

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ENDÜKTİF ÖĞRENME TEKNİĞİ İLE
ÇİZGİKODLARINI TANIMA

YÜKSEK LİSANS TEZİ

Metin BAYRAM

Enstitü Anabilim Dalı : ENDÜSTRİ MÜH.

OCAK 1996

T.C
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ENDÜKTİF ÖĞRENME TEKNİĞİ İLE
ÇİZGİKODLARINI TANIMA

YÜKSEK LİSANS TEZİ

Metin BAYRAM

Enstitü Anabilim Dalı : ENDÜSTRİ MÜH.

Bu tez .. / .. / 19.. tarihinde aşağıdaki jüri tarafından Oybirliği/Oyçokluğu ile kabul edilmiştir.

Jüri Başkanı

Jüri Üyesi

Jüri Üyesi

TEŞEKKÜR

Bu çalışmanın gerçekleşmesi için çalışma süresince her türlü teşvik ve fedakarlığı esirgemeyen , titizlikle yöneten ve bilgi ve tecrübelerinden azami derecede istifade ettiğim kıymetli hocam Sayın Yrd. Doç. Dr. M. Sabih AKSOY' a sonsuz teşekkürlerimi sunuyorum.

Çalışmalarım hakkında her zaman için maddi ve manevi olarak yakın ilgi ve alakalarını gördüğüm ve bize çok emeği geçmiş olan Endüstri Mühendisliği Bölümündeki bütün hocalarımıza teşekkürlerimi sunuyorum.

Bu çalışmanın hazırlanmasında maddi ve manevi emeği geçen mesai arkadaşları olmakla övündüğüm tüm TÜVASAŞ AR-GE personeline, Erdoğan UÇAR'a , Mustafa ÇETİNKAYA'ya ve adını burada anmadığım nice değerli şahıslara teşekkürü bir borç biliyorum.

Metin BAYRAM

İÇİNDEKİLER

KISALTMALAR	V
ŞEKİLLER LİSTESİ	VI
TABLolar LİSTESİ	VII
EKLER	VII
ÖZET	VIII
SUMMARY	IX
BÖLÜM 1. GİRİŞ	1
BÖLÜM 2. ÇİZGİKODU SİSTEMLERİ	4
2.1 Giriş	4
2.2 Uygulama Alanları	5
2.3 Çizgikodu Semboleri - Terimleri	9
2.4 Günümüzde En Çok Kullanılan Çizgikodu Alfabeleri	10
2.4.1 EAN	10
2.4.1.1 EAN-13	10
2.4.1.2 EAN-8	16
2.4.3 UPC	16
2.4.4 ITF-14	18
2.5 Çizgikodu Sisteminin Donanım Kısmı	19
2.5.1 Özel Donanımlı Bilgisayarlar	21
2.5.2 Esas Film	21
2.5.3 Okuyucular	23
2.7 Çizgikodu Yararları	24
2.8 Sonuç	26
BÖLÜM 3. ENDÜKTİF ÖĞRENME VE ALGORİTMALARI	27
3.1 Giriş	27
3.2 ID3 EÖ Algoritması	29
3.3 AQ-1 EÖ Algoritması	30
3.4 RULES-3 EÖ Algoritması	31

3.4.1	Bilgitabanı oluşturmak için örnek problem	33
3.5	Endüktif Öğrenme Uygulamaları	36
3.6	Sonuç	37
BÖLÜM 4. ÇİZGİKODU TANIMA İÇİN ALTERNATİF BİR TEKNİK ...		38
4.1	Giriş	38
4.2	Alternatif çizgikodu tanıma tekniği	39
4.2.1	Alternatif çizgikodu tanıma tekniğinin donanımı	40
4.2.2	Alternatif çizgikodu tanıma tekniğinin anma boyutları, etiket toleransları ve basımı	41
4.2.3	Alternatif çizgikodu tanıma tekniğinin alfabesi	46
4.2.4	Alternatif çizgikodu tanıma tekniğinin okuyucusu	46
4.3	Alternatif Çizgikodu Tekniğinin Yazılımı	46
4.3.1	Uygulamada Kullanılan Örnekler Setinin Elde Edilmesi	48
4.3.2	Bilgi Tabanının Oluşturulması	50
4.3.3	Programda Kullanılan Menüler	54
4.4	Sonuç	56
BÖLÜM 5. SONUÇ		58
BÖLÜM 6. TARTIŞMA VE ÖNERİLER		60
KAYNAKLAR		61
EKLER		
EK-1	63
EK-2	64
ÖZGEÇMİŞ		93

KISALTMALAR

EAN	Avrupa Mal Numaralama Birliđi
EAN-8	Avrupa Mal Numaralama Birliđi'nin 8 Karakterli Kodlama Sistemi
EAN-13	Avrupa Mal Numaralama Birliđi'nin 13 Karakterli Kodlama Sistemi
EÖ	Endüktif Öğrenme
ISBN	Milletlerarası Standart Kitap Numaraları
ISSN	Milletlerarası Standart Seri Yayınlar Numaraları
ITF	Interleaved Two Of Five Code
ITF-14	Interleaved Two Of Five Code'nin 14 Karakterli Kodlama Sistemi
MMNB	Milletlerarası Mal Numaralama Birliđi
MMNM	Milli Mal Numaralama Merkezi
OCR	Optik karakter Okuma (Optical Character Recognition)
PDTS	Personel Devam Takip Sistemi
UPC	Uniform Product Code
US	Uzman Sistem
YZ	Yapay zeka

ŞEKİLLER LİSTESİ

Şekil 2.1	Anma boyutlarda 13 dijital EAN-13 Sembolü	12
Şekil 2.2	Tablo 2.3'teki EAN-13 numaralarına tekabül eden çizgi ve boşluk kombinasyonları	15
Şekil 2.3	Anma boyutlarda 8 dijital EAN- 8 Sembolü	17
Şekil 2.4	Anma boyutlarda 14 dijital ITF-14 Sembolü	20
Şekil 2.5	Baskı Şişmesi İçin Kabul Edilebilir Tolerans	22
Şekil 3.1	Örnek bir karar ağacı	28
Şekil 4.1	Alternatif çizgikodu tanıma tekniğinin donanımı (Seçenek 1)	42
Şekil 4.2	Alternatif çizgikodu tanıma tekniğinin donanımı (Seçenek 2)	42
Şekil 4.3	Alternatif çizgikodu tanıma tekniğinin simülasyon yöntemiyle üretilmiş çizgikodları (Çizgikodu1-30)	43
Şekil 4.4	Çizgikodu1 için alternatif tekniğin okuma alanı	47

TABLolar LİSTESİ

Tablo 2.1	A,B,C modül serileri	14
Tablo 2.2	Sol karakter grubu için kullanılacak modül serileri	14
Tablo 2.3	EAN-13'te karakterlerin kodlanması	15
Tablo 2.4	ITF-14 numaralama sistemi	19
Tablo 3.1	Araba-ivmesi problemi için örnekler seti	33
Tablo 3.2	Araba-ivmesi problemi için endüktif kural çıkarma işleminin özeti	35
Tablo 3.3	Araba-ivmesi problemi için otomatik oluşturulan veri tabanı	36
Tablo 4.1	Alternatif çizgikodu tanıma sisteminin örnekler seti	50
Tablo 4.2	Alternatif çizgikodu tanıma sisteminin endüktif kural çıkarma işleminin özeti	52
Tablo 4.3	Alternatif çizgikodu tanıma sisteminin bilgi tabanı	53

EKLER

EK-1	Alternatif çizgikodu tanıma tekniğinin Veri Tabanı	63
EK-2	Programın Yazılımı	64

ÖZET

2000' li yılların eşiğinde dünya, özellikle bilgisayar alanındaki dahi gelişmeler sebebiyle bilgi çağının zirvesine doğru ilerlemektedir. Kıt kaynaklar arasında zamanın ekonomik biçimde kullanılması için çeşitli araçlar geliştirilerek en kısa zamanda, en fazla verim almanın yol ve yöntemleri uygulanmaya konulmaktadır. Bilgisayarların hızlı ve yoğun işlem yapma özelliklerinden insanoğlu azami derecede istifade etmesini bilmiştir. Bilgisayarlar bilgiyi çok kolay elde etme ve çok kısa süre içinde elde edilen bilginin işlenebilmesi imkanını vermiştir.

Bilgisayarların sağladığı kolaylıkların yanında çizgikodu sistemleri bir dizi ince ve kalın çizgiler ile aralarındaki boşluklardan ibaret basit yapıları sayesinde bilgiye erişme ve gerekli bilgiyi temin etmede sıkça başvurulan bir yöntem haline gelmiştir. Örneğin bir markette bilgisayar çizgikoduna bağlı olarak fiyatı arar ve kasaya bildirir, kasa kaydeder ve aynı anda bu bilgileri alıcının görebileceği yerde ekrana getirir. Tüm işlemler alıcının gözü önünde ve kontrolünde gerçekleşir. Yayıncılık alanında da uzun bibliyografik kimlikler yerine uluslararası düzeyde kullanılan bir kodlama sistemi ile kitabın yazarı, basım yılı, adı, yayınevi vb. bilgilerin tek bir yayını temsil etmesi sağlanmıştır.

Bir çizgikodu, etiketi ve çizgikodu okuyucusu ile sınırlanamaz. Bunların yanısıra özel donanımlı bilgisayarlar, bilgisayar programları, elektronik devre cihazları, master film, okuyucu çizgikodu sistemlerinin temel elemanlarıdır. İlk yatırım maliyetleri fazla olan bu sistemlerin kullanılmadan önce maliyetleri iyice hesaplanmalıdır.

Bu bağlamda Yapay Zeka'nın dallarından olan Endüktif Öğrenme'nin çok ilginç bir uygulama alanı ortaya çıkarılmıştır. Halen uygulamada kullanılan belirli sayıdaki Endüktif Öğrenme algoritmalarından RULES3, daha basit olması ve daha fazla avantajlara sahip olması sebebiyle bu çalışmada kullanılmıştır.

Bu çalışma ile, daha az bir maliyet gerektiren, çizgikodlarını bilgisayarın hafızasında saklamak ve çağırmak için daha az hafızaya ve daha kısa işlem zamanına ihtiyaç duyan Endüktif Öğrenme tekniği ile çizgikodu tanıma işlemi, alternatif bir sistem olarak sunulabileceği rahatlıkla söylenebilmiş olmaktadır.

SUMMARY

Approaching to 2000, computer technology have been rapidly developing to catch the information age. As time is very important some new methods should be developed to get better results within a short time.

Bar-code systems which include a range of thin and thick bars and spaces between them is a frequently applied method for processing knowledge and easily recall. For example, in the big markets that sell wide range of products computers seek the price and other information using bar codes and monitor them. Also customers can monitor all the knowledge. The process is performed within short time because of the simple structure of bar codes. Also with international bar code system all the knowledge represented including the writer, publication date, the name of the book etc. is represented in one code instead of long bibliographic identities.

