720
20690 X = X - 1: y = y - 1: PSET (X, y), 0: GOSUB 20710: GOTO 20720
20700 X = X + 1: y = y - 1: PSET (X, y), 0: GOSUB 20710: GOTO 20720
20710 PSET (X + 131, y), BC: RETURN
20720 IF X = XST AND y = YST THEN DRC = 0: GOSUB 20740: RETURN
20730 GOTO 20420
20740 IF DDR = 0 THEN 20770
20750 IF DRC = 0 THEN GOSUB 20870: FFF(FC) = 0: GOTO 20850
20760 IF DDR (<) DRC THEN GOTO 20870 ELSE CNT = CNT + 1: RETURN
20770 FFF(1) = COL: FFF(2) = ROW: FFF(3) = BC: FFF(4) = 0: FFF(5) = 0
20780 FC = 6

```

```

20790 IF DRC = 0 THEN FFF(FC) = 0: GOTO 20850
20800 DDR = DRC: CNT = 1: RETURN
20810 LOCATE 21, 1: 'INPUT "file:",FI#
20820 OPEN "r", #1, FIL# + ".for", 1
20830 FIELD #1, 1 AS A#
20840 FFC = 1: RETURN
20850 FOR I = 1 TO FC: LSET A# = CHR$(FFF(I))
20860 PUT #1, FFC: FFC = FFC + 1: NEXT I: RETURN
20870 'if cnt=0 then return
20880 IF CNT > 31 THEN FFF(FC) = (DDR - 1) * 32 + 31: CNT = CNT - 31: FC
20890 FFF(FC) = (DDR - 1) * 32 + CNT: FC = FC + 1: DDR = DRC: CNT = 1: RE
30000 'convert line format to picture
30010 PRINT : INPUT "KUTUK ISMI:", FIL#
30020 '
30030 OPEN "r", #1, FIL# + ".for", 1
30040 FIELD #1, 1 AS A#
30050 SCREEN 7: CLS : LINE (1, 1)-(130, 130), 15, B
30060 C$(1) = "R": C$(2) = "D": C$(3) = "L": C$(4) = "U": C$(5) = "f": C$
30070 FFC = 1
30080 WHILE NOT (EOF(1))
30090 FOR FC = 1 TO 5
30100 GET #1, FFC
30110 FFC = FFC + 1
30120 F(FC) = ASC(A#)
30130 NEXT FC: FC = 6
30140 'dddd:
30150 GET #1, FFC: FFC = FFC + 1
30160 A = ASC(A#): IF A = 0 THEN GOSUB 30270: GOTO 30180
30170 F(FC) = A: FC = FC + 1: GOTO 30140
30180 'ffff:
30190 WEND: CLOSE #1: LOCATE 20, 1: INPUT "RESMI DEPOLAMAK ISTIYORMUSUNUZ
30191 IF y# = "E" OR y# = "e" THEN LOCATE 20, 1: PRINT SPACE$(30): LOCATE
30192 LOCATE 20, 1: INPUT "RESMI DOLDURMAK ISTIYORMUSUNUZ (E/H) ", y#
30193 IF y# = "E" OR y# = "e" THEN 30200
30194 GOTO 1000
30200 'doldurma
30210 FOR X = 2 TO 129

```

```
30220 TOG = 0
30230 FOR y = 2 TO 129
30240 IF POINT(X, y) = 0 THEN GOSUB 30370
30250 NEXT y: NEXT X
30260 LOCATE 20, 1: INPUT "RESMI DEPOLAMAK ISTIYORMUSUNUZ (E/H) ", y$
30261 IF y$ = "E" OR y$ = "e" THEN GOSUB 10510
30262 GOTO 1000
30270 'picture:
30280 B$ = ""
30290 PSET (F(1), F(2)), F(3): IF FC = 6 THEN RETURN
30300 '
30310 FOR I = 6 TO FC - 1
30320 d = INT(F(I) / 32)
30330 UZ = INT((F(I) / 32 - d) * 32 + .5)
30335 IF F(4) = 0 THEN F(4) = 4
30340 B$ = "S" + STR$(F(4)) + "TA" + STR$(F(5) * 45) + C$(d + 1) + STR$(
30350 NEXT I
30360 RETURN
30370 'biryer:
30380 IF POINT(X, y - 1) = 0 THEN RETURN ELSE BC = POINT(X, y - 1)
30390 IF TOG = BC THEN TOG = 0: RETURN
30400 YR = y - 1: A = 0
30410 WHILE YR < 129 AND A = 0
30420 YR = YR + 1
30430 A = POINT(X, YR)
30440 WEND
30450 IF A = BC THEN LINE (X, y)-(X, YR), BC: y = YR
30460 'if point(x,y+1)=0 then tog=bc else tog=0
30470 RETURN
35000 'edit
35010 PRINT : INPUT "KUTUK ISMI:", FIL$
35020 PRINT "LUTFEN BEKLEYIN"
35030 NDL = 0
35040 OPEN "R", #1, FIL$ + ".FOR", 1
35050 FIELD #1, 1 AS A$
35055 RC = 1
35060 WHILE NOT (EOF(1))
```

```
35070 NOL = NOL + 1
35080 GET #1, RC
35090 RC = RC + 1
35100 F$(NOL) = ""
35110 IF ASC(A$) = 0 AND LEN(F$(NOL)) >= 5 THEN 35150
35120 F$(NOL) = F$(NOL) + A$
35130 GET #1, RC
35135 RC = RC + 1
35140 GOTO 35110
35150 WEND: CLOSE #1
35160 SCREEN 7: CLS
35170 C$(1) = "R": C$(2) = "D": C$(3) = "L": C$(4) = "U": C$(5) = "F": C$
35180 LINE (1, 1)-(130, 130), 15, B: LINE (132, 1)-(261, 130), 15, B
35195 LOCATE 23, 1: PRINT "F1:Renk+ F2:Renk- F3:Olcek+ F4:Olcek- F5:Don
35200 GOSUB 20060
35210 'KEY DEF
35220 FOR I = 1 TO 14
35230 KEY(I) OFF: KEY(I) ON
35240 NEXT I
35250 ON KEY(1) GOSUB 36000'COLOR INC
35260 ON KEY(2) GOSUB 36100'COLOR DEC
35270 ON KEY(3) GOSUB 36200'MAGNI INC
35280 ON KEY(4) GOSUB 36300'MAGNI DEC
35290 ON KEY(5) GOSUB 36400'ROTATE CCW
35300 ON KEY(6) GOSUB 36500'ROTATE CW
35310 ON KEY(9) GOSUB 36600'CLEAR LINE
35320 ON KEY(7) GOSUB 36700'SELECT
35325 ON KEY(8) GOSUB 37200'ACCEPT
35326 ON KEY(10) GOSUB 37300'EXIT
35330 ON KEY(11) GOSUB 36800'UP
35340 ON KEY(12) GOSUB 36900'LEFT
35350 ON KEY(13) GOSUB 37000'RIGHT
35360 ON KEY(14) GOSUB 37100'DOWN
35370 '
35380 EX = 0
35390 GOTO 35560
35400 'CIZDIRME
```

```

35410 FOR I = 1 TO NQL
35414 XD = 0: YD = 0
35415 GOSUB 35420: GOTO 35540
35420 IF LEN(F$(I)) = 0 THEN RETURN
35425 X = ASC(MID$(F$(I), 1, 1))
35430 Y = ASC(MID$(F$(I), 2, 1))
35440 C = ASC(MID$(F$(I), 3, 1))
35450 M = ASC(MID$(F$(I), 4, 1)): IF M = 0 THEN M = 4
35460 A = ASC(MID$(F$(I), 5, 1))
35470 PSET (X + XD, Y + YD), C
35480 FOR J = 6 TO LEN(F$(I))
35490 DI = INT(ASC(MID$(F$(I), J, 1)) / 32)
35500 LE = INT((ASC(MID$(F$(I), J, 1)) / 32 - DI) * 32 + .5)
35510 B$ = "S" + STR$(M) + "TA" + STR$(A * 45) + C$(DI + 1) + STR$(LE)
35520 DRAW B$
35530 NEXT J
35535 RETURN
35540 NEXT I
35550 RETURN
35560 '
35570 GOSUB 35400
35580 SEL = 0: ACC = 0
35590 DS = 0
35600 SE = 1
35610 IF DS = SE THEN 35700
35620 LINE (133, 2)-(260, 129), 0, BF
35630 XD = 131: I = SE: DS = SE
35640 GOSUB 35420
35700 IF EX = 1 THEN 1000 ELSE IF SEL = 0 THEN 35610
35710 SE$ = F$(SE)
35712 COLOR 4, 0: LOCATE 18, 1: PRINT USING "X:### Y:### DLCEK:##.# AC"
35720 XD = 131: YD = 0: CO = C: MA = M: AN = A'ORIGINAL C,MA,AN
35730 CH = 0
35735 LOCATE 19, 1: PRINT USING "X:### Y:### DLCEK:##.# ACI:### RENK"
35740 LINE (133, 2)-(260, 129), 0, BF
35745 IF SE$ = "" THEN 35765
35750 PSET (X + XD, Y + YD), CO

```

```
35760 GOSUB 35480
35765 IF ACC = 1 THEN ACC = 0: SEL = 0: DS = 0: YD = 0: GOTO 35620
35770 IF CH = 0 THEN 35765
35780 IF EX = 0 THEN 35730
35790 GOTO 1000
36000 'COLOR INC
36010 IF CD < 15 THEN CD = CD + 1
36015 CH = 1
36020 RETURN
36100 'COLOR DEC
36110 IF CD > 1 THEN CD = CD - 1
36115 CH = 1
36120 RETURN
36200 'MAGNI INC
36210 IF M < 250 THEN M = M + 1
36215 CH = 1
36220 RETURN
36300 'MAGNI DEC
36310 IF M > 1 THEN M = M - 1
36315 CH = 1
36320 RETURN
36400 'ROTATE CCW
36410 A = A + 1
36420 IF A > 7 THEN A = 0
36425 CH = 1
36430 RETURN
36500 'ROTATE CW
36510 A = A - 1
36520 IF A < 0 THEN A = 7
36525 CH = 1
36530 RETURN
36600 'CLEAR LINE
36610 SE# = ""
36615 CH = 1
36620 RETURN
36700 'SELECT
36705 SEL = 1
```



```
36710 RETURN
36800 'UP
36810 IF SEL = 0 THEN IF SE > 1 THEN SE = SE - 1: GOTO 36830
36820 IF v + YD > 2 THEN YD = YD - 1
36825 CH = 1
36830 RETURN
36900 'LEFT
36910 IF X + XD > 139 THEN XD = XD - 1
36915 CH = 1
36920 RETURN
37000 'RIGHT
37010 IF X + XD < 260 THEN XD = XD + 1
37015 CH = 1
37020 RETURN
37100 'DOWN
37110 IF SEL = 0 THEN IF SE < NOL THEN SE = SE + 1: RETURN
37120 IF v + YD < 129 THEN YD = YD + 1
37125 CH = 1
37130 RETURN
37200 'ACCEPT
37210 ACC = 1: LOCATE 22, 1: INPUT "DEGISIKLIK IPTALI (E/H):", y$: LOCATE
37215 IF SE$ = "" THEN 37250
37220 X = X + XD - 131: y = y + YD
37230 AC$ = CHR$(X) + CHR$(y) + CHR$(CO) + CHR$(M) + CHR$(A)
37240 SE$ = AC$ + RIGHT$(SE$, LEN(SE$) - 5)
37250 F$(SE) = SE$
37260 LINE (2, 2)-(129, 129), 0, BF
37270 GOTO 35400
37300 'EXIT
37310 LOCATE 22, 1: INPUT "RESIM DEPOLANIYOR (E/H):", y$
37320 IF y$ = "E" OR y$ = "e" THEN GOSUB 45000
37330 EX = 1: CH = 1: RETURN
45000 'SAVE LINE EDIT PICTURE
45010 KILL FIL$ + ".FOR"
45020 OPEN "R", #1, FIL$ + ".FOR", 1
45030 FIELD #1, 1 AS A$
45040 RC = 1
```

```
45050 FOR I = 1 TO NOL
45060 IF LEN(F$(I)) = 0 THEN 45130
45070 FOR J = 1 TO LEN(F$(I))
45080 LSET A$ = MID$(F$(I), J, 1)
45090 PUT #1, RC: RC = RC + 1
45100 NEXT J
45110 LSET A$ = CHR$(0)
45120 PUT #1, RC: RC = RC + 1
45130 NEXT I
45140 CLOSE #1
45150 RETURN