The requirements for processing bar codes are not only labels and scanners. Specially equipped computers, softwares, electronic circuit boards and master films are some other basic elements of a bar code system. Before deciding to install such a system, the financial analysis should be done as the first installment is usually expensive.

In this study, it is demonstrated that bar-code systems are very suitable applications for inductive learning. A number of induction algorithms are available, but, in this study, RULES3 induction algorithm is chosen because of its more simple implementation structure and some advantages over others.

Finally, it can be easily said that Inductive Learning is an alternative way to process bar codes. The system requires less memory to store and call codes, shorter processing time, and is cheaper.

BÖLÜM 1. GİRİŞ

Ülkeler gelişmelerine paralel olarak, içte ve dışta yeni şekiller almaktadır. Ülke içinde hammaddenin hareketi, üretimin planlanması, mamulün pazarlanması alışageldiğimiz boyutları aşarak yepyeni bir felsefe ve sistemle yürütülmesi zaruretini ortaya çıkarmıştır. Aynı şekilde uluslararası düzeyde ticaretin ve ölçülerin yoğunluğu değişmiş, ülkelerin birbirlerine olan ekonomik ve ticari bağımlılıkları artmıştır.

Makro seviyedeki bu değişiklikler, teknolojiadaki gelişmeler, kaliteli ve ekonomik mal ve hizmet üretme ihtiyacı işletme yönetiminde ve üretimde denetim mekanizmalarını kolaylaştırmak, işlemleri daha hızlı ve doğru yapabilmek için bilgisayar uygulamasına geçilmiş ve ticareti yapılan malların trafiğinde bilgisayar mantığına uygun numaralar kullanılmaya başlanmıştır. Ancak, üzerinde herkesin ittifak ettiği standart metodların eksikliği çeşitli firmaların aynı ürünler için farklı numaralar kullanmalarına yolaçmış, bu durum ticarete bir takım engeller ortaya çıkarmıştır. Bu nedenle işletmeler arasında iletişimin kolaylaşmasını sağlamak, perakende satışta hızı artırmak, stok kontrol ve hızlı mal akışını düzenlemek için bütün firmaların kullanacağı ortak bir dil ve standart bir sistemin gereği ortaya çıkmıştır.

20 nci Yüzyılın son çeyreğinde bilim ve teknoloji alanında meydana gelen baş döndürücü gelişmeler sonucu tüketim alışkanlığını daha da arttırmıştır. Bir çok firma ve dolayısıyla bu firma ürünleri piyasalarda kendine yer bulma savaşı vermiştir. Bu tüketim çılgınlığına paralel olarak artan firma ve firma ürünlerinin sayısı uluslararası düzeyde bir denetim organizasyonunu zorunlu kılmıştır.

Bu denetimin gerçekleştirilmesi için bir kodlama sisteminde, herhangi bir kodun tek bir firmanın tek bir mamulünü temsil etmesini sağlamak amacıyla çizgikodu sistemleri ortaya çıkmıştır.

İthal malların giderek artan bir oranda market ve bakkallarımızı doldurduğu bu günlerde, hepimiz ambalajların üzerinde ince ve kalın çizgilerden oluşan çizgikodları görmekteyiz. Çizgikodu bilgisayarlara bilgi aktarmak için geliştirilmiş bir yöntemdir. Çizgikod bir dizi ince ve kalın çizgiler ile aralarındaki boşluklardan oluşan basit bir yapıya sahiptir. Kıt kaynaklar arasında en önemlisi olan zamanın ekonomik biçimde kullanılması için çeşitli araçlar geliştirilerek, en kısa zamanda, en fazla verim almanın yol ve yöntemleri uygulamaya konulmaktadır. Hızlı ve yoğun işlem yapma ihtiyacı bilgisayarın ortaya çıkmasına ve gelişmesine sebep olmuştur. Bugün hayatımızın her

safhasında yaygın olarak kullanılabilen bu araç çağdaşlaşmanın bir işareti haline gelmiştir.

Türkiye ekonomisi son yıllarda hızla dışa açılmış ve özellikle Avrupa Mal Numaralama Birliği (EAN)'a üye ülkeler ile ticaretini geliştirmiştir. Türkiye'nin dışa sattığı tarım ürünlerinin ve sanayi mamullerinin önemli bir bölümü tüketim mallarıdır. İhraç mallarımızın genelde parça ambalajını gerektirmesi ve dış pazarlarda etiketleme ve tanımlama yönünden sorunlarla karşı karşıya kalması ülkemizde de çizgi kod işaretlerini uygulama ihtiyacını ortaya çıkarmıştır. Bu nedenle başta tüketim malları ihraç eden kuruluşlarımız olmak üzere tüm ihracatçı ve üretici şirketlerimiz çizgi kodu benimseyip süratle uygulamışlardır.

Çizgi kod sisteminin firma ürünlerine uygulanmasıyla üreticiler dünyanın birçok yerinde ticaret yaptıkları kuruluşlar nezdinde geçerli olan bir numara ve işarete sahip olmuşlardır. Böylece firma ürünleri gerek iç pazarda gerekse de dış pazarlarda denetim altına alınmış ve belli bir ortak dil kullanılarak standartlaştırılmışlardır.

Firmaların çizgi kod sistemini uygulamaları, sistemden yararlanmaları yanında üretim, ticaret ve satış hizmeti ile ilgili bilgilere sahip olmalarına ve bu alandaki projelerin uygulama ve geliştirilmesi yönündeki çalışmalara katılmalarına da imkan verecektir.

1960'lı yıllarda ABD'deki perakendeci ve bilgisayar yapımcısı firmaların ortak araştırmaları sonucu ilk tarama teknikleri ve bilgisayar bağlantılı teçhizatın büyük mağazalarda uygulanmasına başlanmıştır. Daha sonra sistem geliştirilerek 12 basamaklı numaralardan oluşan Uniform Product Code (UPC) Sistemi malların tanımlanmasında standart olarak kabul edilmiştir.

UPC sisteminin kuruluşunu takip eden yıllarda, Avrupa ülkelerinde de malların tanımlanmasında bir standart belirlemek amacı ile çalışmalar başlamıştır. 1974 yılında Fransa, Belçika, Hollanda, Almanya, Avusturya, İngiltere, İsveç, Norveç, Finlandiya, Danimarka, İtalya ve İsviçre'nin üretici ve satıcı işletmelerinden oluşan temsilciler teknik bir çalışma grubu meydana getirerek UPC sistemi ile uyumlu Avrupa Mal Numaralam Sistemi'ni geliştirmişlerdir.

1977 yılında ise, sisteme yasal bir statü kazandırmak ve uygulama kurallarını ortaya koymak amacıyla merkezi Brüksel'de bulunan EAN kurulmuştur. Birliğe kısa bir süre sonra dünyanın birçok ülkesinden katılanlar olmuş ve EAN Avrupa sınırlarını aşp, uluslararası bir nitelik kazanmıştır. 1981 yılında Birliğin adı da bu uluslararası

statüsünü belirtecek şekilde Milletlerarası Mal Numaralama Birliği (MMNB) olarak değiştirilmiştir. Halen Türkiye’inde üye olduğu EAN’nin kısa adı ise, numaralama ve işaretleme sistemini belirtmek üzere aynı kalmıştır.

Çizgi kod uygulamalarının temel amacı, üretici, her kademedeki satıcı/aracı ve tüketici arasında uluslararası bir iletişim dili oluşturmaktır. Bu sistem şimdiden tüm Avrupa ülkeleri ve Japonya’nın yanısıra, Güney Afrika, İsrail, Avustralya, Yeni Zelanda ve en son olarak Türkiye dahil olmak üzere 40’a yakın ülkede kabul görmüştür.

Dünya genelinde muhtelif kodlama sistemleri endüstride ve ticari hayatta muhtelif amaçlarla kullanılmaktadır. Örnek verecek olursak mağazalarda kasa çıkışları, personel devam kontrol sistemleri, işletmelerde malzeme ve stok takip işi ve üretim izleme faaliyetlerinde çizgikodlarından azami derecede faydalanılmaktadır.

Çizgikodunun yapısı basit olmasına karşın sadece bir etiket ve okuyucudan ibaret olmayıp özellikle özel donanımlı bilgisayarlara ve elektronik devre cihazları gibi karmaşık ve ekonomik olarak bir külfet gerektiren bir sistem olması, bu iş daha başka nasıl yapılabilir ve daha basitleştirilebilir mi sorularını gündeme getirmiştir.

BÖLÜM 2. ÇİZGİKODU SİSTEMLERİNİN TANITILMASI

2.1 Giriş

Çizgikod bilgisayarlar için geliştirilmiş bir yöntemdir. Çizgikod bir dizi ince ve kalın çizgiler ile aralarındaki boşluklardan oluşur. Optik bir okuyucu ile algılanan çizgiler bir elektronik devre ile dijital sinyal olarak bilgisayara iletilir. Bu sayede klavye yerine çok daha hassas ve süratli bir şekilde bilgiler bilgisayara aktarılmış olur. Optik bir okuyucu ile algılanan çizgiler, bir elektronik devre ile sayısal olarak bilgisayara iletilir. Daha doğrusu çizgikodu sistemlerinde bilgiler klavye yerine çizgikodları ile girilir. Bilgisayar kullanan kuruluşlar işlemlerle ilgili verilerini direk klavyeden (On-line) veya önce kağıda aktarıp (Off-line) daha sonra klavyeden bilgisayara girerler. Her iki yöntemde de iş hacminin artması oranında hata payları yükselmektedir. Kullanıcıların bilgileri klavye ile girilmesindeki hata oranı 1:300' dür. Oysa bilgilerin çizgikodlanması ile bu hata oranı 1:10.000.000 gibi çok düşük bir orana erişir.

Bilginin mantıksal düzendeki siyah çizgiler ve beyaz alanların kullanılmasıyla ifade edildiği çizgikod teknolojisinin başarısı basit olarak gerçekleştirilmesinde yatar, fazla masraf gerektirmeyen güvenilir bir teknolojidir.

Çizgikodu sistemlerini iki grupta inceleyebiliriz. Birinci grup, uluslararası bir hüviyete sahiptir ki, bir birim malın hangi ülkenin hangi işletmesinde üretildiği veya ambalajlandığını, malın cinsini ve çeşitli özelliklerini tanımlamak amacıyla, önceden belirlenmiş kurallara uygun yani standartlaştırılmış bir dizi ince ve kalın çizgi ve boşluklardan ve anlamlı numaralardan meydana gelmektedir. İkinci grup ise bir işletme içinde, personelin giriş çıkış saatlerini izleyerek puantajın yapılması, üretimi izleme ve stok kontrolü ve benzeri faaliyetlerde veya bir satış mağazasında satın alınan ürünün fiyatının belirlenip en kısa zamanda alışverişin gerçekleştirilmesinde olduğu gibi uluslararası standart bir numaraya sahip olmayan, fakat standart bir çizgikodu alfabesi ile ifade edilen dar kapsamlı spesifik alanlar için geçerli olan sadece bir dizi ince ve kalın çizgi ve boşluklardan müteşekkildir.

Normal olarak bir ürünün ambalajı veya Personel Devam Takip Sistemi (PDTTS)'de olduğu gibi bir kimlik veya benzeri bir kartın üzerine basılan çizgikod, optik okuyuculu bir kalem yardımı ile veya bir ışın tarayıcısı ile okunabilen bir şifredir ve bu sayede malların bilgisayarlara bağlantılı optik okuyucu cihazlar ile yorumlanabilmesi ve malların kimliklendirilmesi işlemlerini rahatlıkla yapılabilmesini öngörür.

2.2 Çizgikodu Sistemlerinin Uygulama Alanları

Başlangıçta mağazalarda kasa çıkışlarında otomasyonu sağlayacak tüketim mallarını numaralama ve sembol işaretleme sistemi geliştirildikten sonra çizgikodu sistemleri, üreticiler, toptancılar ve perakendeciler tarafından kullanılacak üretim yöntemi, sevk, envanter hazırlama ve stok kontrolü gibi pek çok işlemlerde de benzer kolaylıkların sağlanması yollarını araştırmıştır. Araştırmalar sonucu dünyada mevcut çerçeveye uyum sağlayacak çeşitli numaralama yaklaşımları belirlenmiştir. Böylece bütün sevk maddelerinin Çizgikodu sistemlerinin temel ilkelerine uygun olarak belirgin bir şekilde tarif edilmesi garanti edilmiştir.

Çizgikod başlangıçta kütle halinde üretilen tüketim malları için uygulanmış daha sonra gazete, dergi, doküman, elektrikli ve elektronik cihazlar, ilaç gibi birçok alanda da uygulamaya konulmuştur. Süpermarketlerde bilgisayara bağlı yazarkasa terminalleri bu çizgikodları okuyarak binlerce malzeme içerisinde o malı "Tanıma" işlemini yapabilmektedir. Kitapların ve seri yayınların biyografik künyelerini şekil ve muhteva açısından standartlaştırmak için, belirli bir kitaba dair bilgi alışverişini tek bir numara ile gerçekleştirmek amacı ile Uluslararası Standardizasyon Organizasyonu (International Organization For Standardization) tarafından hazırlanan Milletlerarası Standart Kitap Numaraları (ISBN) ve Milletlerarası Standart Seri Yayınlar Numaraları (ISSN)'nin EAN çerçevesinde yürütülmesi 1980'de yapılan bir anlaşma ile sağlanmıştır. Böylece kitap ve seri yayınlar EAN flaması ile basılarak çizgikodlar yardımı ile standartlaştırılmıştır. Türkiye'de ISBN ve ISSN sisteminin uygulanması için Kültür ve Turizm Bakanlığı Kütüphaneler ve Yayınlar Genel Müdürlüğü çalışmalarına başlanmış ve bitirilmiştir.

EAN, kendisine üye ülkelerdeki firmalara yardımcı olmak amacıyla Kuzey Amerika'daki Benzer Mamul Kodlama Konseyi (Uniform Product Code Council) ile koordineli olarak bir program ortaya koymuştur. Programa göre Kuzey Amerika'ya ihracat yapan firmalar mamulleri üzerinde UPC numaralarını kullanabileceklerdir. Ayrıca EAN ve UPC sistemlerini bütünleştirilerek standart bir kodlama sistemi oluşturulması konusundaki çalışmalar devam etmektedir.

Yapılan arařtırmalar sonucu elde edilen verilere göre bir ürün pazara çıkartılıncaya kadar yapılan ham/ara/mamül madde taşıma hizmetlerinin malın cinsine göre maliyetin %30 ila %95'ini oluşturmaktadır. Bu işlemlerin takibinde bilgisayar ve çizgikodu sistemleri kullanılabilir.

İřletmelerde, hammadde girişinden sevkiyata kadar geçen süre içinde ;

Hangi mal? Ne zaman?

Kaç kişiyle? Kiminle? Ne kadar zamanda?

Hangi iş istasyonunda? Kaç parça?

Hangi malzeme ile? Kaç adet? Nerede?

Ne zaman? Hangi vasıta ile? Ne kadar? vb. sorulara anında doğru yanıt alabilmek; doğru malzemeyi , doğru metod ile istenilen yere istenilen zamanda ve minimum maliyetle göndermek/almak/götürmek mümkündür. Bu kadar çok bilgiyi Çizgikodu yöntemi kullanmadan anında takip etmek çok zordur.

Her işletmede;

- “Dün üretim hattımızda planladığımız işleri bırakıp başka mal üretmek zorunda kalınması ve gerekli malzemelerden birisi kayıtlarımıza göre imalatçı firmadan henüz bize gönderilmemiş olması durumları sözkonusu olabilmekte, oysa adı geçen mal üç günden beri ambar girişinde bir palet üzerinde beklediğini tesadüfen farkedilmesi”,

- "Envanter kayıtlarında yapılan bir hata nedeniyle kritik bir malzemeyi kısa sürede teslim almak için üreticisine %50 daha fazla para ödenmesi",
- "Bir üretim hattını, gerekli malzemenin yanlış hatta gitmesi nedeni ile 6 saat durdurmak zorunda kalınması",
- "Kolilere koyulacak malların tartılması , cinslerine göre ayrılması ve koli etiketi hazırlanması çok sorun doğurur ve emek yoğun işlem olduğu için hatanın fazla yapılması",
- " Yanlış yükleme , eksik yükleme nedenleri ile kaybettiğim zaman dışında malın israf olması da ayrı olay vb. sorunlar dolayısıyla sevk masraflarının artması durumları ile sıkça karşılaşılmaması" durumları her zaman için sözkonusudur.

Bunlara ilaveten işletmelerde malzeme ve stok takip işi , üretim izleme ve ambar takip sistemleri çizgikodu sistemleri ile başarıyla gerçekleştirilebilmektedir.

Çizgikodları hemen hemen her sektör için gerekli olan çizgikodlu "Personel Devam Kontrol Sistemi"nde başarıyla uygulanmaktadır. Çizgikodlu kimlik kartına sahip personelin işe giriş ve çıkışında bu kartı çizgikodu okuyucu cihazına okutması sonucu sistem, puantöre gerek kalınmaksızın puantajları hazırlayarak bordronun elde edilmesine imkan vermektedir. Bilgisayar kontrollü kart okuma cihazlarından toplanan giriş çıkış bilgilerine bağlı olarak personelin bir puantaj dönemi içerisinde değişik mesailerden kaç gün ve kaç saat süreyle çalışmış olduğunun tespiti, raporlanabilmesi ve bu bilgilerin bordra hesaplayan bir programa aktarılması rahatlıkla sağlanabilmektedir. Bunun yanında tüm personelin devam durumunu izlemek, çalışma sürelerinin kişi departman ve bölümler toplamı bazında raporlamak gibi amaçlarda sözkonusudur. Beşyüzden fazla işgören çalıştıran kurum ve işletmelerde puantör kadroları ile rahatlıkla günlük giriş/çıkış hareketleri ve değerlendirmesi yapılabilmesine karşın yönetim açısından hızlı raporlama ve değerlendirme ancak çizgikodu sistemleri ile gerçekleştirilebilmekte ve gereğinden fazla puantör istihdamı önlenebilmektedir.

Çizgikodları siparişler, stok kontrolleri, ulusal ve uluslararası düzeyde kütüphanelerarası ödünç verme işlemleri, kitaplara ilişkin bilgilerin bilsayara yüklenmesiyle kitap ve kütüphane ile ilgili diğer faaliyetlerin yürütülmesinde büyük kolaylıklar sağlamaktadır. Çizgikodu basılı kitaplar ve broşürler, eğitici filmler/videolar ve slayt dahil diğer benzeri yayınlar kaset şeklindeki kitaplar, mikro bilgisayar yazılımları, elektronik yayınlar, mikroform yayınlar, görmezler için yayınlar, haritalar ve benzerleri için kullanılır.

Mağazadaki bilgisayarlara malın kodu, adı, cinsi, kalitesi, ait olduğu raf, alış ve satış fiyatı, asgari stok seviyesi her mal için işlenir ve genel mal kütüğü elde edilir. Fiyat ve mal akımındaki değişiklikler de bu kütüğe ilave edilir.

Fabrikalardan toptancı ve perakendecilere gelen malın numarası optik okuyucular ile okunur, verilen bu kodun bilgisayarlarda yüklü bilgileri otomatik olarak ekrana gelir. Bu safhada gelen malların miktarı bilgisayara işlenir ve mal ait olduğu rafa gönderilir.

Mağaza veya stok yerlerindeki raflar özel cihazlar ile malın kodunu ve fiyatını kolayca gösterecek şekilde etiketlenir. Yeni mal siparişleri verileceği zaman, portatif bir okuyucu raflardaki çizgikodları okur ve sipariş miktarını kaydeder. Bu portatif cihaz bağlandığı bilgisayarlara bu bilgileri aktarır. Hazırlanan listeler ilgili firmalara gönderilerek siparişler son derece seri ve hatasız şekilde yapılır.

Ayrıca, yaş meyve, sebze ve et gibi özelliği olan mallar tüketici talepleri açısından parakende satıldığından, bu mallar bölünerek tartılır ve elektronik tartı aletine sadece bu malın kodu verilir. Tartı aleti bu malın ağırlığı ile birim fiyatını çarparak malın çizgikodunu, ağırlığını ve fiyatını etiket halinde basar.

Perakendeciler ise, müşteriler aldıkları malları ile kasa önüne gelirler. Kasiyer, malların çizgikod basılı kısımların seri şekilde optik okuyucunun önünden geçirdiğinde yazar kasa o malın adını ve fiyatını mal kütüğünden bularak fişe yazar. Bu kayıtlar, merkezi bilgisayarda toplanır ve defter kayıtları otomatik olarak elde edilir.

2.3 Bazı Çizgikodu Terimleri

TSE tarafından hazırlanan bazı çizgikodu terimleri şunlardır;

Anma boyutları: Çizgikod sembolünün referans boyutları olup, gerçek boyutu bulmak için anma boyutları büyültme faktörü ile çarpılır.

Çizgikod sembolü: Bilgileri kodlamak için kullanılan bir dizi ince ve kalın çizgiler ile aralarındaki boşluklardan meydana gelen ve bilgilerin okuyucular yardımıyla bilgisayarlara aktarılmasını sağlayan semboldür.