50000 'FILE UTILITIES
50010 CLS : LOCATE 5, 25: PRINT "KUTUK ISLEM LISTESI"
50020 LOCATE 7, 25: PRINT "0. ANA LISTEYE DONUS"
50030 LOCATE 9, 25: PRINT "1. CIZIM KUTUKLERININ LISTESI"
50040 LOCATE 11, 25: PRINT "2. CIZGI FORMUNDAKI KUTUKLERIN LISTESI"
50050 LOCATE 13, 25: PRINT "3. KUTUK ISMINI DEGISTIRME"
50060 LOCATE 15, 25: PRINT "4. KUTUK SILME"
50080 LOCATE 19, 28: INPUT "SECIMINIZI GIRINIZ : ", SEL1$
50090 IF SEL1$ = "" THEN 50000
50100 SEL1$ = LEFT$(SEL1$, 1): IF SEL1$ < "0" OR SEL1$ > "4" THEN 50000
50110 ON VAL(SEL1$) GOTO 50200, 50300, 50400, 50500
50120 GOTO 1000
50200 'LIST DRAWING FILES
50210 CLS
50220 FILES "*.HLD"
50230 PRINT : PRINT "DEVAM ETMEK ICIN HERHANGI BIR TUSA BASINIZ"
50240 WHILE INKEY$ = "": WEND
50250 GOTO 50000
50300 'LIST LINE FORMAT FILES
50310 CLS
50320 FILES "*.FOR"
50330 PRINT : PRINT "DEVAM ETMEK ICIN HERHANGI BIR TUSA BASINIZ"
50340 WHILE INKEY$ = "": WEND
50350 GOTO 50000
50400 'RENAME FILES
50410 PRINT : INPUT "ESKI Kutukismi: ", DLF$
```

```
50420 PRINT : INPUT "YENI Kutukismi: ", NEF#
50430 PRINT
50440 NAME OLF# AS NEF#
50450 GOTO 50000
50500 'DELETE FILES]
50510 PRINT : INPUT "KUTUKISMI: ", NEF#
50520 KILL NEF#
50530 GOTO 50000
51000 'PRINTOUT
51010 CLS
51020 PRINT "YAZICI MODU "; PO#
51030 PRINT "BU MODU DEGISTIRMEK ISTIYORMUSUNUZ ?";
51040 INPUT " E/H : ", y#
51050 IF y# = "E" OR y# = "e" THEN 51060 ELSE GOTO 1000
51060 IF PO# = "KAPALI" THEN PO# = "ACIK" ELSE PO# = "KAPALI"
51070 IF PO# = "KAPALI" THEN KEY(15) OFF: GOTO 1000
51080 KEY 15, CHR#(&H4) + CHR#(&H19)
51090 KEY(15) ON
51100 ON KEY(15) GOSUB 51200
51110 GOTO 1000
51200 'PRINT SCREEN
51210 WIDTH "LPT1:", 255
51220 LOCATE 21, 1: PRINT "YAZICI HAZIR OLDUGUNDA HERHANGI BIR TUSA BASIM
51230 WHILE INKEY# = "": WEND
51240 LOCATE 21, 1: PRINT "LUTFEN BEKLEYIN "
51250 LPRINT CHR#(27); "3"; CHR#(29)
51260 ROW1 = 2: ROW2 = 65: COL1 = 2: COL2 = 65
51270 LPRINT "KUTUKISMI:"; FIL#
51280 LPRINT "BOLUM 1": LPRINT
51290 GOSUB 52000
51300 ROW1 = 66: ROW2 = 129
51310 LPRINT : LPRINT : LPRINT : LPRINT : LPRINT : LPRINT : LPRINT "BOLU
51320 GOSUB 52000
51330 ROW1 = 2: ROW2 = 65: COL1 = 66: COL2 = 129
51340 LPRINT : LPRINT : LPRINT : LPRINT : LPRINT : LPRINT : LPRINT "BOLU
51350 GOSUB 52000
51360 ROW1 = 66: ROW2 = 129
```

```

51370 LPRINT : LPRINT : LPRINT : LPRINT : LPRINT : LPRINT : LPRINT "BOLL
51380 GOSUB 52000
51390 LOCATE 21, 1: PRINT "
51400 RETURN
52000 FOR XP = ROW1 TO ROW2
52010 LPRINT CHR$(27); "K"; CHR$(0); CHR$(2);
52020 FOR YP = COL1 TO COL2
52030 PP = POINT(YP, XP)
52040 ON PP + 1 GOSUB 52500, 52510, 52520, 52530, 52540, 52550, 52560, 5
52050 NEXT YP
52060 LPRINT
52070 NEXT XP
52080 RETURN
52500 LPRINT CHR$(0); CHR$(0); CHR$(0); CHR$(0); CHR$(0); CHR$(0); CHR$(
52510 LPRINT CHR$(0); CHR$(255); CHR$(153); CHR$(153); CHR$(255); CHR$(1
52520 LPRINT CHR$(0); CHR$(255); CHR$(153); CHR$(153); CHR$(153); CHR$(1
52530 LPRINT CHR$(0); CHR$(255); CHR$(129); CHR$(129); CHR$(255); CHR$(1
52540 LPRINT CHR$(0); CHR$(255); CHR$(225); CHR$(177); CHR$(152); CHR$(1
52550 LPRINT CHR$(0); CHR$(255); CHR$(195); CHR$(165); CHR$(153); CHR$(1
52560 LPRINT CHR$(0); CHR$(255); CHR$(129); CHR$(153); CHR$(165); CHR$(1
52570 LPRINT CHR$(0); CHR$(255); CHR$(255); CHR$(255); CHR$(255); CHR$(2
52580 LPRINT CHR$(0); CHR$(255); CHR$(231); CHR$(195); CHR$(129); CHR$(1
52590 LPRINT CHR$(0); CHR$(255); CHR$(143); CHR$(143); CHR$(255); CHR$(2
52600 LPRINT CHR$(0); CHR$(255); CHR$(241); CHR$(241); CHR$(241); CHR$(2
52610 LPRINT CHR$(0); CHR$(255); CHR$(129); CHR$(255); CHR$(129); CHR$(2
52620 LPRINT CHR$(0); CHR$(255); CHR$(253); CHR$(248); CHR$(241); CHR$(2
52630 LPRINT CHR$(0); CHR$(255); CHR$(195); CHR$(231); CHR$(255); CHR$(2
52640 LPRINT CHR$(0); CHR$(255); CHR$(171); CHR$(213); CHR$(171); CHR$(2
52650 LPRINT CHR$(0); CHR$(255); CHR$(129); CHR$(189); CHR$(189); CHR$(1
53000 'BOYUT HESAPLAMA
53010 CLS : KEY OFF
53020 LOCATE 5, 30: PRINT "BOYUT HESAPLAMA LISTESI"
53030 LOCATE 7, 25: PRINT "0. ANA LISTEYE DONUS"
53040 LOCATE 9, 25: PRINT "1. IP KALINLIGI HESABI"
53050 LOCATE 11, 25: PRINT "2. IP KALINLIGINA GORE KALITE HESABI"
53060 LOCATE 13, 25: INPUT "SECIMINIZI GIRINIZ : ", SEC$
53070 IF SEC$ = "" THEN 53020

```

```

53080 SEC$ = LEFT$(SEC$, 1): IF SEC$ ( "0" OR SEC$ ) "2" THEN 53020
53090 ON VAL(SEC$) GOTO 53150, 53500
53100 GOTO 1000
53150 'IP KALINLIGI HESABI
53160 CLS
53170 LOCATE 2, 2: COLOR 15, 0: PRINT "VERILER"
53180 LOCATE 4, 5: COLOR 4, 0: PRINT "HALININ KALITESI"
53190 LOCATE 6, 5: PRINT "10'cmdeki enine dugum sayisi      (<"; EDS; " "
53200 LOCATE 7, 5: PRINT "10'cmdeki boyuna dugum sayisi    (<"; BDS; " "
53210 LOCATE 8, 5: PRINT "Cozgu ipi kalinligi(mm)          (<"; CK; " "
53220 LOCATE 9, 5: PRINT "Iki dugum arasindaki ara atki sayisi (<"; AT; " "
53230 LOCATE 10, 5: PRINT "SIPARIS EBATI"
53240 LOCATE 11, 5: PRINT "En(cm) <"; EN; ">"; TAB(22); ":"
53250 LOCATE 12, 5: PRINT "Boy(cm) <"; BOY; ">"; TAB(22); ":"
53260 LOCATE 15, 2: COLOR 15, 0: PRINT "SONUCLAR"
53270 LOCATE 16, 5: COLOR 4, 0: PRINT "Halinin enine dugum sayisi  :"
53280 LOCATE 17, 5: PRINT "Halinin boyuna dugum sayisi  :"
53290 LOCATE 18, 5: PRINT "Ilmelik ipi kalinligi(mm)   :"
53300 LOCATE 19, 5: PRINT "Atki ipi kalinligi(mm)     :"
53310 COLOR 15, 0: LOCATE 6, 51: INPUT "", ARP: IF ARP (<) 0 THEN EDS = ARP
53320 LOCATE 7, 51: INPUT "", ARP: IF ARP (<) 0 THEN BDS = ARP
53330 LOCATE 8, 51: INPUT "", ARP: IF ARP (<) 0 THEN CK = ARP
53340 LOCATE 9, 51: INPUT "", ARP: IF ARP (<) 0 THEN AT = ARP
53350 LOCATE 11, 23: INPUT "", ARP: IF ARP (<) 0 THEN EN = ARP
53360 LOCATE 12, 23: INPUT "", ARP: IF ARP (<) 0 THEN BOY = ARP
53370 IPK = INT((10 - EDS * 2 * CK * .1) / EDS * 100) / 100
53380 ATK = INT((10 / BDS) / AT * 100) / 100
53390 HEDS = INT(EN * EDS / 10 * 100) / 100: HBDS = INT(BOY * BDS / 10 *
53400 LOCATE 16, 34: PRINT HEDS
53410 LOCATE 17, 34: PRINT HBDS
53420 LOCATE 18, 34: PRINT IPK * 10
53430 LOCATE 19, 34: PRINT ATK * 10
53440 LOCATE 22, 5: PRINT "Yapilan islemler dogru mu ?. (E/H) "
53450 CUG$ = "": WHILE CUG$ = "": CUG$ = INKEY$: WEND
53460 IF CUG$ = "H" OR CUG$ = "h" THEN 53150 ELSE IF CUG$ = "E" OR CUG$ =
53500 'ip kalinligina gore kalite hesabi
53505 CLS

```

```

53510 LOCATE 2, 2: COLOR 15, 0: PRINT "VERILER"
53520 LOCATE 4, 5: COLOR 4, 0: PRINT "Cozgu ipi kalinligi(mm)
53530 LOCATE 5, 5: PRINT "Atki ipi kalinligi (mm)          (<"; AT
53540 LOCATE 6, 5: PRINT "Ilmelik ipi kalinligi (mm)      (<"; IP
53550 LOCATE 7, 5: PRINT "Iki dugum arasindaki ara atkisi sayisi (<"; AT
53560 LOCATE 8, 5: PRINT "Siparis ebati"
53570 LOCATE 9, 5: PRINT "En(cm) (<"; EN; ">"; TAB(22); ":"
53580 LOCATE 10, 5: PRINT "Boy(cm) (<"; BOY; ">"; TAB(22); ":"
53590 LOCATE 13, 2: COLOR 15, 0: PRINT "SONUCLAR"
53600 LOCATE 14, 5: COLOR 4, 0: PRINT "Halinin enine dugum sayisi      :"
53610 LOCATE 15, 5: PRINT "Halinin boyuna dugum sayisi      :"
53620 LOCATE 16, 5: PRINT "Halinin kalitesi"
53630 LOCATE 17, 5: PRINT "10cm'deki enine dugum sayisi      :"
53640 LOCATE 18, 5: PRINT "10cm'deki boyuna dugum sayisi      :"
53650 LOCATE 4, 56: COLOR 15, 0: INPUT "", ARP: IF ARP (<) 0 THEN CK = AR
53660 LOCATE 5, 56: INPUT "", ARP: IF ARP (<) 0 THEN ATK = ARP
53670 LOCATE 6, 56: INPUT "", ARP: IF ARP (<) 0 THEN IPK = ARP
53680 LOCATE 7, 56: INPUT "", ARP: IF ARP (<) 0 THEN AT = ARP
53690 LOCATE 9, 23: INPUT "", ARP: IF ARP (<) 0 THEN EN = ARP
53700 LOCATE 10, 23: INPUT "", ARP: IF ARP (<) 0 THEN BOY = ARP
53710 BEDK = INT((2 * CK + IPK) * .1 * 100) / 100
53720 EDS = INT(10 / BEDK * 100) / 100
53730 BBDK = INT((ATK * AT) * .1 * 100) / 100
53740 BDS = INT(10 / BBDK * 100) / 100
53750 HEDS = INT(EN * EDS / 10 * 100) / 100: HBDS = INT(BOY * BDS / 10
53760 LOCATE 14, 35: PRINT HEDS
53770 LOCATE 15, 35: PRINT HBDS
53780 LOCATE 17, 35: PRINT EDS
53790 LOCATE 18, 35: PRINT BDS
53800 LOCATE 22, 5: PRINT "Yapilan islemler dogru mu ?. (E/H)"
53810 CUG$ = "": WHILE CUG$ = "": CUG$ = INKEY$: WEND
53820 IF CUG$ = "H" OR CUG$ = "h" THEN 53500 ELSE IF CUG$ = "E" OR CUG$
55000 'BOYUT
55010 PCOPY 0, 2
55015 CLS
55020 LOCATE 5, 5: PRINT "BOYUT HESAPLAMA LISTESI"
55030 LOCATE 7, 5: PRINT "O ..GERI DONUS"

```