Dijital karakter: Sayısal değerleri çubuk ve boşluk ile temsil eden modül grubudur.

EAN: MMNB tarafından geliştirilen numaralama ve sembol işaretleme sistemidir.

Esas Film: Kodlanan bilgileri çizgikod sembolüne dönüştüren, hassas fotoçizici makinalarda üretilen ve ambalaj baskısında bire bir olarak kullanılan pozitif veya negatif klişe filmdir.

Işık Marjı: Çizgikod sembolünün sağlıklı okunabilmesi için sembolün etrafında bırakılması gereken açıklıktır.

Kodlama: Kod şekline getirmek işlemidir.

Kontrol Dijiti: Bilgilerin doğru kodlanıp kodlanmadığını belirlemek amacıyla bilgiye eklenen dijittir.

Optik Karakter Okuma (OCR-B): Makinalar tarafından okunabilen yazı karakteridir.

Okuma: Çizgikodların çizgikod okuyucular ile taranmasıdır

Okuyucu kalem: Elde taşınabilir, kalem şeklinde çizgikod okuyucudur.

Optik Okuyucu: Çizgikod sembolünü lazer ışınıyla tarayan cihazdır.

2.4 Günümüzde En Çok Kullanılan Çizgikodu Alfabeleri

Ülkemizde içinde üye olarak bulunduğu EAN süpermarket ve perakende satışta kullanılan bir çizgikodu standarttır. ABD’ de bu amaçla UPC, Japonya’da JPN, Avustralya’da da AAN kodları kullanılmaktadır. Tüm dünyada etiketleme ve ambalaj standartının bir parçasıdır.

Market çizgikodları dışında değişik endüstrilerde standart haline gelmiş CODE 39 (Kağıt, Sağlık, Otomotiv ve Askeri) gibi kırkın üzerinde değişik çizgikodu alfabeti bulunmaktadır. Her piyasada olduğu gibi çizgikodu konusundada değişik firmalar kendi geliştirdikleri çizgikodları endüstri standartı haline getirmeye çalışmışlar ve sonunda EAN, ITF, UPC, Code 11, Code 93, Code 39, Codebar, Code 49, 12/5 ve 2/5 çizgikodu alfabetleri en çok kullanılan çizgikodu alfabetleri olmuşlardır.

Aşağıda günümüzde en geçerli uluslararası bir numaraya da sahip olan çizgikodu alfabetlerinden sırasıyla EAN, UPC ve ITF-14 kısaca izah edilecektir.

2.4.1 EAN

EAN’da iki çeşit numaralama vardır. Bunlar Avrupa Mal Numaralama Birliği ‘nin 13 Karakterli Kodlama Sistemi (EAN-13) ve Avrupa Mal Numaralama Birliği ‘nin 8 Karakterli Kodlama Sistemi (EAN-8)’dir.. Genel olarak mamuller üzerinde 13 karakterden oluşan EAN-13 kullanılmasına karşın, ciklet gibi daha küçük mamullerde basılacak alanın darlığından dolayı, 8 karakterden oluşan EAN-8 kullanılır.

2.4.1.1 EAN-13

EAN-13 sistemi, 13 rakamdan oluşur ve sırasıyla o mamulün üretildiği ülkeyi, üretici veya satıcı firmayı ve ürünü tanımlar. Sistemde numaraların dizimi sağdan sola doğrudur. Yani ilk rakam 13 üncü hanedir. 13 rakamdan oluşan EAN-13 kodunda ayrıca sol ve sağ başta birer "koruyucu" ve sol ve sağ karakter grubunu birbirinden ayıran "orta" bulunur. Bunların boyları diğer karakter çizgilerinden uzundur.

- 1- 13-12 nci (ve bazen 11 nci) hane ülke kodudur. Her ülkeyi temsil eden numaralama teşkilatı EAN'a üye olarak bir ülke kimlik numarası alır.
- 2- 10-9-8-7 nci hane firma kodudur. Her ülkenin milli numaralama teşkilatına üye olmak isteyen üretici ve satıcı firmalara birer üretici ve/veya satıcı kimlik numarası verilir.
- 3- 6-5-4-3-2 nci hane firmanın ürün kodudur. Kendi ülkesindeki numaralama teşkilatına üye olan firmalar üretici kimlik numarasını aldıktan sonra kendi mamullerinin niteliklerini tanımlayan mamul kimlik numarası verir.
- 4- 1 nci hane kontrol rakamıdır. ilk 12 rakamın doğru kodlanıp kodlanmadığının kontrolünü sağlar.

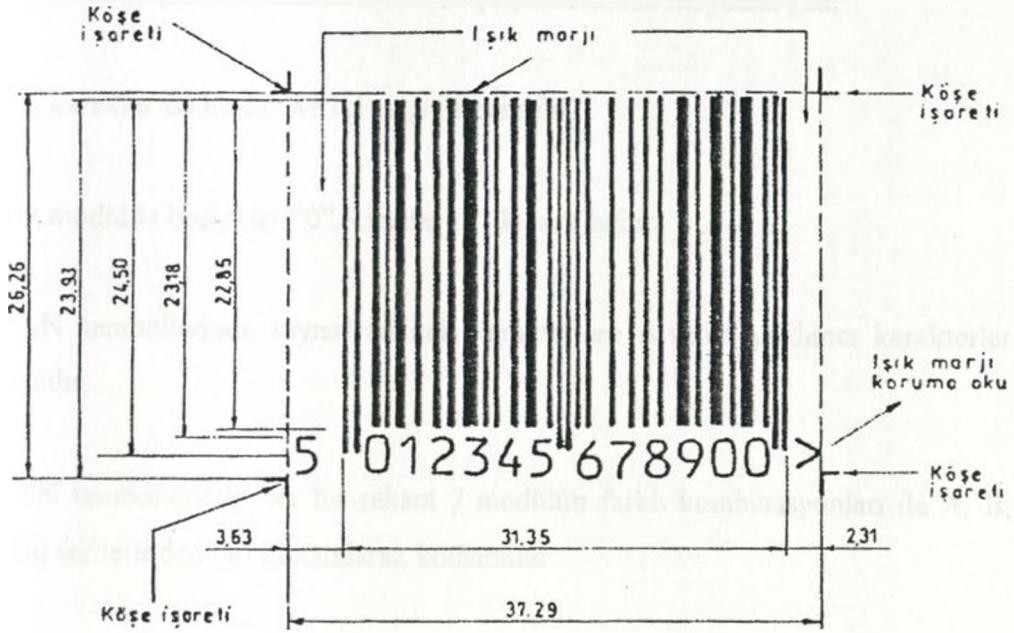
X X X	XXXXX	XXXXX	X
Ülke Kodu	Üretici	Ürün	Kontrol

Şekil 2.1'de EAN-13 çizgikodu alfabesinin anma boyutları ışık marjları ile birlikte verilmektedir.

EAN-13 Numaralarının Çizgikodu Sembolü Haline Dönüştürülmesi;

EAN-13 numaralarının çizgikodu sembolü haline haline dönüştürülmesinde aşağıdaki kurallara uyulur.

Ölçüler mm'dir



Şekil 2.1 Anma boyutlarda 13 dijital EAN-13 sembolü

- EAN sembolleri, dikdörtgen biçimli, hayali bir referans çizgisine dik ardı ardına çizilmiş çizgi ve boşluklardan meydana gelir.

- EAN sembollerinde bir karakter 7 modülden meydana gelir.

- Çubuk ve boşluk kalınlıkları modülün çarpanlarından meydana gelir.

- Her karakter iki boşluk ve iki çizgi ile belirtilir.

- Her modülde boşluklar "0", çizgiler "1" ile tarif edilir.

- EAN sembollerinde sayısal (dijital) karakterlere ilaveten yardımcı karakterler de kullanılır.

- EAN sembollerinde her bir rakam 7 modülün farklı kombinasyonları ile A, B, C, modül serilerinden biri kullanılarak kodlanırlar.

- A ve B modül serilerindeki sayılar 0 (boşluk), C modül serisindeki sayılar ise 1 (çizgi) ile başlar.

- EAN-13 soldan sağa doğru sırasıyla başlangıç koruyucu, sol karakter grubu için altı dijital karakter, orta karakter, sağ karakter grubu için altı dijital karakter ve bitiş koruyucudan meydana gelir.

- EAN-13'te ülke kodunun ilk rakamı olan on üçüncü dijital karakter olarak gösterilemez. on üçüncü dijital, sol karakter grubunun kodlama düzenini belirler. on üçüncü dijite göre, on iki ila yedinci dijitaler arasındaki rakamlar için kullanılacak modül serileri Tablo 2.2'de verilmektedir.

- EAN-13'te sol karakter grubunda yer alan altı dijital, on üçüncü dijitalin değerlerine bağlı olarak A ve B modül serilerinin karışımı olarak kodlanır. Sağ karakter grubunda yer alan altı dijital ise C modül serisi kullanılarak kodlanır.

Tablo 2.1 A,B,C Modül Serileri

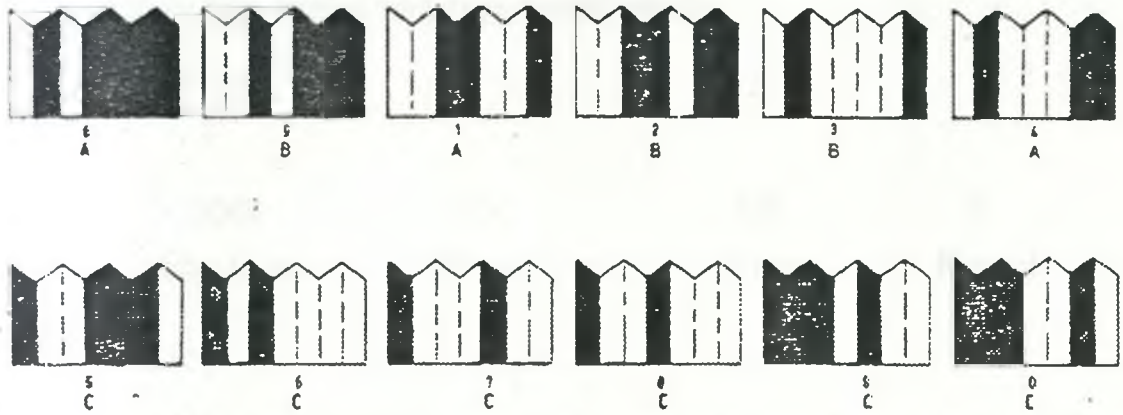
Dijital Değer	A Modül Serisi	B Modül Serisi	C Modül Serisi
0	0 0 0 1 1 0 1	0 1 0 0 1 1 1	1 1 1 0 0 1 0
1	0 0 1 1 0 0 1	0 1 1 0 0 1 1	1 1 0 0 1 1 0
2	0 0 1 0 0 1 1	0 0 1 1 0 1 1	1 1 0 1 1 0 0
3	0 1 1 1 1 0 1	0 1 0 0 0 0 1	1 0 0 0 0 1 0
4	0 1 0 0 0 1 1	0 0 1 1 1 0 1	1 0 1 1 1 0 0
5	0 1 1 0 0 0 1	0 1 1 1 0 0 1	1 0 0 1 1 1 0
6	0 1 0 1 1 1 1	0 0 0 0 1 0 1	1 0 1 0 0 0 0
7	0 1 1 1 0 1 1	0 0 1 0 0 0 1	1 0 0 0 1 0 0
8	0 1 1 0 1 1 1	0 0 0 1 0 0 1	1 0 0 1 0 0 0
9	0 0 0 1 0 1 1	0 0 1 0 1 1 1	1 1 1 0 1 0 0

Tablo 2.2 Sol Karakter Grubu İçin Kullanılacak Modül serileri

13. Dijitteki Rakam	Sol Karakter Grubu için Kullanılacak Modül Serileri					
	12	11	10	9	8	7
0	A	A	A	A	A	A
1	A	A	B	A	B	B
2	A	A	B	B	A	B
3	A	A	B	B	B	A
4	A	B	A	A	B	B
5	A	B	B	A	A	B
6	A	B	B	B	A	A
7	A	B	A	B	A	B
8	A	B	A	B	B	A
9	A	B	B	A	B	A

Tablo 2.3 EAN-13'te Karakterlerin Kodlanması

Dijital Değer	13	Başlangıç Koruyucu	12	11	10	9	8	7	Orta Karakter	6	5	4	3	2	1	Bitiş Koruyucu	
Modül Serisi			A	B	A	B	B	A		C	C	C	C	C	C		C
Örnek	8		6	9	1	2	3	4		5	6	7	8	9	0		



Şekil 2.2 Tablo 2.3'teki numaralara tekabül eden çizgi ve boşluk kombinasyonları.

- EAN-13'te on üçüncü dijital ülke kodunun ilk rakamı olması sebebiyle her ülke için düzenleme sabittir. Türkiye'nin kodu 869 olduğuna göre Tablo 2.2'e göre 8 rakamına karşılık gelen sol karakter grubu için kullanılacak modül serileri ABABBA'dır. Buna göre Tablo 2.1'deki "A" ve "B" modül serisinin rakamları serinin durumuna göre yerleştirilmelidir. Tablo 2.3'te ve Şekil 2.2'teki gibi modül serilerine göre karakterler kodlanır.

2.4.1.2 EAN-8

EAN-8'de ise, ilk 3 hane ülke kimlik numarası, daha sonra gelen 4 hane milli numaralama teşkilatları tarafından belirlenmiş mamul numarası ve 8 nci hane kontrol numarasıdır.

Ancak bazı ürünlerde yukarıda kısaca anlattığımız EAN 13'ü uygulamak için yeterli alan bulamayabiliriz. Örneğin bir çiklet ambalajında alan son derece kısıtlıdır. Benzeri durumlarda ancak yetkili makamın onayı doğrultusunda EAN-8 uygulanabilir. EAN-8'de ise bilgi dağılımı aşağıda gösterildiği gibidir

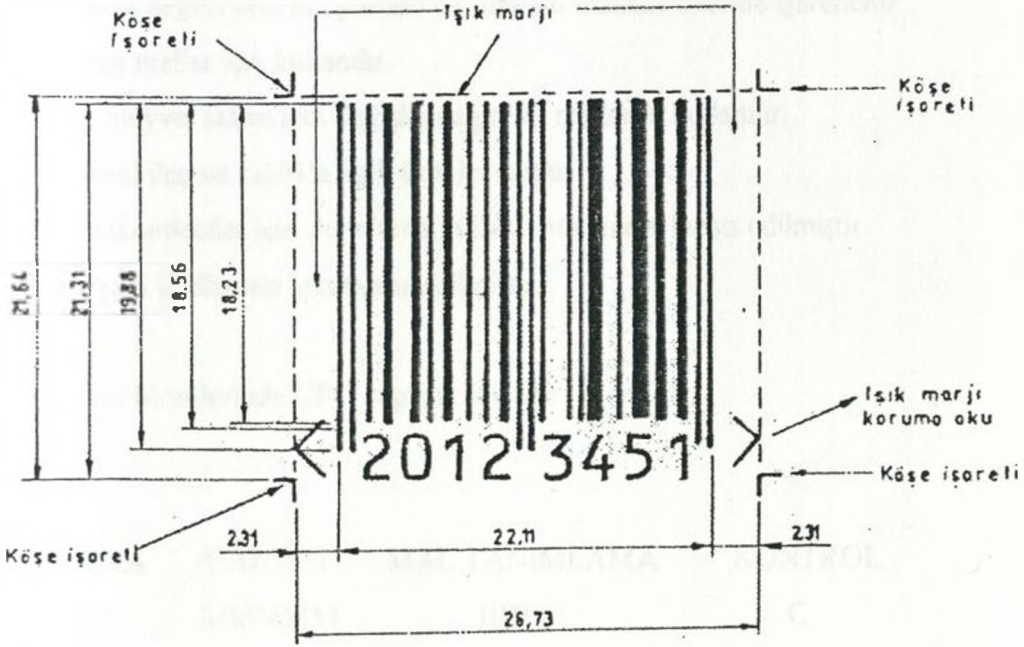
XXX	XX	XX	X
Ülke Kodu	Üretici	Ürün	Kontrol

Şekil 2.3'de EAN-8 çizgikodu alfabesinin ana boyutları ışık marjları ile birlikte verilmektedir.

2.4.2 UPC

ABD ve Kanada'da tüketim malları için UPC numaralama sistemi kullanılmaktadır. Toplam 12 basamaktan oluşan UPC numaralarında solda bir tek flama yer almaktadır. Sağda ise kontrol basamağı yer alır. Sistemde flama sol koruyucu çizgisinin dışında,

Ölçüler mm'dir



Şekil 2.3 Anma boyutlarda 8-dijitli EAN-8 sembolü

kontrol basamağı ise sağ koruyucu çizgisinin dışında yer almaktadır. Sistemin yapısı aşağıdaki şekildedir:

<u>FLAMA</u>	<u>KOD NUMARASI</u>	<u>KONTROL</u>
(0) P	XXXXXXXXXX	C

- P: Flama değeri üretim aşamasında tüketim malları üzerine işaretlenir.
 O: Tüm mallar için kullanılır.
 2: Et, meyve, sebze gibi değişken ağırlıklı mallarda kullanılır.
 3: Ulusal ilaç ve sağlıkla ilgili ürünler kodu.
 4: Perakendeciler için depolarda kullanılmak üzere tahsis edilmiştir.
 5: Kupon kodlaması için tahsis edilmiştir.

Tüketim birimlerinde UPC yapıları şöyledir:

<u>FLAMA</u>	<u>ÜRETİCİ</u>	<u>MAL TANIMLAMA</u>	<u>KONTROL</u>
(0)	MMMMM	IIII	C

EAN sisteminin kullanıcıları 13 basamak uzunluğuna göre bilgileri bilgisayara işlediklerinden EAN sisteminde sol başa 0 koyarak UPC numaraları okunabilir. Ancak UPC sisteminde sadece 12 basamak olduğundan, bu basamak alanı uzunluğunu arttırmadan EAN numaralarını kullanamazlar.

2.4.3 ITF-14

ITF-14, 14 dijitten meydana gelir. Dijital (sayısal) değerler sağdan sola doğru artar. ITF-14 numaralama sisteminde Çizelge-1'deki esaslara uyulur.

Tablo 2.4 ITF-14 Numaralama Sistemi

Dijital Değer	14	13	12	11	10	9	8	7	6	5	4	3	2	1
ITF-14	0	Ü	Ü	Ü	F	F	F	F	M	M	M	M	M	K
Örnek	0	8	6	9	1	2	3	4	5	6	7	8	9	0

Burada:

0= İlave basamak

Ü = MMNB tarafından bu birliğe üye olan kuruluşlara tahsis edilen iki veya üç dijital ülke kod numarası olup, Türkiye'nin kodu 869 olarak tespit edilmiştir.

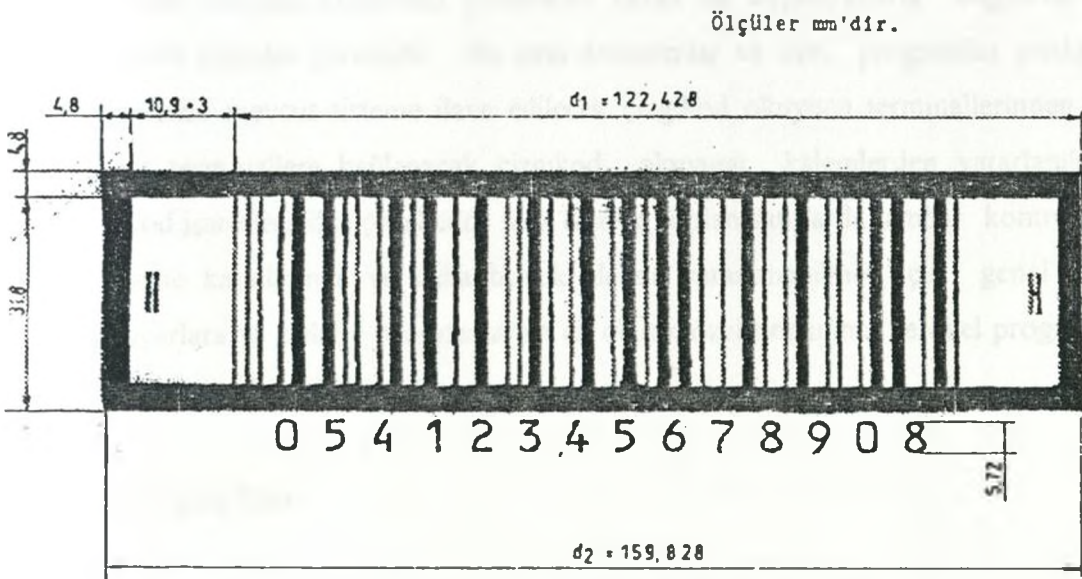
F= Her ülkedeki mamül numaralama merkezi tarafından firmalara tahsis edilen dört veya beş dijital firma kodudur.

K= Kontrol dijiti

Şekil 2.4'te ITF-14 çizgikodu alfabesinin anma boyutları ışık marjları ile birlikte verilmektedir.