```

55040 LOCATE 9, 5: PRINT "1 ..IP KAL. GORE BOYUT H."
55050 LOCATE 11, 5: PRINT "2 ..IP KAL. VE BOYUT H."
55060 LOCATE 13, 5: INPUT "SECIMINIZI GIRINIZ :", SEC#
55070 IF SEC# = "" THEN 55015
55080 IF SEC# < "0" OR SEC# > "2" THEN 55015
55090 ON VAL(SEC#) GOTO 55200, 55400
55100 PCOPY 2, 0
55110 LOCATE 20, 1: PRINT SPACE$(25): RETURN
55200 'IP KALINLIGINA GORE BOYUT HESAPLAMASI
55205 CLS
55210 LOCATE 2, 2: PRINT "VERILER"
55220 LOCATE 4, 3: INPUT "Cozgu Kal..", CK
55225 LOCATE 5, 3: INPUT "Ilmelik Kal..", IK
55230 LOCATE 6, 3: INPUT "Atki Kal..", AK
55240 LOCATE 7, 3: INPUT "Ara Atki Sayisi..", AAS
55250 LOCATE 8, 3: PRINT "Pencerenin tum haliya orani."
55260 LOCATE 9, 3: INPUT "Yatay eksende..", y
55270 LOCATE 10, 3: INPUT "Dikey eksende..", d
55280 LOCATE 12, 2: PRINT "SONUCLAR"
55290 LOCATE 13, 3: PRINT "Cizilen kismin eni..":  $(X2 - X1) * (2 * CK +$ 
55300 LOCATE 14, 3: PRINT "Cizilen kismin boyu..":  $(Y2 - Y1) * (AAS * AK$ 
55310 LOCATE 15, 3: PRINT "Kalite"
55320 LOCATE 16, 3: PRINT "Enine dugum s..":  $10 * (X2 - X1) / (X2 - X1) +$ 
55330 LOCATE 17, 3: PRINT "Boyuna dugum s..":  $10 * (Y2 - Y1) / (Y2 - Y1)$ 
55340 LOCATE 18, 3: PRINT "Tum hali eni..":  $(X2 - X1) * (2 * CK + IK) *$ 
55350 LOCATE 19, 3: PRINT "Tum hali boyu..":  $(Y2 - Y1) * (AAS * AK) * d$ 
55360 LOCATE 21, 3: PRINT "Devam etmek icin bir tusa basiniz."
55370 WHILE INKEY# = "": WEND: GOTO 55015
55400 'IP KALINLIKLARI VE BOYUT HESABI
55405 CLS
55410 LOCATE 2, 2: PRINT "VERILER"
55420 LOCATE 4, 3: PRINT "Kalite"
55430 LOCATE 5, 3: INPUT "10cm'deki enine dug. s..", ENS
55440 LOCATE 6, 3: INPUT "10cm'deki boyuna dug. s..", BOS
55450 LOCATE 7, 3: INPUT "Cozgu kal..", CK
55460 LOCATE 8, 3: INPUT "Ara Atki Sayisi..", AAS
55470 LOCATE 9, 3: PRINT "Pencerenin tum haliya orani."

```

```
55480 LOCATE 10, 3: INPUT "Yatay eksende..", y
55490 LOCATE 11, 3: INPUT "Dusey eksende..", d
55500 LOCATE 13, 2: PRINT "SONUCLAR"
55510 LOCATE 14, 3: PRINT "Ip kal..";  $10 / \text{ENS} - 2 * \text{CK}$ 
55520 LOCATE 15, 3: PRINT "Atki kal..";  $10 / (\text{BDS} * \text{AAS})$ 
55530 LOCATE 16, 3: PRINT "Cizilen kismin eni..";  $(\text{X2} - \text{X1}) * (2 * \text{CK} + 1)$ 
55540 LOCATE 17, 3: PRINT "Cizilen kismin boyu..";  $(\text{Y2} - \text{Y1}) * (\text{AAS} * 10)$ 
55550 LOCATE 18, 3: PRINT "Tum hali eni..";  $(\text{X2} - \text{X1}) * (2 * \text{CK} + \text{IK}) * y$ 
55560 LOCATE 19, 3: PRINT "Tum hali boyu..";  $(\text{Y2} - \text{Y1}) * (\text{AAS} * \text{AK}) * d$ 
55570 GOTO 55360
```


EK-B PROGRAM KULLANIM ÖRNEKLERİ

B.1 Yeni Bir Çizime Başlama

Programlanan fonksiyon tuşlarını kullanarak, hareketli bir kursör yardımıyla yeni bir desen oluşturulmaktadır.

işlem : Ana listeden 1 numaralı kod seçilerek yeni bir çizim yapma moduna geçilir. 1 numaralı kod seçildikten sonra desenin depolanacağı kütük isminin girilmesi gerekmektedir.

Birinci ekran:

ANA LİSTE

0. PROGRAMDAN ÇIKIŞ
 1. YENİ BİR ÇİZİME BAŞLAMA
 2. MEVCUT BİR ÇİZİMDE DEĞİŞİKLİK YAPMA
 3. RESMİN SINIRLARINI BULMA
 4. SINIRLARDAN ORJİNAL RESMİ ELDE ETME
 5. SINIRLAR ÜZERİNDE DEĞİŞİKLİK YAPMA
 6. KÜTÜK İŞLEMLERİ
 7. YAZICI (AÇIK/KAPALI)
 8. BOYUT HESAPLAMA
- SEÇİMİNİZİ GİRİNİZ : 1

ikinci ekran:

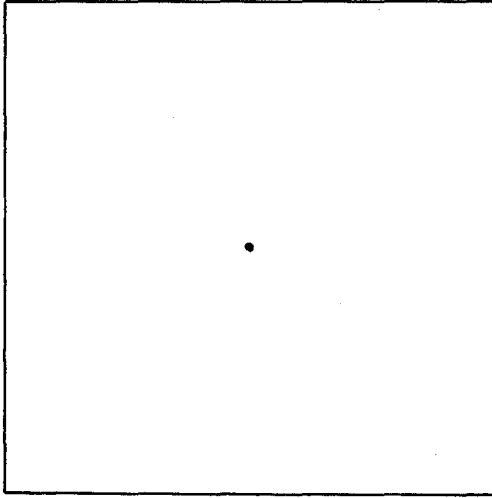
Kütük ismi:

Kütük isminin girilmesinden sonra ekran, çizim yapmak için gerekli birtakım yardımcı fonksiyonları sağlayacak şekilde düzenlenir.

Çizim yapmak için ekranın sol-üst köşesinden başlanarak 128x128 noktadan oluşan bir alan ayrılır. Bu çizim alanının sağında çizimde kullanılacak 16 renk, renk kodlarıyla birlikte sıralanır. Ekranın alt kısmında ise

çizimde kolaylık sağlanması açısından programlanan fonksiyon tuşları ve bunların işlevleri belirtilir. Ayrıca seçilen fonksiyonlarla ilgili birtakım açıklayıcı bilgiler de ekranda yer alır.

Üçüncü ekran:



NO	RENK
0	SIYAH
1	MAVi
2	YEŞİL
3	CAM.G.MAVi
4	KIRMIZI
5	MOR
6	KAHVERENGi
7	BEYAZ
8	GRi
9	PARLAK MAVi
10	PARLAK YEŞİL
11	PARLAK C.G.MAVi
12	PARLAK KIRMIZI
13	PARLAK MOR
14	SARI
15	PARLAK BEYAZ
	RENK:

F1:Çizme F2:Silme F3:Normal F4:Renk F5:Ekran sil.
F6:Pencere F7:Çıkış

Şekil B.1 Ana listeden 1 nolu kod seçildiğinde ekran düzenlemesi.

Çizim alanının tam ortasında hareketli kursör bulunmaktadır. Seçilen fonksiyona göre işlemler bu kursörün hareketiyle sağlanmaktadır.

Fonksiyon tuşları ve işlevleri:

F1:ÇİZME

F1 tuşuna basıldığında program çizim yapma moduna geçer. Seçilmiş olan herhangi bir renk ile hareketli kursörün yardımıyla sağa-sola-yukarı-aşağı tuşları kullanılarak çizim yapması sağlanır. Bu moda geçildiğinde ekrana çıkan mesaj,

ÇİZGİ MODU

olacaktır.

F2:SİLME

Çizilmiş olan bir çizgiyi silme modudur. Çizim işleminde olduğu gibi silme işleminde de fonksiyon, hareketli kursör yardımıyla nokta nokta ilerleyerek gerçekleştirilir. Bu mod seçildiğinde ekrandaki mesaj,

SİLME MODU

olacaktır.

F3:NORMAL

Çizim ya da silme işlemlerinin durdurulacağını belirten moddur. Bu yardımcı komut sayesinde çizim ya da silme modundan çıkılarak, diğer fonksiyonları kullanıp çizim devam edilebilir.

F4:RENK

Çizim sırasında renk seçmek amacıyla kullanılan fonksiyondur. Bu mod seçildiğinde ekrandaki mesaj,

RENK

olacaktır.

Seçilmek istenen rengin renk kodu bu mesajdan sonra girilerek, çizime seçilen bu renk ile devam edilir. Ekran-
da, renklerin numaraları ve isimleri için ayrılan bölgede,
seçilen rengin renk kodunun yanına "*" işareti konur.

F5:EKRAN SİL

Bu modun seçilmesiyle, bir çizim sırasında çizim
alanı tamamen silinerek, ekran yeni bir çizim için kul-
lanılabilir hale getirilir.

F6:PENCERE

Çizim sırasında, istenen bölümleri pencere içine
alarak bu bölümler üzerinde birtakım işlemler yapılması
sağlanır. Bu moda geçmeden önce F3 fonksiyon tuşuyla çizim
ya da silme modundan çıkılması gerekir. Hareketli kursör
yardımıyla önce pencere içine alınması düşünülen bölgenin
uç noktalarından birine gidilir. F6 tuşuna basıldıktan
sonra sağ-sol-aşağı-yukarı tuşlarını kullanarak üzerinde
işlem yapılması düşünülen bölge pencere şeklinde belir-
lenir. Bu fonksiyonun yürütülmesi sırasında ekranda,

PENCERE

mesajı görülür.

Üzerinde işlem yapılması düşünülen bölge seçildikten
sonra ekranda yapılabilecek çeşitli fonksiyonları gösteren
mesaj görülür.

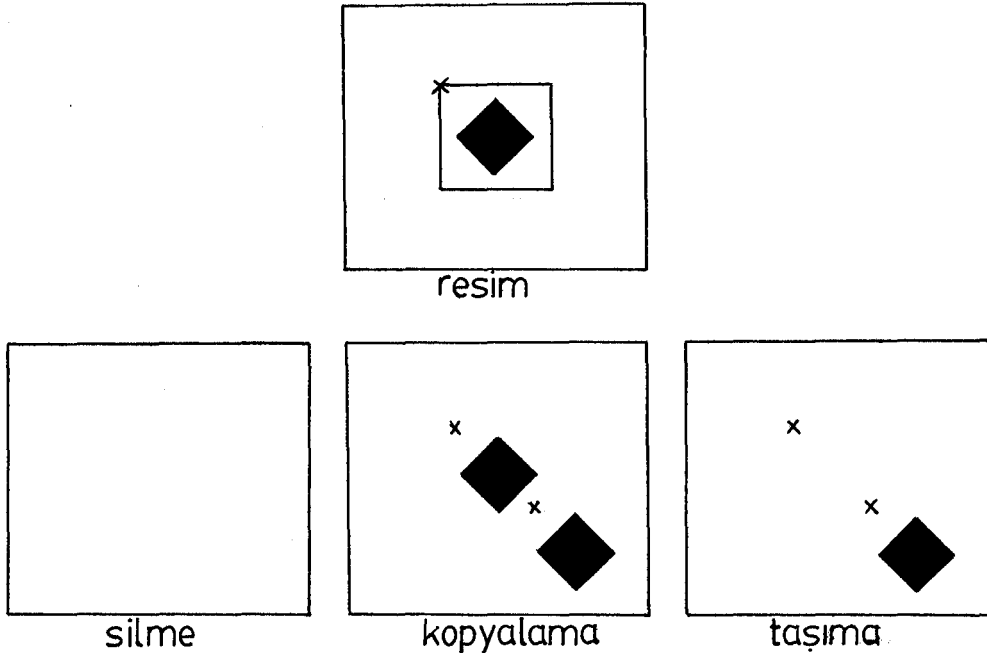
SİLME KOPYALAMA TAŞIMA

SİLME: "S" ya da "s" harfleri girilerek bu mod seçilir. Bu modun fonksiyonu, pencere içine alınmış olan bölgenin resim üzerinden silinmesidir.

KOPYALAMA: "K" ya da "k" harfleriyle bu mod seçilir. Bu mod seçilmeden önce, pencere içine alma işlemine başlanan nokta referans alınarak, bu noktanın taşınması düşünülen yer belirlenir. Daha sonra bu mod seçildiğinde pencere içine alınmış olan bölüm, belirlenen nokta referans alınarak aynen kopya edilir.

TAŞIMA: "T" ya da "t" harfleriyle bu moda geçilir. Pencere içine alınmış olan kısım, belirlenen diğer bir nokta referans alınarak taşınır. Orijinal yerdeki kısım ise silinir. Bu mod seçilmeden önce referans olarak alınacak noktanın belirlenmesi gerekir.

Pencereleme işleminin alt fonksiyonlarıyla, çizim üzerinde, blok olarak silme, kopyalama ve taşıma işlemleri kolaylıkla gerçekleştirilir.



Şekil B.2 Pencere içine alarak silme kopyalama ve taşıma işlemleri

F7:ÇIKIŞ

Çizim tamamlandığında ya da çizim yapma modundan çıkmak istenildiğinde bu mod seçilir. Program ana listeye dönmeden önce, isteğe bağlı olarak çizim otomatik olarak depolanabilir. Kütük ismi olarak, yeni bir çizim yapma modu seçildiğinde girilen isim alınır. Bu mod seçildiğinde ekrandaki mesaj,