2.5 Çizgi Kodu Sisteminin Donanım Kısmı

Çizgikod sistemleri ve yazar kasaları genelde standart IBM uyumlu bilgisayarlara bağlanmaktadır. Çizgikodu sistemini bir bütün olarak ele alındığında çizgikodu sadece sistemin bir parçasıdır. Bir çizgikodu etiketi ve de bir çizgikodu okuyucu olarak sınırlanamaz. Bir çizgikodu sisteminin temel elemanları çizgikodu, elektronik devre cihazları, esas film, okuyucu, bilgisayar ve bilgisayar programlarıdır. Bunlara ilaveten uygulama alanına göre ilaveten çizgikodlarını okuyabilen tipte yazar kasalar, büyük satış reyonlarında kullanılan ışıklı çizgikodu okuyucuları ile mal siparişi için raflarda yapıştırılmış çizgikodları okuyan portatif okuyucu ve hafıza sistemleri ve meyve, sebze, et gibi malları tartarak fiyatını ve çizgikodunu etikete basan perakende mal



Şekil 2.4 Anma boyutlarda 14-dijitli ITF-14 sembolü

paketleme ve etiketleme ve benzeri cihazlarda çizgikodu sistemlerinde kullanılmaktadır.

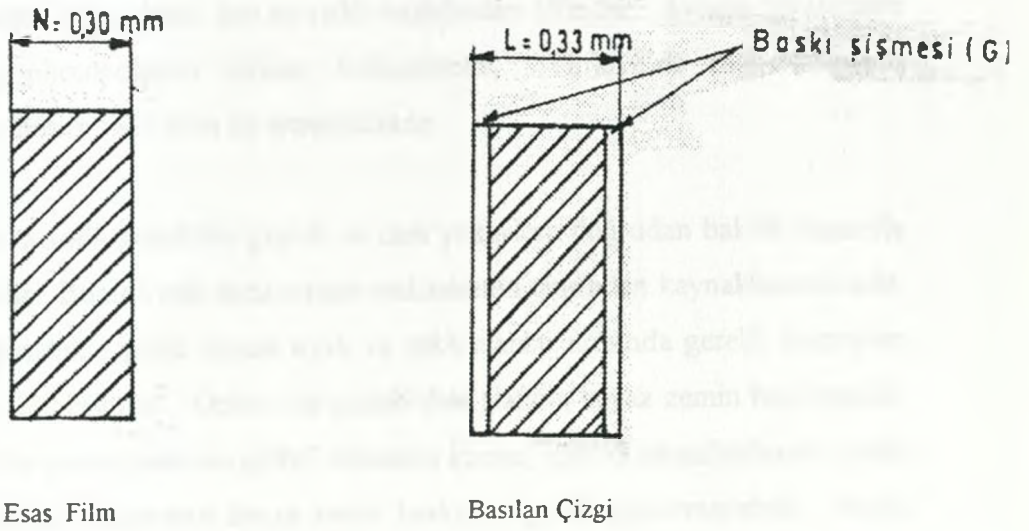
2.5.1 Özel Donanımlı Bilgisayarlar

Çizgikodu sistemlerini oluşturan temel ekipmanlar daha önce de bahsedildiği üzere bilgisayarlar, okuyucular ve çizgikodu etiketleridir.

Çizgikodun okunabilmesi, yani bu şifrenin çözülebilmesi için ışıklı kalem, sabit ışık üstü cam, okunan çizgikodu gösterecek ekran ve bilgisayarlarla bağlantılı özel donanımlı cihazlar gereklidir. Bu özel donanımlar ve özel programlar yardımıyla işletmelerde mevcut sisteme ilave edilecek çizgikod okuyucu terminallerinden veya mevcut terminallere bağlanacak çizgikod okuyucu kalemlerden yararlanılabilir. Çizgikod işaretlerinden perakende satış dışında toptan satışlarda, ambar kontrolünde, muhasebe kayıtlarında ve daha birçok alanda yararlanabilmek için genel amaçlı bilgisayarlar ve onların programlarına ek olarak özel donanıma ve özel programlara gerek duyulmaktadır.

2.5.2 Esas film

Çağımızın yaygın bilgisayarlarında uygun yazılım kullanılarak çizgikod düzenlemesi yapmak mümkündür. Bilgisayarlar, çizgikod esas film hazırlama sistemlerinde ancak front-end olarak kullanılmaya elverişlidir. Bunların baskı ünitelerinde 5 mikron duyarlılığına ulaşmak teorik ve pratik olarak mümkün değildir. Bilgisayarların en duyarlı print üniteleri, laser-printer'lerdir. Bunlarda günümüz standardı 300 d.p.i.'dir. Konuyu biraz açacak olursak; 1 inç =25,4 mm., baskı rezulasyonu 300 d.p.i. olduğuna göre ulaşacağımız hassasiyet $25,4/300=0.085$ mikrondur. Riskin %50 olduğu varsayımında bile risk 40 mikronu aşar yani kabuledilebilir toleransın 8 katından fazla risk vardır. Çok nadir rastladığımız 600 d.p.i. rezulasyonlu yeni jenerasyon bilgisayarlar da ki bunların değeri 25.000 \$ civarındadır. En iyi niyetle risk kabul edilebilir oranının 4 katından fazladır [İşmen, M., 1988].



Şekil 2.5 Baskı Şişmesi için kabul edilebilir tolerans.

Esas film, profesyonel cihazlarda üretilir. Baskı anında kullanılan esas filmlerin kalitesi ve teknik özellikleri okunabilir kodlar üretmek açısından çok önemlidir. Baskı yöntemine göre filmin negatif-pozitif, emülsiyon üstte-altta üretilmesi ve bunların kontakt print kademelerinden geçmeden birinci aşamada gerçekleştirilmesi zorunludur.

Flexo baskıda kullanılan baskı kalıbına bağımlı olarak sürekli problem oluşabilir. Bu başlangıçta karşılaşılabilecek sorundur. Ancak baskı sürecinde de mürekkep viskozitesi değişimine ve diğer faktörlere bağımlı olarak riskler oluşabilir. Ülkemiz koşulları gözönüne alınırsa, Flexo baskı için en riskli baskılardan birisidir. Avrupa ülkelerinde yaygın olarak photopolymer klişeler kullanılırken, Türkiye’de rağbet kauçuk klişeyedir. Bu durum riski daha da arttırmaktadır.

Serigrafi baskı metodu genellikle plastik ve cam yüzeylere doğrudan bakıda başarıyla uygulanmaktadır. Burada risk daha ziyade malzemenin cinsinden kaynaklanmaktadır. Malzemenin genellikle şeffaf olması siyah ve ışıklı alanlar arasında gerekli kontrastın oluşmasına engel oluşturur. Önlem ise çizgikodun alanına beyaz zemin basılmasıdır. Bazı durumlarda ise malzemenin şeffaf olmasına karşın, içine doldurulabilecek likidin kontrast bir zemin oluşturması beyaz zemin baskısına gerek göstermeyebilir. Böyle durumlarda deneme yaparak karara varılır.

Gravür baskıda, baskı sürecinde genellikle fazla bir problemle karşılaşılmaz, nedir ki, bu yöntemde baskı silindirleri oldukça pahalıdır. Bu nedenle, sorunlar baskı silindiri yapılmadan çözülmeli ve imajın esas filme uygunluğuna özen gösterilmelidir.

Üstünde önemle durulması gereken bir diğer nokta da siyah/beyaz alanların kontrast dansite farklılığı ve yansıtma ölçümüdür. Çoğu kez karton, kağıt vb. baskılı ambalaj nihai imalat olmamakta son kat olarak şeffaf wrapper kullanılabilmektedir.

2.5.3 Okuyucular

Çizgikod okuyucuları, sabit pozisyonlu on-line üniteler veya üzerinde hafıza bulunan portatif sistemler olabilir. En çok kullanılan okuyucu tipleri kalem okuyucular, sabit pozisyonlu okuyucular, laser tarayıcılar, sabit pozisyonlu laser tarayıcılar ve tele-kameralardır.

1- Kalem okuyucular: Oldukça basit yapıda hafif, kullanılması kolay ve ucuz cihazlardır. Yapılarının basitliği nedeniyle temas ile 6 ila 200 cm/sn. hızlarında okuma yapabilirler. Uygulama alanlarına göre kırmızı ışık, infrared ve yüksek, orta ve düşük çözünürlüklü tipleri vardır. Güç harcamaları çok düşüktür.

2- Sabit pozisyonlu okuyucular: Kalem okuyuculara benzer yapıda fakat sabit yerlere monte edilerek kullanılan cihazlardır. 7 ile 200 cm/sn. arasında değişen hızlarda yaklaşık 12 mm. uzaklıktan çizgikod sembollerini okuyabilirler. Uygulama alanlarına göre kırmızı ışık, infrared, yüksek ve düşük çözünürlü tipleri vardır. Güç harcamaları çok düşüktür.

3- Laser tarayıcılar: Saniyede 30-40 civarında tarama ile yaklaşık 120 cm. den okuma yapabilirler. Hata oranı düşük okuyuculardır.

4- Sabit pozisyonlu laser tarayıcılar: Saniyede 500 civarında tarama ile yaklaşık 120 cm. den okuma yapabilirler. Hata oranı oldukça düşüktür.

5- Tele-kameralar: Endüstriyel uygulamalarda, örneğin, bir konveyörün üzerinden akıp giden çimento torbalarının kontrolünde kullanılır. Okuma uzaklığı 10 metreye kadar çıkabilmektedir ve saniyedeki 1000 tarama hızı ile hata oranı yok denecek kadar azdır.

2.6 Çizgikodun Yararları

Bilginin mantıksal düzendeki siyah çizgiler ve beyaz alanların kullanılmasıyla ifade edildiği çizgikod teknolojisinin başarısı basit olarak gerçekleştirilmesinde yatar, fazla masraf gerektirmeyen güvenilir bir teknolojidir.

Çizgikod sisteminin uygulandığı malların satıldıkları yerlerde fiyatlar ve diğer özelliklerin elektronik cihazlar tarafından okunması, kasa önlerinde beklemeyi kısaltır. Ayrıca, stok seviyesinin tesbitini hızlandırması, stokları eriyen malların zamanında sipariş edilebilmesi ve değişen fiyatlara anında uyum sağlanması açısından da çizgikod sistemi büyük kolaylıklar getirmektedir.

Her ülkenin dünya çapında tartışılmaz bir üretim numarası işaretleme sistemine kavuşmasıyla üreticiler dünyanın birçok yerindeki ticaret yaptıkları kuruluşlar nezdinde tek bir numaraya sahip olur. Bu sistem firmanın müşteriyile olan bilgi alışverişini kolaylaştırır ve mevcut yada yeni ürünlerinin satışındaki başarı (veya başarısızlıktan), üretim artışıdaki değişimleri hemen ve doğru haber almayı mümkün kılar.