DEPOLAMAK İSTİYORMUSUNUZ (E/H)

olacaktır.

E: "E" seçildiğinde, çizim alanı içinde her noktanın renk kodu tek tek "kütükismi.HLD" kütüğüne depolanır.

H: "H" seçildiğinde program doğrudan ana listeye döner.

B.2 Mevcut Bir Çizimde Değişiklik Yapma

Yeni bir çizim yapma modunda, çizilip depolanmış olan desen tekrar çağırılıp, desen üzerinde birtakım değişiklikler ve düzeltmeler yapılabilmektedir.

işlem: Ana listeden 2 numaralı kodun seçilmesiyle bu moda geçilir. 2 numaralı kod seçildikten sonra, üzerinde değiştirme ya da düzeltme yapılacak olan desenin depolandığı kütük isminin girilmesi gerekmektedir ("kütükismi.HLD").

Birinci ekran:

ANA LİSTE

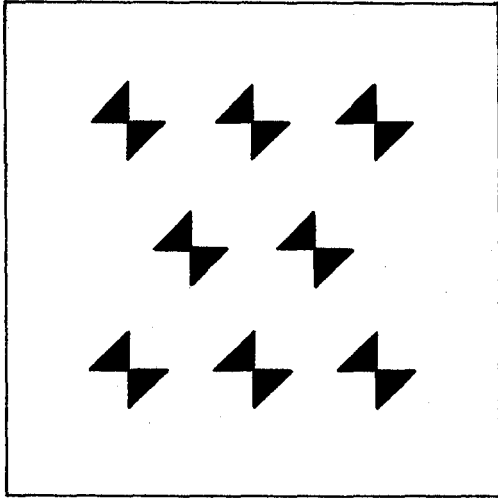
0. PROGRAMDAN ÇIKIŞ
1. YENİ BİR ÇİZİME BAŞLAMA
2. MEVCUT BİR ÇİZİM ÜZERİNDE DEĞİŞİKLİK YAPMA
3. RESMİN SINIRLARINI BULMA

4. SINIRLARDAN ORJİNAL RESMİ ELDE ETME
 5. SINIRLAR ÜZEEİNDE DEĞİŞİKLİK YAPMA
 6. KUTUK İŞLEMLERİ
 7. YAZICI (AÇIK/KAPALI)
 8. BOYUT HESAPLAMA
- SEÇİMİNİZİ GİRİNİZ: 2

İkinci ekran:

Kütük ismi:

Üçüncü ekran:



NO	RENK
0	SİYAH
1	MAVİ
2	YEŞİL
3	CAM.G.MAVİ
4	KIRMIZI
5	MOR
6	KAHVERENGİ
7	BEYAZ
8	GRİ
9	PARLAK MAVİ
10	PARLAK YEŞİL
11	PARLAK C.G.MAVİ
12	PARLAK KIRMIZI
13	PARLAK MOR
14	SARI
15	PARLAK BEYAZ

RENK:

F1:Çizme F2:Silme F3:Normal F4:Renk F5:Ekran sil
F6:Pencere F7:Çıkış

Şekil B.3 Ana listeden 2 nolu kod seçilmesiyle ekran düzenlemesi

Girilen kütük ismindeki desen, çizim için ayrılmış olan alana çizilir. Fonksiyon tuşları ve bunların işlevleri 1 nolu kod seçildiği zamanki ile aynıdır. Hareketli kursör ve yardımcı komutlarla, desen üzerindeki birtakım hatalar düzeltilip bazı değişiklikler yapılabilir.

B.3 Resmin Sınırlarını Bulma

Herhangi bir kütük isminde depolanmış olan bir deseni oluşturan motiflerin sınırları kapalı alanlar şeklinde belirlenip, içleri boşaltılır. Sınır bulma işlemi daha önceki bölümlerde açıklanan bir filtre (Şekil 6.3) yardımıyla gerçekleştirilir. Belirlenen bu sınırlar daha sonra sınır takip etme yöntemi ile (6.1)'deki formülasyonla formül kütüğünde ("kütükismi.FOR") depolanır.

işlem: Bu modun seçilmesi için ana listeden 3 numaralı kodun girilmesi gerekmektedir. 3 numaralı kod seçildikten sonra, sınır bulma işleminin gerçekleştirilmesi istenen desenin depolandığı kütük ismi girilir.

Birinci ekran:

ANA LİSTE

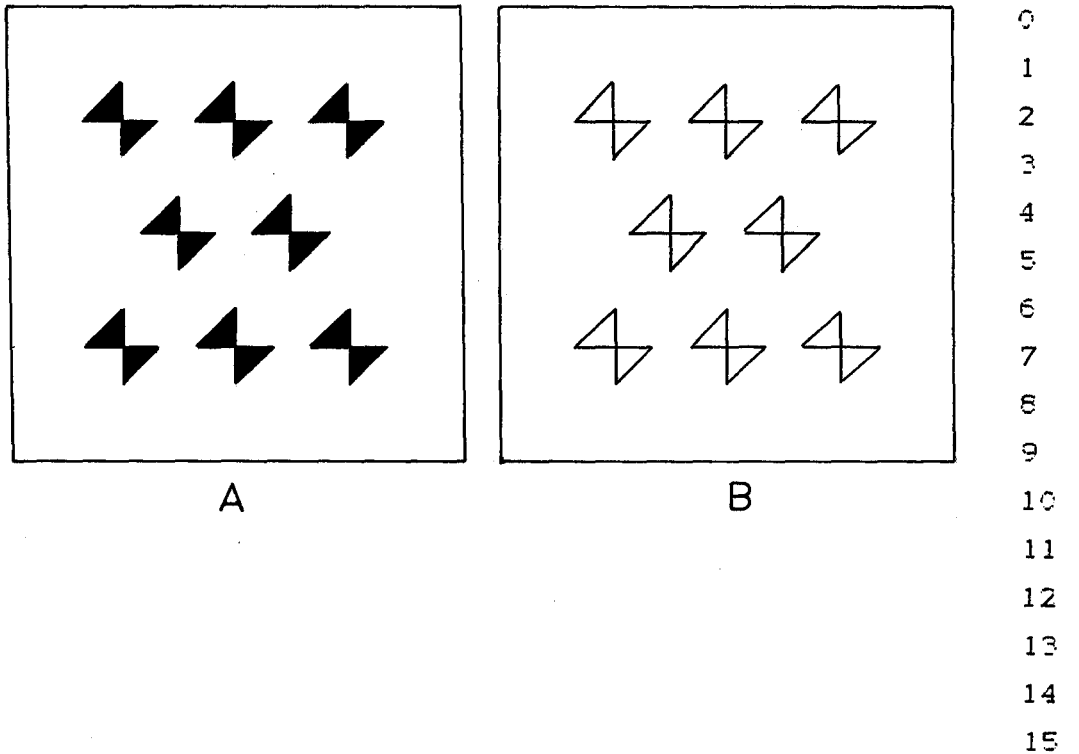
0. PRORAMDAN ÇIKIŞ
 1. YENİ BİR ÇİZİME BAŞLAMA
 2. MEVCUT BİR ÇİZİMDE DEĞİŞİKLİK YAPMA
 3. RESMİN SINIRLARINI BULMA
 4. SINIRLARDAN ORJİNAL RESMİ ELDE ETME
 5. SINIRLAR ÜZERİNDE DEĞİŞİKLİK YAPMA
 6. KUTUK İŞLEMLERİ
 7. YAZICI (AÇIK/KAPALI)
 8. BOYUT HESAPLAMA
- SEÇİMİNİZİ GİRİNİZ: 3

ikinci ekran:

Kütük ismi:

Ekran düzenlemesi ilk iki modun seçilmesinde oluşturulan düzenden farklıdır. 128x128 noktadan oluşan iki çizim alanı vardır. Bunların ilkinde (A) sınırlarının bulunması istenen desen çizilir. İkinci alana (B) ise, birinci alandaki desen bir filtre yardımı ile nokta nokta taranarak sınır olarak bulunan noktalar çizilir. Deseni oluşturan motiflerin sınırlarının bulunması işlemi tamamlandıktan sonra, ekranın ilk kısmındaki desen silinir ve ikinci kısımda sınırlar halinde belirlenmiş olan desen bu bölgeye aktarılır.

Üçüncü ekran:



Şekil B.4 Ana listeden 3 nolu kod seçildiğinde ekran düzenlemesi

(A) çizim alanı üzerinde sınır takip etme yöntemiyle her motif kapalı olan ya da olmayan bir alan şeklinde formül kütüğüne depolanır. Sınır takip etme işleminde, yön ve

uzunluk bilgilerinin hesaplanmasında, incelenen noktanın komşulukları taranmaktadır. Bu nedenle takip etme işlemine başlanan noktaya ulaşıldığının ya da takip edilen renkteki sınırın sona erdiğinin anlaşılabilmesi için, incelenen noktanın komşulukları taranıp yeni yön bulunduğu o nokta ekranda siyah olarak görülür. Böylece sınır takip etme olayı ekran üzerinde rahatlıkla izlenebilir.

Bir deseni oluşturan motiflerin, yön ve uzunluk bilgileri ile ifade edilerek formül kütüğüne depolanmasından sonra program otomatik olarak ana listeye döner. Böylece desenin orijinali "kütükismi.HLD" kütüğünde, sınırlarla belirlenmiş hali ise "kütükismi.FOR" kütüğünde saklanır.

B.4 Sınırlardan Orijinal Resmî Elde Etme

Verilen bir kütük ismiyle formül kütüğünde depolanmış olan desen ekrana çizdirilmekte ve isteğe bağlı olarak, sadece sınırları belirlenmiş olan bir desenden, orijinal renkli haline geçilebilmektedir.

işlem: Programın bu modda çalışması için ana listeden 4 numaralı kod seçilir. Bu kodun seçilmesinden sonra, girilen kütük ismiyle, o isimde formül kütüğünde bulunan desen (sadece sınırlar halinde belirlenmiş) ekrana çizdirilir.

Birinci ekran:

ANA LİSTE

0. PROGRAMDAN ÇIKIŞ
1. YENİ BİR ÇİZİME BAŞLAMA
2. MEVCUT BİR ÇİZİMDE DEĞİŞİKLİK YAPMA
3. RESMİN SINIRLARINI BULMA
4. SINIRLARDAN ORJİNAL RESMİ ELDE ETME
5. SINIRLAR ÜZERİNDE DEĞİŞİKLİK YAPMA
6. KUTUK İŞLEMLERİ

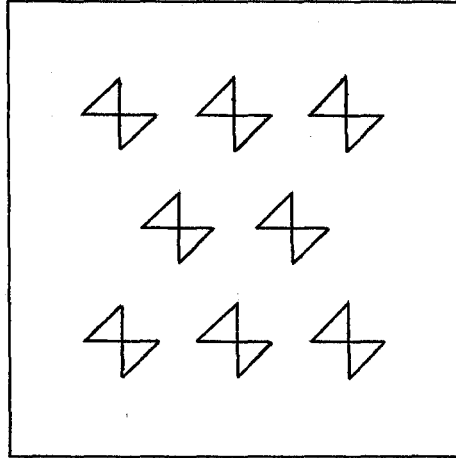
7. YAZICI (AÇIK/KAPALI)
 8. BOYUT HESAPLAMA
 SEÇİMİNİZİ GİRİNİZ: 4

ikinci ekran:

Kütük ismi:

Kütük isminin girilmesinden sonra, motifleri ifade eden, uzunluklarıyla birlikte yönleri de içeren bilgiler kütükten teker teker alınır. Bu bilgiler (6.1)'deki formülasyonla kütük içine depolanmış olduğundan, önceki bölümlerde anlatılmış olan (6.2) ve (6.3)'deki formüllerle bu bilgiler çözümlenerek yönler ve bu yönlerdeki uzunluklar hesaplanır. Hesaplanan bu değerler, "BASIC" programlama dilindeki "DRAW" komutu ile kullanılabilecek şekilde düzenlenir. Ekran düzenlemesi sadece 128x128 noktadan oluşan bir çizim alanı şeklindedir.

Üçüncü ekran:



Şekil B.5 Ana listeden 4 nolu kod seçildiğinde ekran düzenlemesi

Kütükten alınan bilgilerle desen ekranda ayrılan alana çizildikten sonra ekrandaki mesaj,

RESMİ DEPOLAMAK İSTİYORMUSUNZ (E/H)

olacaktır.

E: "E" seçildiğinde, çizim alanı içindeki desen, aynı ya da farklı bir kütük ismi ile "Kütükismi.HLD" kütüğüne depolanır. "E" seçimi ile ekrandaki yeni mesaj,

Kütük ismi:

olacaktır.

H: "H" seçildiğinde, işlem yapılmadan bir sonraki mesaja geçilir.

"E" seçilip girilen kütük ismindeki desen depolandıktan sonra ya da "H" seçilip hiç bir işlem yapılmadığında ekrandaki mesaj,

RESMİ DOLDURMAK İSTİYORMUSUNUZ (E/H)

olacaktır.

H: "H" seçimi ile program ana listeye geri döner.

E: "E" seçimi ile sınırlar halinde belirlenmiş olan desende, aynı renkte iki sınır arasındaki bölge sınır rengi ile doldurularak orijinal renkli resim elde edilebilir. Bu işlemi takip eden mesaj,

RESMİ DEPOLAMAK İSTİYORMUSUNUZ (E/H)

olacaktır.

E: "E" seçildiğinde, ana listeden 4 nolu kod seçildiğinde girilen kütük ismi referans alınarak resim "Kütükismi.HLD" kütüğüne depolanır.

H: "H" seçildiğinde program ana listeye döner.

B.5 Sınırlar Üzerinde Değişiklik Yapma

Bir deseni oluşturan motiflerin sınırları belirlenip bir formül kütüğünde depolandıktan sonra, bu mod seçil-

diğinde, ekranda ayrılan ilk çizim alanında, kütükten alınan bilgilere göre desen çizdirilmektedir. İkinci çizim alanında ise, kütüğe alış sırasına göre, her motifin sınırları tek tek çizdirilerek, seçilen bir motif üzerinde istenen değişiklikler yapılmaktadır. Bu değişikliklerden sonra elde edilen yeni motif, istenirse eskisi silinerek orijinal desen üzerine yerleştirilmektedir.

işlem: Ana listeden 5 numaralı kod seçilerek program bu modda çalışmaya başlar. 5 numaralı kod seçildikten sonra, sınırlarının ekrana çizilmesi istenen desenin depolandığı kütük isminin girilmesi gerekmektedir.

Birinci ekran:

ANA LİSTE

0. PROGRAMDAN ÇIKIŞ
1. YENİ BİR ÇİZİME BAŞLAMA
2. MEVCUT BİR ÇİZİMDE DEĞİŞİKLİK YAPMA
3. RESMİN SINIRLARINI BULMA
4. SINIRLARDAN ORJİNAL RESMİ ELDE ETME
5. SINIRLAR ÜZERİNDE DEĞİŞİKLİK YAPMA
6. KÜTÜK İŞLEMLERİ
7. YAZICI (AÇIK/KAPALI)
8. BOYUT HESAPLAMA

SEÇİMİNİZİ GİRİNİZ: 5

İkinci ekran:

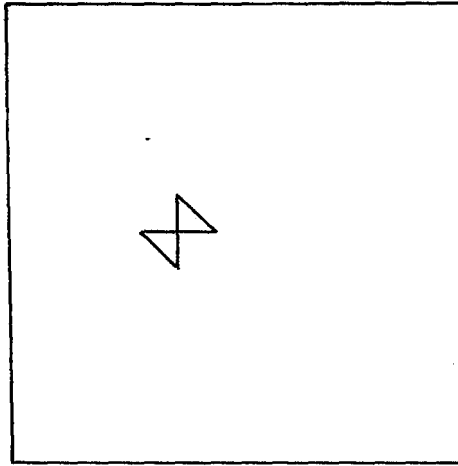
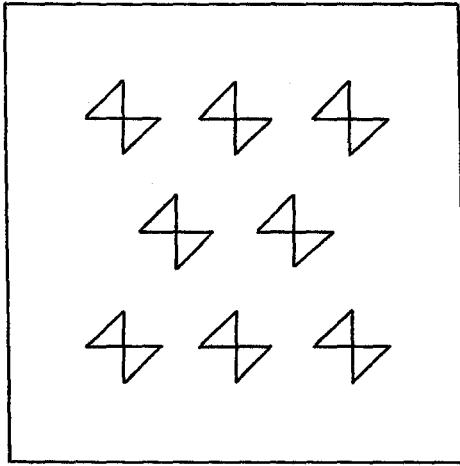
Kütük ismi:

Üçüncü ekran:

LÜTFEN BEKLEYİN

Kütük ismi girildiğinde, bu isimde formül kütüğünde bulunan sınır bilgileri tek tek alınır. Bu bilgiler alınırken aynı zamanda her motife ait olan veriler bir vektör içine atılır. Her motifin farklı bir vektör içine

Dördüncü ekran:



RENK

0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

X:40 Y:64 ÖLÇEK:1 AÇI:0 RENK:4

X:40 Y:64 ÖLÇEK:1 AÇI:90 RENK:4

F1:RENK+ F2:RENK- F3:ÖLÇEK+ F4:ÖLÇEK- F5:DÖNME+
F6:DÖNME- F7:SEÇİM F8:KABUL F9:SİLME F0:ÇIKIŞ

Şekil B.6 Ana listeden 5 nolu kod seçildiğinde
ekran düzenlemesi.

atılmasından sonra çizim işlemine geçilir. Ekranda 128x128 noktadan oluşan iki çizim alanı ayrılır. İlk çizim alanında, her vektördeki, motifi tanımlayan bilgiler "DRAW" komutu ile kullanılacak şekilde düzenlenerek desen çizilir. İkinci çizim alanında ise yukarı (↑) ve aşağı (↓) tuşları kullanılarak, kütüğe depolanış sırasına göre her motif tek tek çizdirilir. Üzerinde değişiklik yapılacak motife bu şekilde ulaşıldıktan sonra programlanan fonksiyon tuşlarıyla çeşitli değişiklikler yapılabilir. Yukarı (↑) tuşu kullanılarak vektör numarası arttırılır. Aşağı (↓) tuşu kullanılarak vektör numarası azaltılır.

Bu mod seçildiğinde ekran düzenlemesi Şekil E.6' da görülmektedir

Fonksiyon tuşlarının kullanımı:

F1:RENK+

Üzerinde değişiklik yapma amacıyla seçilmiş olan motifin renk kodu referans alınarak, bu tuşa her basılışta motifin renk kodu bir arttırılır. Örneğin; incelenen motifin renk kodu "8" olduğunda, F7 tuşunun her kullanılışında renk kodunun değeri bir arttırılarak 9,10,11,... olur. Böylece motifi ifade eden vektördeki renk parametreside değiştirilerek, her seferinde motif yeniden ikinci çizim alanına çizilir.

F2:RENK-

Bu tuşun her kullanımında, üzerinde değişiklik yapılacak olan motifin renk kodu referans alınarak, renk kodunun değeri bir azaltılır. Örneğin; motifin orijinal renk kodu "8" ise bu tuşa her basılışta renk kodu değeri 7,6,5,... olur. Değiştirilen renk kodları ile birlikte, motif her seferinde yeniden çizilmektedir.

F3:ÖLÇEK+

Ölçek faktörünün büyütülmesi için kullanılan fonksiyon tuşudur. Motifin orijinal ölçek faktörü referans alınarak, bu tuşa her basılışta ölçek faktörü üst sınır değeri olan 250'ye kadar birer birer arttırılır. Ölçek faktörünün her değişiminde motifin atıldığı vektördeki, ölçek parametresi değiştirilerek motif ikinci çizim alanına çizilir.

F4:ÖLÇEK-

Ölçek faktörünün küçültülmesi için kullanılan fonksiyon tuşudur. Motifin orijinal ölçek faktörü referans alınarak, bu tuşa her basılışta ölçek faktörü alt sınır değeri olan 1'e kadar birer birer azaltılır. Ölçek faktörünün aldığı her yeni değerden sonra motifin şekli ikinci çizim alanından göçlenir.

F5:DÖNME+

Motifin saatin tersi yönünde döndürülmesini sağlayan tuştur. Bu fonksiyon tuşuna her basılışta motif saatin tersi yönünde, orijinal dönme açısı referans alınarak 45° döner. Bu değişim motifin atıldığı vektörde, dönme açısı parametresinin de değişmesini sağlar. Böylece "DRAW" komutu içinde dönme açısını belirleyen değer de değişerek, motif kendi merkezi etrafında dönmüş olarak çizilir.

F6:DÖNME-