Çizgikodu sistemleri dağıtımda ve yönetimde önemli imkanların ve faydaların ortaya çıkmasına yol açar. Tarama, bir yandan hesap ödeme ve denetim işlemlerinin hızlanması gibi dolaysız yararlarla yol açarken bir yandan da dağıtımçıların idari bilgilerini geliştirmelerini mümkün kılar. Bu sistemle stok kontrolü ve sipariş işlemleri çok daha kolay ve hatasız yapılır.

Çalışanlar çalışma koşullarının rahatlığı dolayısıyla daha az yorulur. Ayrıca bilgi işlem ve pazarlama gibi alanlarda da yeni iş imkanları oluştur. Ayrıca hızlı ve hatasız hesap ödeme işlemleri sayesinde, müşteriler de sistemden yararlanırlar. Fiyatlama ve hesaplamaadaki hataları hemen hemen yok edilir.

Çizgikod işaretlerinin kullanımı, üreticiler, toptancılar, perakendeciler ve tüketiciler arasında alış verişini kolaylaştırma, güven duygusunu geliştirme, bilgi akışını sağlama gibi birçok açıdan kolaylıklar getirmektedir.

Uluslararası standart numaralı çizgikodu sistemleri ile malın satışı, sistemin kullanıldığı her yerde ve her ülkede aynı numara ile işlem görmesini mümkün kılmaktadır. Böylece üretimden tüketim kadar giden yolda birlik sağlanmaktadır.

Ayrıca, çizgikod sistemi malların hareketi ve müşteri davranışları hakkında da büyük ölçüde veri sağlamaktadır. Böylece mevcut ve yeni ürünlerin satışı konusundaki başarı hızlı ve doğru bir şekilde ortaya çıkmakta ve satış yönetimi ve pazarlama faaliyetleri daha etkin olmaktadır.

2.7 Sonuç

Çizgikod teknolojisinin başarısı basitliği sebebiyledir ve çizgikodu sistemleri fazla masraf gerektirmeyen güvenilir bir teknolojidir.

Çizgikodu sistemlerini iki grupta inceleyebiliriz. Birincisi, uluslararası standart numaralarla birlikte ifade edilen EAN, ITF gibi çizgikodu sembolleri veya alfabeleri, ikincisi uluslararası standart numaralarla birlikte ifade edilmeyen Code 39, Codebar gibi çizgikodu sembolleri veya alfabeleridir. Çizgikodunu oluşturan çizgi ve boşluklar bir numaraya tekabül eder. Bu numaralar örneğin EAN'da herhangi bir ülkedeki bir firmanın herhangi bir mamülünü ifade etmesine karşın, Code 39'da bir PDKT sisteminde bir işgöreninin sicil nosunu, bir stok takibinde ise bir stok numarasına tekabül eder.

Çizgikodunun yapısının basit olmasına karşılık, özel donanımlı bilgisayarlar ve elektronik devre cihazları gibi karmaşık ekipmanlardan oluşması sebebiyle pahalı ve karmaşık bir sistem olması, bu iş daha başka ve daha basit yapılabilme imkanlarını araştırmaya sevk etmiştir.

BÖLÜM 3. ENDÜKTİF ÖĞRENME VE ALGORİTMALARI

3.1 Giriş

Bilgi kazanımı darboğazının aşılması için geliştirilmiş tekniklerden bir tanesi de Endüktif Öğrenme (EÖ)'dir. En genel anlamıyla EÖ, "özel örneklerden genel kurallar çıkarabilen bir tekniktir" [Forsyth, 1989; Hancox et al., 1990; Quinlan, 1988]. Geniş manada özelden genele doğru spesifik örneklerden genel kurallar çıkaran bir metod olarak tanımlanabilir.

Burada girdi, konuyla ilgili örnekler setidir.

EÖ ile bir bilgi tabanı oluşturmak için ilk iş, konuyla ilgili örnekleri toplamaktır. Burada söz konusu örnekler, bir uzman tarafından belirlenebilir veya bazı kaynaklardan (arşiv , belge gibi) gelebilir ve yahutta uzmanlık alanıyla ilgili çeşitli gözlem ve deneylerden elde edilir. Her bir örnek bir seri karakteristik değeri ve bunların ait olduğu sınıfı içerir. Örneğin mevsimler ile ilgili bir problemde temel karakteristikler *hava durumu, sıcaklık ve ağaçlar/yaprakların rengi* olabilir. Sınıflar ise mevsimler olup *ilkbahar, yaz, sonbahar, kış* olabilir. Bunların alabileceği değerler ise Hava durumu için *yağmurlu, karlı, güneşli*; sıcaklık için *orta, yüksek, normal* ve ağaçlar/yapraklar için *sarı, yeşil, yapraksız* olabilir.

Bu alanda geliştirilmiş bir çok algoritma mevcuttur. Bunlardan bazıları CLS, ID3, ID4, BCT ve AQ11 dir. Bunlardan bir tanesi de bu çalışmada kullanılan RULES-3 algoritmasıdır [Aksoy, 1993].

Bir EÖ algoritmasının çıktısı ya bir karar ağacı, veya bir kurallar setidir. Bu ise bir Uzman Sistem (US) için bilgi tabanıdır. Bu yolla kural elde etme hem daha kolay hem de daha ucuzdur.

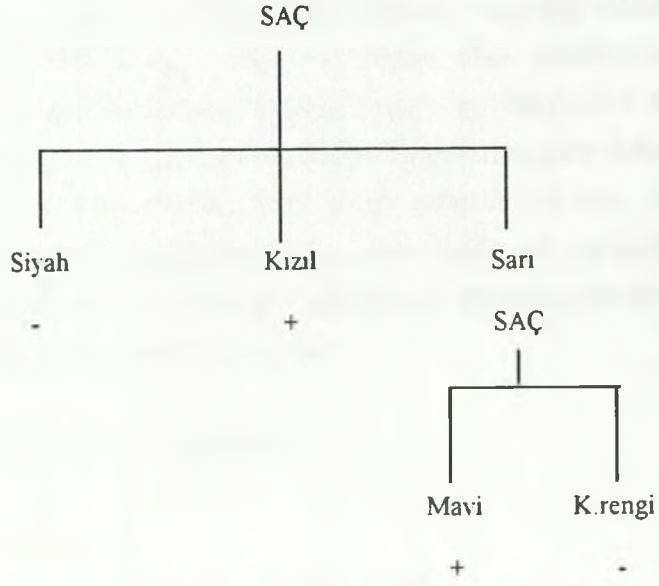
Elde edilen karar ağacı kolayca EĞER-İSE kurallarına dönüştürülebilir. Şekil3.1'de gösterilen karar ağacından çıkarılan kurallar şunlardır.

KURAL-1 : EĞER Saç = Siyah İSE Sınıf = -

KURAL-2 : EĞER Saç = Kızıl İSE Sınıf = +

KURAL-3 : EĞER Saç = Sarı AND Göz = Mavi İSE Sınıf = +

KURAL-4 : EĞER Saç = Sarı AND Göz = Kahvarengi İSE Sınıf = -



Şekil 3.1 Örnek Bir Karar Ağacı

3.2 ID3 EÖ Tekniđi

Bu teknik aslında CLS tekniđinin geliştirilmiř bir halidir. CLS bu sahada geliştirilmiř ilk algoritma olup, dokuz versiyonu mevcuttur [Hunt,E.B., Marin,J., ve Stone,P.J., 1966]. ID3, çok dallı karar ađacı oluşturulabilen bir algoritmadır. Karar ađacını oluřturmaya önce ađacın kök deđeri bulunarak başlanır. Bunu yapmak için ID3, problemi tanımlayan karakteristikler içinden, seçildiđi taktirde entropisi en düşük veya sahip olabileceđi bilgi en büyük olan karakteristiđi seçer. Her karakteristiđin entropisi hesaplanır. Entropi deđeri en düşük olan karakteristik kök olarak seçilir ve örnekler seti bu karakteristiđin deđerlerine göre dallandırılır. Yani bu deđerler sayısınca alt setlere ayrılır. řayet bu alt setlerde bulunan örneklerin hepsi aynı sınıfa aitse problem çözülmüş olur, aksi halde alt setlerin de entropileri hesaplanarak daha alt setlere ayrılır. Bu işlemler, alt setlerin hiçbirisinde farklı sınıfa ait örnekler kalmayıncaya kadar devam eder.

Bu proses řu şekilde formüle edilebilir:

S: Bir örnekler seti olsun

A: Karakteristikler setini göstereyin (A_1, A_2, \dots, A_m)

C: Örneklerin ait oldukları sınıfların bir seti olsun (C_1, C_2, \dots, C_p)

V: Her karakteristiđin deđişkenler seti olsun (V_1, V_2, \dots, V_n)

Bu durumda bir örnek

$(V_1, V_2, \dots, V_n \Rightarrow C_k)$ şeklinde olacaktır.

P_{s,c_k} her sınıfın verilen örnekler setinden elde edilen karar ađacının bir dalındaki (alt setteki) ihtimali olsun. Buna göre bir karakteristiđin entropisi (8,18)

$$E(A_i, S) = \sum_{S_j \in A_i} (S_j/S) I(S_j) \quad \text{olup, burada}$$

$$I(S) = - \sum_{k=1}^p P_{s,c_k} \text{Log}_2 P_{s,c_k} \quad \text{dir.}$$

Bu karakteristikten elde edilebilecek bilgi ise:

$$\text{Kazanç} = I(S) - E(A_i; S) \quad \text{dir.}$$

ID3, entropisi minimum veya bilgi kazancı maksimum olan karakteristiği esas olarak verilen seti ve/veya alt setleri dallandırır. Bu şekilde elde edilen karar ağacından daha sonra kurallar elde edilir.

3.3 AQ-11 EÖ Tekniği

Karar-ağacı üreten algoritmalarından farklı olarak AQ direk olarak EĞER İSE kuralları üretir. AQ verilen bir eğitim seti veya örnekler setinden değişken-değerli VL kuralları geliştiren bir algoritmadır. VL kuralları karar verme kurallarına denktir. Aşağıda bir VL kuralı verilmektedir.

$$|x \neq 6| \quad |x = 3,5| \quad V \quad |x > 7| \quad \rightarrow \quad \text{karar} = B$$

Bu VL kuralı şöyle yorumlanır ;

EĞER (x ≠ 6 VE x = 3 VEYA 5) VEYA x > 7 İSE Karar = B

AQ'da kullanılan terimler:

Seçiciler; Örnek karakteristikleri hakkında hükümlerdir (örn. saç rengi niteliği hakkındaki hükümler "siyah" ve "kırmızı"dır).

Birleştirilen kavramlar; Seçici gruplarıdır.