Motifin merkezi etrafında saat yönünde dönmesini sağlayan tuştur. Bu fonksiyon tuşuna her basılışta motif, orijinal dönme açısı referans alınarak saat yönünde 45° döner. Bu parametrenin her değişiminde, motifin aldığı yeni şekil ikinci çizim alanında görülür.

F7:SEÇİM

Yukarı (↑) ve aşağı (↓) tuşları kullanılarak, vektöre atılış sırasına göre tek tek çizdirilen motiflerden, üzerinde değişiklik yapılmak üzere bir motif seçmek için kullanılan tuştur. Bu tuşun kullanımından sonra yukarı (↑) ve aşağı (↓) tuşlarının fonksiyonları değiştirilerek, motifin başlangıç koordinatlarını değiştirmek amacıyla kullanılır.

F8:KABUL

Diğer fonksiyon tuşlarını kullanarak, seçilen motif üzerinde istenen tüm değişiklikler yapıldıktan sonra, bu tuşa basılarak, motif ilk çizim alanına yerleştirilir. Motifin yeni şekliyle orijinal desen içine yerleştirilmesinden önceki mesaj,

DEĞİŞİKLİK İPTALİ (E/H)

olacaktır.

E: "E" seçimi ile desen üzerinde yapılan tüm değişiklikler iptal edilerek, işlemlere kalındığı yerden devam edilir.

H: "H" seçildiğinde, motif üzerinde yapılan tüm değişiklikler, eskisi silinerek orijinal desen üzerine aktarılır. Böylece o desene ait eski değerler, motifin atıldığı vektörden silinerek, yeni değerler aynı vektöre atılır. Bu işlemin ardından ikinci çizim alanı tamamen silinerek, aynı işlemlerin yeniden tekrarlanabilmesi için uygun bir hale getirilir.

F9:SİLME

Üzerinde değişiklik yapılmak üzere seçilmiş olan motifin silinmesi için kullanılan tuştur. Böylece o motife ait tüm bilgiler, motifin saklı olduğu vektörden silinir.

F0:ÇIKIŞ

Programın ana listeye dönmesi için kullanılan tuştur.

F0 tuşuna basıldığında ekrandaki mesaj,

RESİM DEPOLAMAK İSTİYORMUSUNUZ (E/H)

E: "E" seçildiğinde, bazı motifleri üzerinde değişiklik yapılmış olan desen, eskisi silinerek, aynı kütük ismi ile formül kütüğüne depolanır ve ana listeye dönlür.

H: "H" seçildiğinde, üzerinde değişiklik yapılmış olan desen herhangi bir kütük isminde depolanmadan ana listeye dönlür.

Çizim alanı olarak ayrılmış kısımların altında, çizim me ait bazı parametrelerin yazımı için ayrılmış bir bölge vardır. İlk satırdaki parametre değerleri, üzerinde değişiklik yapmak üzere seçilmiş olan motifin orijinal parametreleridir. İkinci satırdaki parametre değerleri ise, ikinci çizim alanında birtakım değişiklikler yapıldıkça sürekli değişerek, motifin yeni parametre değerlerini gösterir. "F7:SEÇİM" tuşunun kullanımından sonra yukarı (↑)-aşağı (↓)-sağa (→)-sola (←) tuşları, seçilen motifin başlangıç koordinatlarını değiştirmek için kullanılır.

B.6 Kütük işlemleri

Kütüklerle ilgili birtakım yardımcı işlemler gerçekleştirilmektedir.

işlem: programın bu modda çalışması için ana listeden 6 numaralı kodun seçilmesi gerekmektedir.

Brinci ekran:

ANA LİSTE

0. PROGRAMDAN ÇIKIŞ
1. YENİ BİR ÇİZİME BAŞLAMA

2. MEVCUT BİR ÇİZİMDE DEĞİŞİKLİK YAPMA
 3. RESMİN SINIRLARINI BULMA
 4. SINIRLARDAN ORJİNAL RESMİ ELDE ETME
 5. SINIRLAR ÜZERİNDE DEĞİŞİKLİK YAPMA
 6. KUTUK İŞLEMLERİ
 7. YAZICI (AÇIK/KAPALI)
 8. BOYUT HESAPLAMA
- SEÇİMİNİZİ GİRİNİZ: 6

6 numaralı kodun seçilmesinden sonra program, kütüklerle ilgili birtakım yardımcı işlemlerin gerçekleştirildiği alt listeye geçer.

ikinci ekran:

KUTUK İŞLEMLERİ LİSTESİ

0. ANA LİSTEYE DÖNÜŞ
1. ÇİZİM KUTUKLERİNİN LİSTESİ
2. ÇİZGİ FORMUNDAKİ KUTUKLERİN LİSTESİ
3. KUTUK İSMİNİ DEĞİŞTİRME
4. KUTUK SİLME

SEÇİMİNİZİ GİRİNİZ:

0: "0" seçildiğinde program bu alt listeden çıkarak ana listeye döner.

1: "1" seçildiğinde program bütün çizim kütüklerinin listesini verir (kütük ismi ve tipiyle birlikte).

2: "2" seçildiğinde program, sınırları belirlenip formülüne edilmiş ve bu şekilde bir formül kütüğüne depolanmış olan çizimlerin listesini verir.

3: "3" seçildiğinde program kütük isimlerinin değiştirilebilmesine olanak sağlar. Bu mod seçildikten sonra çıkan ilk mesaj,

ESKİ kütükismi:

olacaktır.

Değiştirilmek istenen kütük isminin girilmesinden sonraki ikinci mesaj,

YENİ kütükismi:

olacaktır.

Bu mesajdan sonra kütüğün alması düşünülen yeni ismi girilir.

4: "4" seçildiğinde ise, istenmeyen kütükler, kütük ismi girilerek silinebilir. Bu modun seçilmesinden sonra ekranda görülen mesaj,

Kütükismi:

olacaktır.

Bu mesajdan sonra silinmesi düşünülen kütüğün ismi girilir.

B.7 Yazıcı (Açık/Kapalı)

Program herhangi bir modda çalışırken, çizim alanı olarak ayrılan bölgedeki desen yazıcıdan elde edilebilmektedir. Renkli yazıcı kullanılmadığından ve desenler renkli olduğundan, yazıcıdan desen renklerinin ayırt edilebilmesi için her renk 8x8'lik bir nokta matrisi içinde farklı şekillerle tanımlanmaktadır.

işlem: Programın bu modda çalışabilmesi için ana listeden 7 numaralı kodun seçilmesi gerekmektedir.

Birinci ekran:

ANA LİSTE

0. PROGRAMDAN ÇIKIŞ
 1. YENİ BİR ÇİZİME BAŞLAMA
 2. MEVCUT BİR ÇİZİMDE DEĞİŞİKLİK YAPMA
 3. RESMİN SINIRLARINI BULMA
 4. SINIRLARDAN ORJİNAL RESMİ ELDE ETME
 5. SINIRLAR ÜZERİNDE DEĞİŞİKLİK YAPMA
 6. KUTUK İŞLEMLERİ
 7. YAZICI (AÇIK/KAPALI)
 8. BOYUT HESAPLAMA
- SEÇİMİNİZİ GİRİNİZ:7

7 numaralı kodun seçilmesinden sonraki mesaj,

YAZICI MODU...

yazıcı modunun herhangi bir değişiklik yapılmadan önceki durumunu gösterir ("AÇIK" ya da "KAPALI").

"AÇIK" olması yazıcıdan çıktı alınabileceğini,"KAPALI" olması ise çıktı alınamayacağını ifade eder.

Bu mesajın hemen ardından görülen ikinci mesaj,

BU MODU DEĞİŞTİRMEK İSTİYORMUSUNUZ ? E/H

E: "E" seçildiğinde yazıcı modu "AÇIK" ise "KAPALI" durumuna, "KAPALI" ise "AÇIK" durumuna getirilir.

H: "H" seçildiğinde yazıcı modunun bulunduğu komanda bir değişiklik yapılmaz.

Desen çıktısının yazıcıdan alınabilmesi için yazıcı modu "AÇIK" durumuna getirilir ve ALT-P tuşu, ekrandaki deseni yazıcıya aktarma fonksiyonunu gerçekleştirecek şekilde programlanır.

7 numaralı modda isteğe göre yazıcı modu ve ALT-P tuşu "AÇIK" konumuna getirildikten sonra program ana listeye döner.

Program diğer modlarda çalışmaya devam ederken herhangi bir anda desen çıktısı alınmak istendiğinde ALT-P tuşuna basılır. Bu tuşa basıldığında ekrandaki mesaj,

YAZICI HAZIR OLDUĞUNDA HERHANGİ BİR TUŞA BASINIZ

olacaktır.

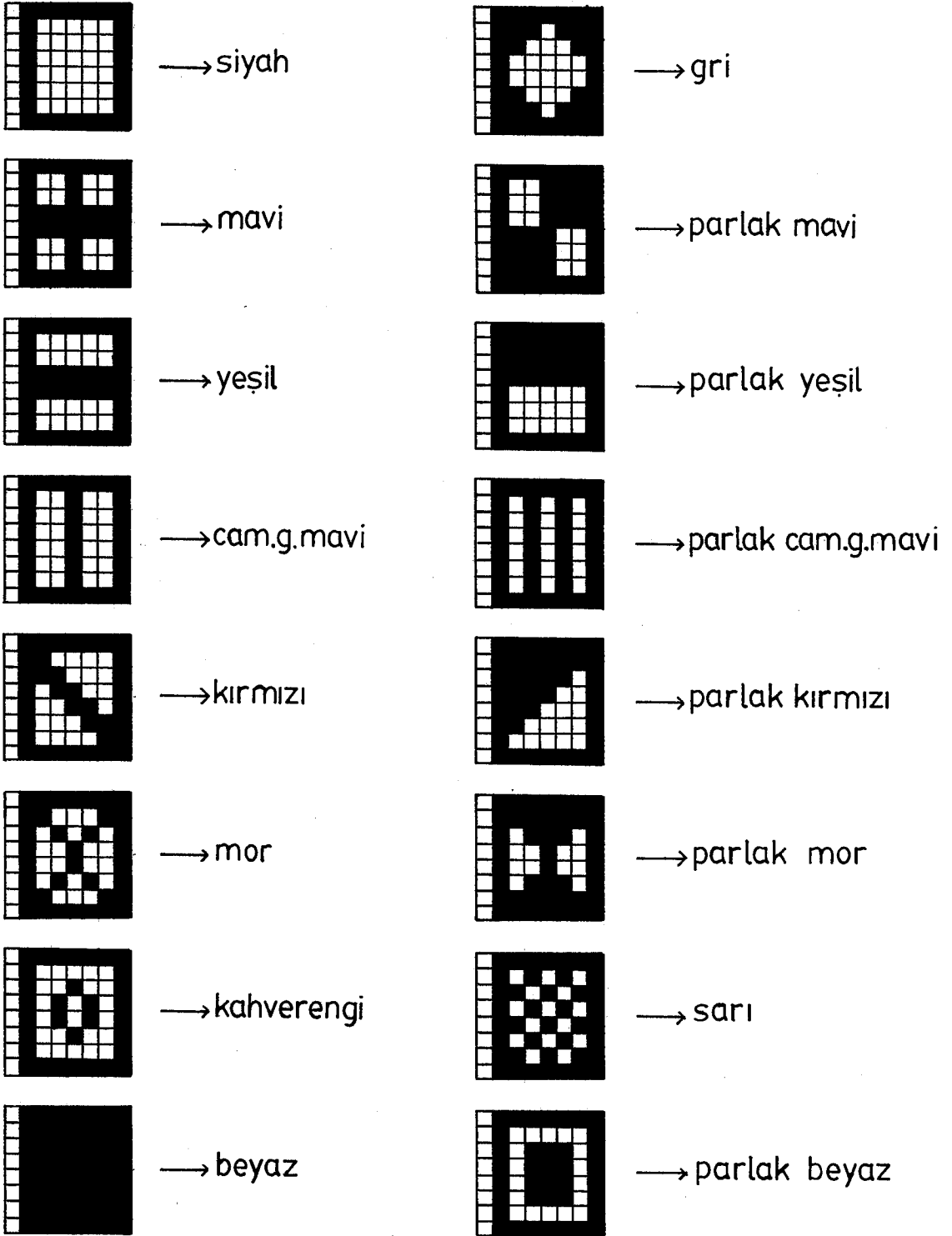
Yazıcı hazırlandıktan sonra herhangi bir tuşa basılarak, işlemlerin devam etmesi sağlanır. Bu anda ekrandaki mesaj,

LÜTFEN BEKLEYİN

olmaktadır.

Bu arada yazıcı programlanır. 128x128 noktadan oluşan çizim alanı dörde bölünür ve bir desen dört bölüm halinde yazıcıdan elde edilir. İlk olarak sol-üst köşeden başlanarak 64x64 noktadan oluşan alan satır satır taranır. Bir satırdaki 64 noktanın renk kodları belirlenir. Her renk için oluşturulmuş olan 8x8'lik renk tanımlama matrisine göre, desendeki renkler değişik şekiller halinde yazıcıdan elde edilir. Böylece dört bölgedeki noktalar sıra ile taranarak tüm desen yazıcıdan çıkarılmış olmaktadır. Renkler ve renkleri ifade eden şekiller şekil B.7'de gösterilmektedir.

Yazıcıdan, her renk kendisini ifade eden 8x8'lik bir matrisle simgelenir. Daha özel bir kullanım amacına göre, eğer resim bir halı deseni şeklinde ise yazıcıdan elde edilen çıktılarla ip renkleri ve düğüm sayıları kolaylıkla görünerek dokuma işlemi gerçekleştirilebilir.



E.8 Boyut Hesaplama

Sipariş ebatı, kalitesi ve çözümlü kalınlığı belli olan bir halının bu özelliklerde gerçekleştirilmesi için gerekli ilmelik ve atkı kalınlıkları ile halının eninde ve boyunda olması gereken düğüm sayıları hesaplanmaktadır. Ayrıca ip kalınlıkları belli olan bir halının dokunduktan sonra kalitesi ve toplam düğüm sayılarının hesabı da yazılım tarafından gerçekleştirilmektedir.

işlem : Ana listeden 8 numaralı kod girilerek programın bu modda çalışması sağlanır. 8 numaralı kodun seçilmesinden sonra program boyut hesaplama alt listesine geçmektedir.

Birinci ekran:

ANA LİSTE

0. PROGRAMDAN ÇIKIŞ
 1. YENİ BİR ÇİZİME BAŞLAMA
 2. MEVCUT BİR ÇİZİMDE DEĞİŞİKLİK YAPMA
 3. RESMİN SINIRLARINI BULMA
 4. SINIRLARDAN ORJİNAL RESMİ ELDE ETME
 5. SINIRLAR ÜZERİNDE DEĞİŞİKLİK YAPMA
 6. KUTUK İŞLEMLERİ
 7. YAZICI (AÇIK KAPALI)
 8. BOYUT HESAPLAMA
- SEÇİMİNİZİ GİRİNİZ:8

ikinci ekran:

BOYUT HESAPLAMA LİSTESİ

0. ANA LİSTEYE DÖNÜŞ
 1. İP KALINLIĞI HESABI
 2. İP KALINLIĞINA GÖRE KALİTE HESABI
- SEÇİMİNİZİ GİRİNİZ:

0: "0" seçildiğinde program hiç bir işlem yapmadan ana listeye geri döner.

1: "1" seçildiğinde program verilen belli parametrelere göre gerekli olan ilmelik ve atkı ipi kalınlığının belirlenmesi moduna geçer.

Üçüncü ekran:

VERİLER:

HALININ KALİTESİ

10cm'deki enine düğüm sayısı :
 10cm'deki boyuna düğüm sayısı :
 Çözgü kalınlığı (mm) :
 İki düğüm arasındaki ara atkısı sayısı :
 Sipariş ebatı
 En (cm) :
 Boy (cm) :

SONUÇLAR:

Halının enine düğüm sayısı :
 Halının boyuna düğüm sayısı :
 ilmelik kalınlığı (mm) :
 Atkı ipi kalınlığı (mm) :

"VERİLER" kısmında belirtilen değerlere göre, bölüm 6.2.6'daki yaklaşımların ışığında, "SONUÇLAR" kısmındaki değerler hesaplanmaktadır. Hesaplanan değerler "SONUÇLAR" bölümünde gösterildikten sonra ekrandaki mesaj,

DEVAM ETMEK İÇİN HERHANGİ BİR TUŞA BASINIZ

olmaktadır. Böylece program Boyut Hesaplama Listesine geri döner.

2: "2" seçildiğinde program, belirlenen ip kalınlıklarına göre halı kalitesinin ne olacağını hesaplanması moduna geçer.

Dördüncü ekran:

VERİLER:

Çözgü ipi kalınlığı(mm) :
Atkı ipi kalınlığı(mm) :
İlmelik ipi kalınlığı(mm) :
İki düğüm arasındaki ara atkısı sayısı :
Sipariş ebatı
En(cm) :
Boy(cm) :

SONUÇLAR:

Halının enine düğüm sayısı :
Halının boyuna düğüm sayısı :
HALININ KALİTESİ
10cm'deki enine düğüm sayısı :
10cm'deki boyuna düğüm sayısı :

"VERİLER" kısmında belirlenen parametrelere göre (6.3-4)'deki formülasyonlarla "SONUÇLAR" bölümündeki değerler hesaplanmaktadır. Bu değerlerin "SONUÇLAR" kısmında gösterilmesinden sonra ekrandaki mesaj,

DEVAM ETMEK İÇİN HERHANGİ BİR TUŞA BASINIZ

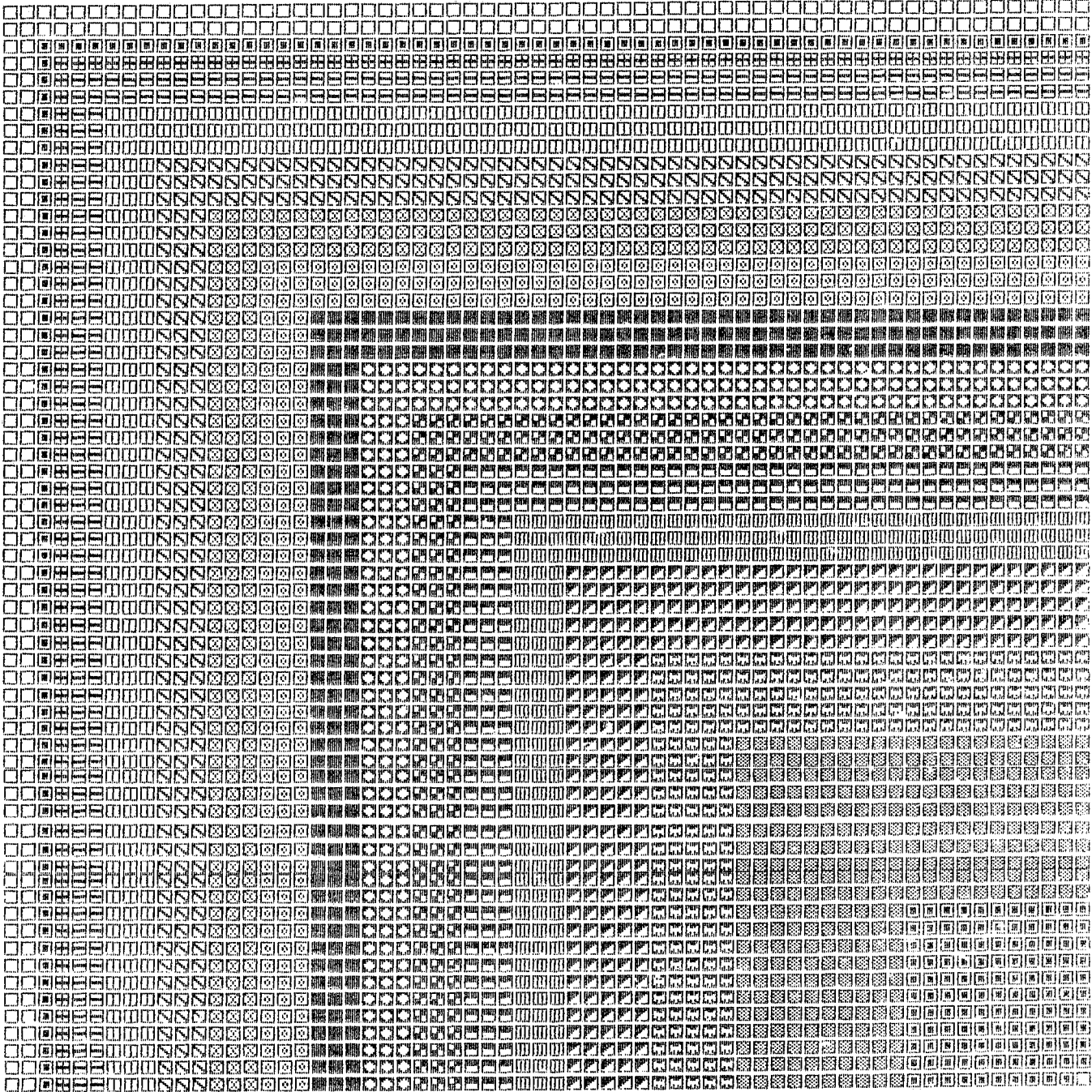
olmaktadır. Böylece program Boyut Hesaplama Listesine geri döner.

Ayrıca EK'B'deki, "Yeni Bir Çizime Başlama" ya da "Mevcut Bir Çizimde Değişiklik Yapma" bölümlerinde anlatılan pencere içine alma işlemindeki Silme, Taşıma ve Kopyalama fonksiyonlarına ek olarak Boyutlandırma kısmı da eklenmiştir. Böylece çizim esnasında da çizilen kısım pencere içine alınarak belirlenen çözgü, atkı, ip kalınlıklarına göre halının kalitesi ve çizilen kısmın ebatları hesaplanabilmektedir. Ayrıca çizilen kısmın tüm haliya olan oranı belirlendikten sonra halının dokuma işleminden sonraki boyutunun tamamının ne olacağı da bulunmaktadır. Boyut hesaplamada ikinci bir işlem olarak da halının kalitesi ve çözgü ipi kalınlığı belirlendiğinde ilmelik ve atkı kalınlıklarının ne olması gerektiği program tarafından kullanıcıya verilmektedir. İkinci işlem içinde de, pencere içine alınmış olan desenin tüm haliya olan oranı belirlendiğinde halının dokunduktan sonraki boyu ve eninin ne olacağıda hesaplanmaktadır.

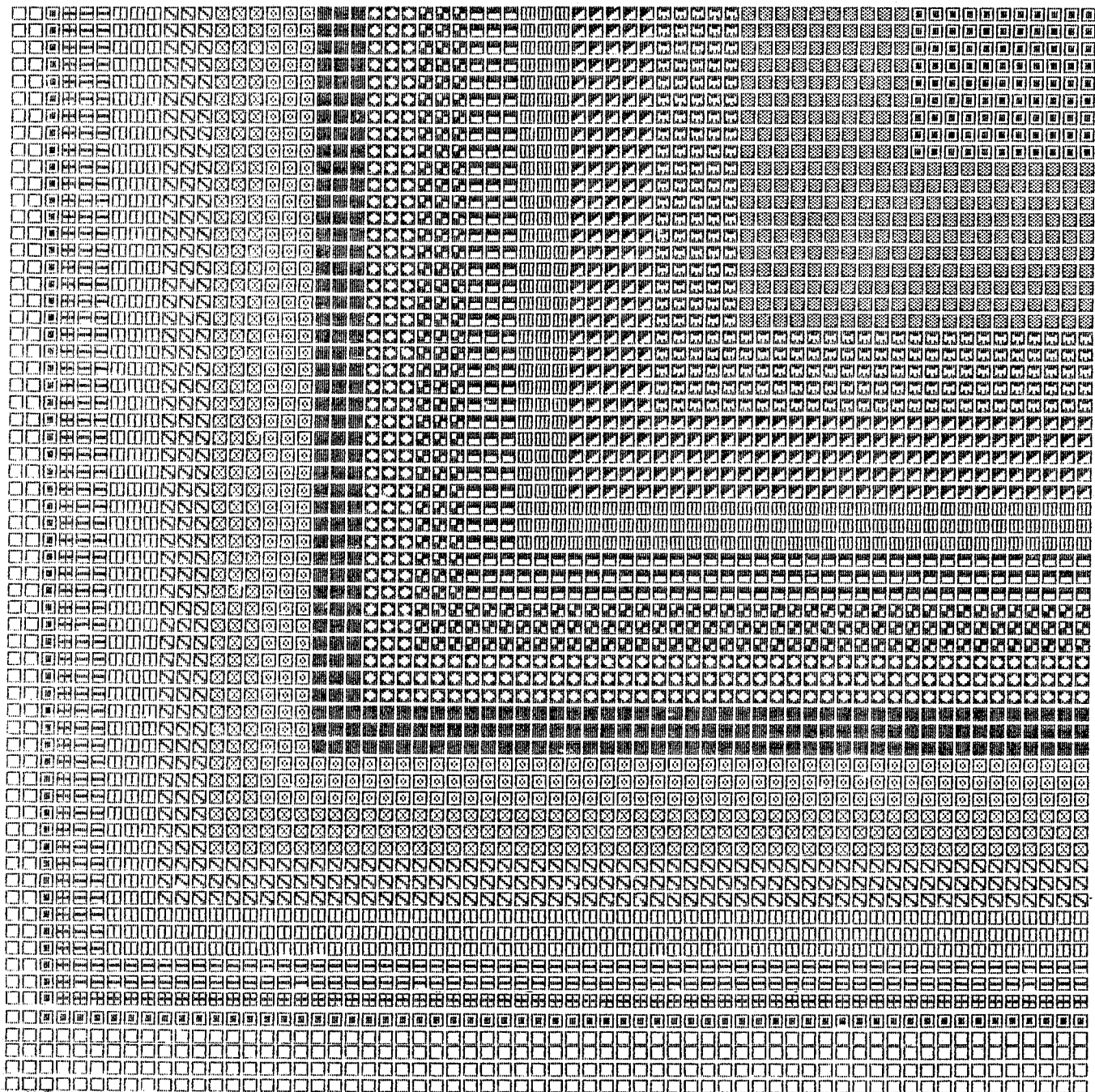
EK-C PROGRAM ÇIKTILARI

KÜTÜKİSMİ : DESEN 01

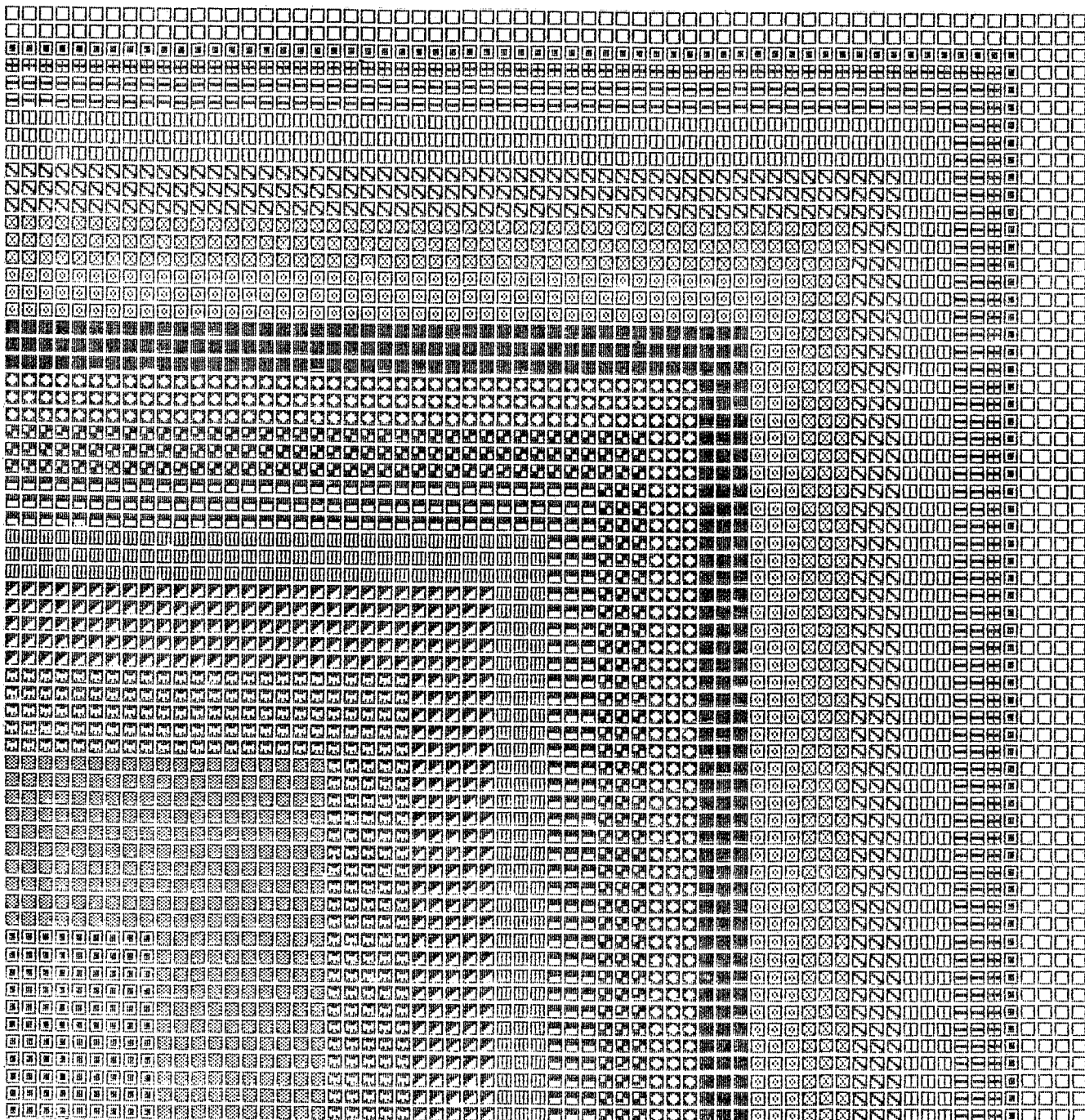
BÖLÜM 1



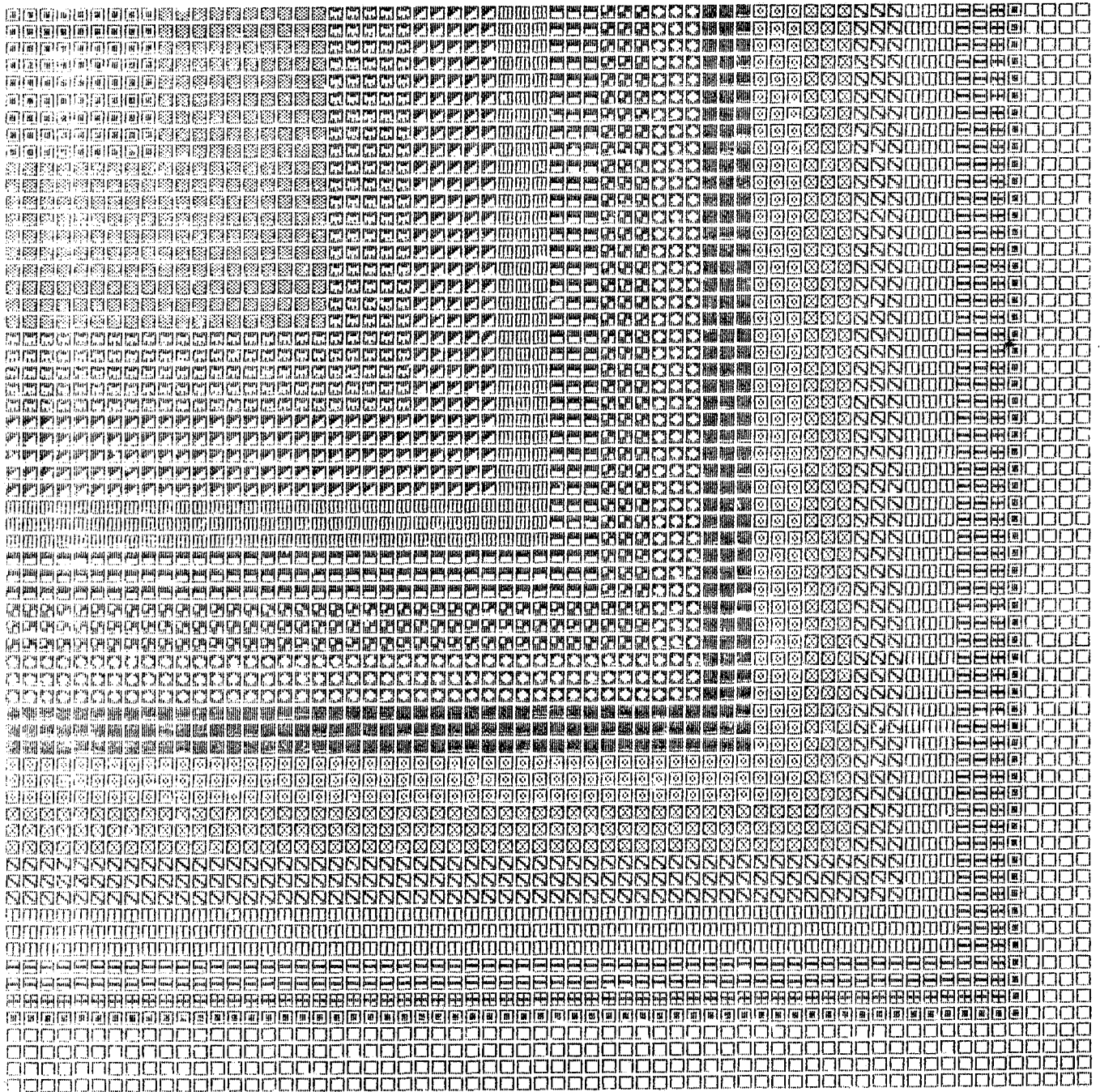
BÖLÜM 2



BÖLÜM 3

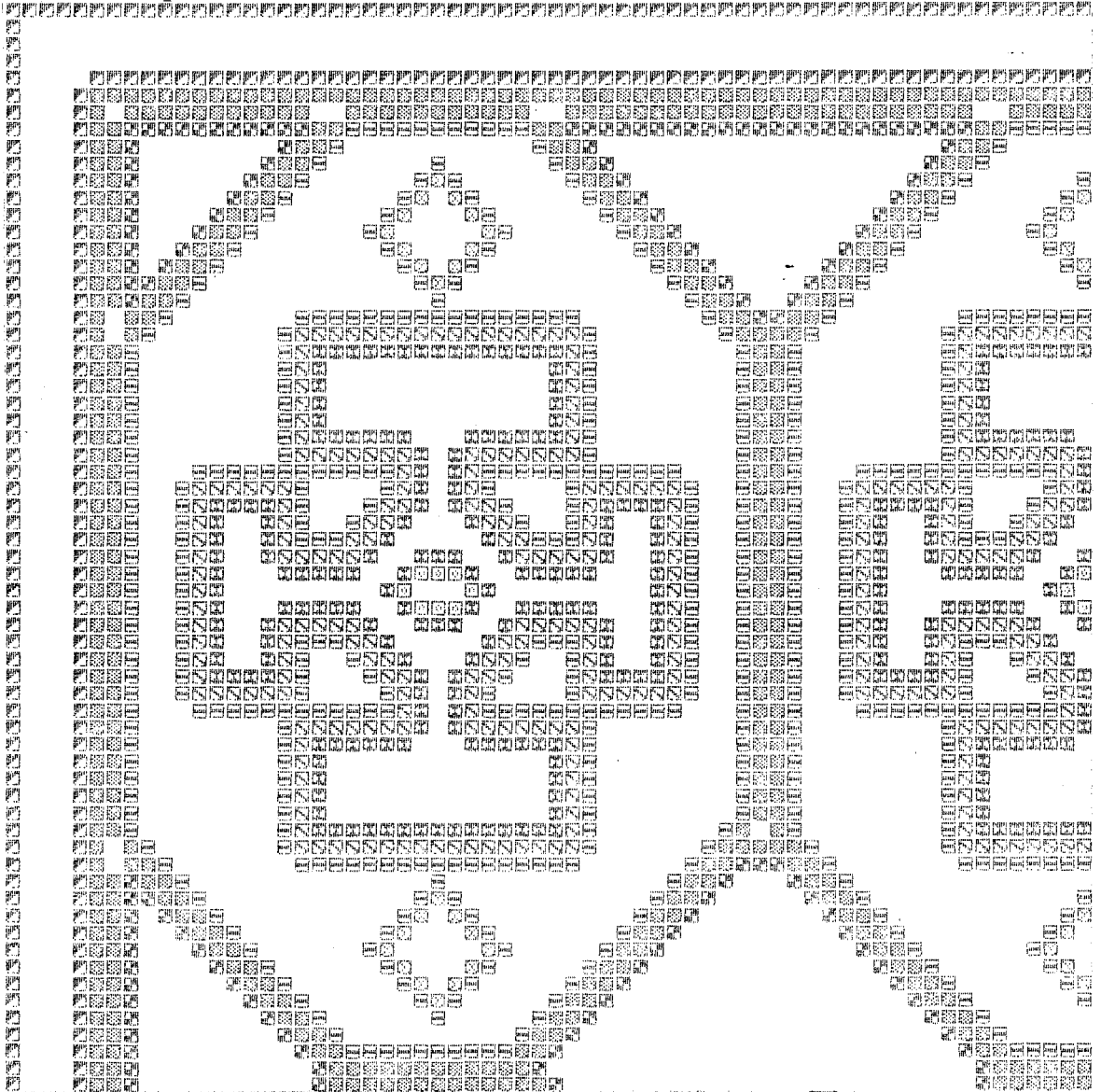


BÖLÜM 4

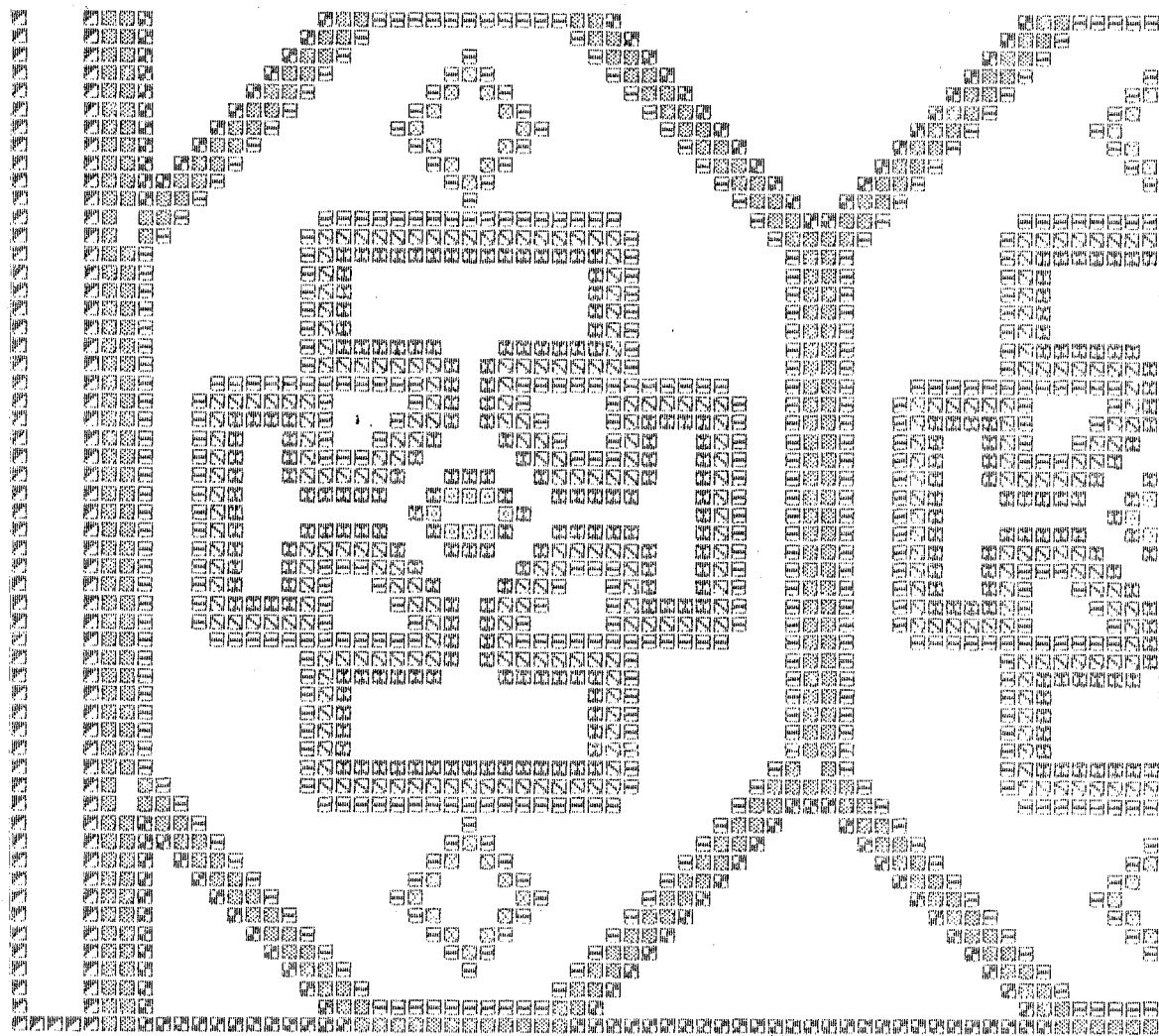


KÜTÜKİSMİ : DESEN03

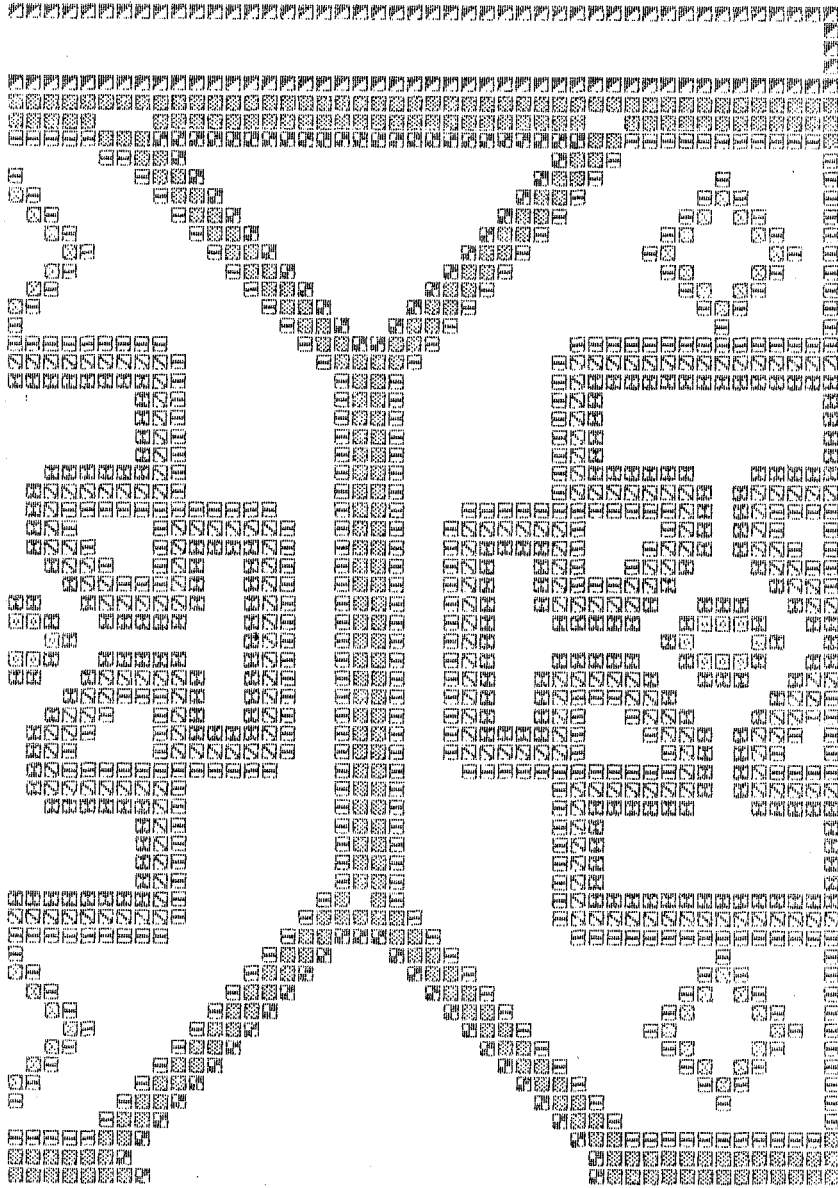
BÖLÜM 1



BÖLÜM 2



BÖLÜM 3



BÖLÜM 4