Tamamlayan Bir Tanımlama; Tüm pozitif örnekleri doğru olarak tanımlayan karakteristik-değer ikilisinin bir grubudur (Bir örneğe ait olan sınıf, pozitif sınıf kabul edilir, diğer tüm sınıflar negatif sınıfı oluşturur).

Uyuşan Bir Tanımlama; Negatif örneklerin hiçbirine uymayan karakteristik-değer ikilisinin bir grubudur.

Star; Seçilen sınıflanmamış bir örneğin yani çekirdeğin olası mümkün tüm alternatif tanımlamalarıdır.

Çekirdek; Sınıflanmamış örneği kapsar ve negatif sınıf içindeki hiçbir örnek kapsamına alınmaz.

E Karşı Genişletme Prosesi; Negatif set içindeki örneklerin aynı niteliği için değerler setine karşı çekirdek içinde görülen bir niteliğinin değerini genişletmek için AQ'da kullanılan bir prosestir.

AQ Algoritması, AQ'nun genel ve basit bir versiyonu aşağıdaki gibi özetlenmiştir.

- Adım 1. Sınıflanmamış bir örnek seç. Seçilen örneğin yani çekirdeğin sınıfını pozitif sınıf olarak, diğer örneklerin sınıfını negatif sınıf olarak kabul et.
- Adım 2. Örnek için bir star üretmek amacıyla E Karşı Genişletme Prosesini tatbik et.
- Adım 3. Adım 2'de bulunan bir stardan, Uyuşum ve Tamamlama değerlerinden yararlanılarak bir tanımlama (D) seç.
- Adım 4. Şayet D pozitif tüm örnekleri içeriyorsa, Adım(6) ya git.
- Adım 5. İçermiyorsa, sadece D tarafından içerilmeyen örnekleri kapsamına almak için pozitif örnekler setini azalt.
- Adım 6. Üretilmiş tüm tanımlamaları bir VL kural formuna çevir (VL kuralı pozitif sınıf içindeki örneklerin tümünü çevirebilir) ve sınıflanmamış örnekler setinden bu örnekleri ayırır. Her sınıf için bir tek VL kuralı üretilerek, temel algoritma her sınıf için bir kez gerçekleştirilir.

3.4 RULES-3 (Rules Extraction System-3) EÖ Tekniği

Şimdiye kadar geliştirilmiş EÖ algoritmaları içerisinde en basit ve en anlaşılır olanı RULES-3 algoritmasıdır.

Basit gösterimi sebebiyle ID3 belki de şimdiye kadar geliştirilen EÖ algoritmaları içinde en tanınmış olanıdır. Fakat RULES-3 bu algoritmadan da daha basit ve daha anlaşılır bir algoritmadır. RULES-3, ID3'nin başarısız kaldığı iki önemli dezavantajının üstesinden gelmiş bir algoritmadır. Birincisi, gereksiz şartlar içeren üretim kurallarına sahip olabilmesi ve ikincisi ise algoritmanın icrası esnasında bilgisayarın ana rastgele giriş hafızasında saklı olan örnek setinin çok büyük olması

durumlarında “windowing” adı verilen setin seçilen belli bir kısmını kullanmak ile sınırlı olmamasıdır. Windowing her zaman için tüm örnek setini temsil edemeyebilir. ve gereksiz şartlarla uğraşmak zaman açısından kayıplara neden olabilmektedir. Ayrıca RULES-3 diğer EÖ algoritmaları ile kıyas edildiğinde daha başka üstünlüklere de sahiptir.

RULES-3 algoritması aşağıdaki adımlardan oluşur:

- Adım 1. Sayısal değerler içeren karakteristikler için aralıklar tanımla.
- Adım 2. Her kural için minimum şart sayısını (N_{Cmin}) gir.
- Adım 3. Rastgele bir örnek seç.
- Adım 4. $N_C = N_{Cmin} - 1$.
- Adım 5. Şayet $N_C < N_a$ ise $N_C = N_C + 1$ yap.
- Adım 6. Eldeki örnekte kullanılmış değerleri al.
- Adım 7. Bu değerlerin N_C li kombinasyonlarını bul.
- Adım 8. Şayet bu kombinasyonlardan en az bir tanesi sadece bir sınıfta geçiyorsa, bunu kural haline getir.
Aksi halde Adım 5 ' e git.
- Adım 9. Şayet Adım 8' de birden fazla kural elde edildiyse, bunların içinden, örnekler setinde en fazla örneği sınıflandıran kuralı seç.
- Adım 10. Seçilmiş kuralı kullanarak sınıflandırılabilen örnekleri ayır.
- Adım 11. Şayet geriye sınıflandırılmamış örnek kalmadıysa DUR
Aksi halde Adım 3'e git.

Burada N_C , bir kuraldaki şart sayısını, N_a ise bir örnekteki karakteristik sayısını göstermektedir. Şayet örnekler setinde sayısal değerler içeren karakteristikler varsa Adım 1'de bunlar için aralıklar tanımlanır. Dolayısıyla aslında nominal değerler içeren karakteristikler için kullanılan EÖ ile sayısal değerli karakteristikler için de kurallar çıkarılabilir. Bu işlem şöyle yapılır: Söz konusu karakteristikler için, verilen örneklerde kullanılan minimum ve maksimum değerler bulunur. Daha sonra kullanıcının isteğine göre bu iki değer arasında eşit aralıklar tanımlanır. Her aralığa bir etiket verilerek örnekler seti bu etiketlere göre yeniden düzenlenir. Bir kural en az bir, en fazla N_a yani toplam karakteristik sayısı kadar şart bulundurabilir. Adım 2'de bir kuralda bulunması istenilen minimum şart sayısı girilir. Adım 3'te, örnekler setinden, daha önce kullanılmamış bir örnek rastgele seçilir. Adım 4 ve 5'te kullanılan ifadelerle eldeki örnek için şart sayısı düzenlenir. Bu sayı başlangıçta N_{Cmin} dur. Adım 6'da eldeki örnekte kullanılan N_a tane değer bir diziye atanır. Adım 7'de bu dizinin N_C li kombinasyonları bulunur. Adım 8'de bu kombinasyonların herbirisinin kural olup

olamayacağı test edilir. Şayet bir kombinasyon bütün örnekler setinde sadece bir sınıfta geçiyorsa bu bir kural haline getirilir. Eğer hiç kural bulunamıyorsa şart sayısı bir arttırılır. Bu işlem en az bir kural bulununcaya kadar devam eder. Adım 9'da; Şayet Adım 8'de birden fazla kural bulunduyorsa, bunların içerisinde, verilen örnekler setinde en fazla örneği sınıflandırabilen bir tanesi seçilir. Adım 10'da bu kural kullanılarak, kendisi için daha önce kural çıkarılmamış tüm örneklerden sınıflandırılabilenler ayklanır. Adım 11'de şayet geriye sınıflandırılmamış örnek kalmadıysa algoritma durur, aksi halde Adım 3'e gidilerek işlemler tekrar edilir. Sonuçta elde edilen tüm kurallar bir araya getirilerek bir bilgi tabanı elde edilir. Bu bilgi tabanı eldeki örnekler setinin genelleştirilmiş halidir.

3.4.1 RULES-3 İle Bilgi Tabanını Oluşturmak İçin Bir Örnek Problem

RULES-3 ile bir bilgi tabanının nasıl oluşturulduğunu görmek için aşağıdaki örneği inceleyelim. Bu problemde bir arabanın hacmi, yakıt cinsi ve maksimum hızına göre nasıl bir ivmeye sahip olacağının tesbiti söz konusudur. Problemde sekiz tane örnek verilmiştir. Her bir örnek, yukarıda belirttiğimiz üç farklı karakteristiğinin birer değerine sahiptir. Bu karakteristiklerden arabanın hacmi {geniş, kompakt, orta}; yakıt {motorin, benzin, propan} ve maksimum hız {yüksek, orta, düşük} değerlerine sahiptir. Buna göre ivme {çok-iyi, iyi veya orta} olabilir. Örnekler seti Tablo 3.1'de verilmiştir. Bu örnekler seti [Yasdi, 1991] den uyarlanmıştır.

Tablo 3.1 Araba-ivmesi Problemi için Örnekler Seti.

Örnek No	Hacim	Yakıt	Maksimum Hız	Ivme
1	geniş	motorin	yüksek	iyi
2	geniş	propan	yüksek	iyi
3	kompakt	benzin	yüksek	çok-iyi
4	geniş	benzin	yüksek	çok-iyi
5	orta	motorin	düşük	iyi
6	kompakt	benzin	düşük	iyi
7	orta	benzin	orta	çok-iyi
8	orta	motorin	orta	orta

RULES-3 ile bu örnekler setinden genel kuralların (bilgi tabanının) elde edilişi, adım adım şu şekilde olmaktadır: Birinci iterasyonda;

Problemde sayısal değerlere sahip karakteristik olmadığından, Adım 1'de hiç bir karakteristik için aralık tanımlanmamıştır.

Adım 2'de, kurallar için minimum şart sayısı 1 olarak kararlaştırılmıştır.

Adım 3-8 kullanılarak 1 no.lu örnek'ten iki kural çıkarılabilir. Bunlar:

1. *Eğer hacim geniş ve yakıt motorin ise ivme iyi'dir.*
2. *Eğer yakıt motorin ve maksimum hız yüksek ise ivme iyi'dir.*

Bu kurallardan her biri, verilmiş örnekler setinde en fazla birer örnek sınıflandırabilmektedir. Bunlardan herhangi birisi rastgele seçilebilir. 1 no.lu kural seçildiği takdirde bununla 1 no.lu örnek sınıflandırılabilir. Bu örnek setten çıkarılırsa geriye 7 tane örnek kalır.

İkinci İterasyonda 2 no.lu örnekten bir kural çıkarılabilir. Bu da:

Eğer yakıt propan ise ivme iyidir.

Bu kural ile sadece 2 no.lu örnek sınıflandırılabilir. Bu örnek setten çıkarıldığında geriye 6 tane örnek kalır.

Üçüncü iterasyonda 3 No.lu örnekten 2 kural çıkarılabilir. Bunlar:

1. *Eğer hacim kompakt ve maksimum hız yüksek ise ivme çok-iyi'dir.*
2. *Eğer yakıt benzin ve maksimum hız yüksek ise ivme çok-iyi'dir.*

Birinci kural bir örnek(örnek 3), ikinci kural ise iki örnek (örnek 3 ve 4) sınıflandırabilirler. Dolayısıyla daha genel olduğundan ikinci kural seçilir. Geriye işlenmemiş 4 tane örnek kalır.

Dördüncü iterasyonda 5 no lu örnekten aşağıdaki kural çıkarılabilir:

Eğer maksimum hız düşük ise ivme iyi'dir.

Bu kural ile 5 ve 6 no.lu örnekler sınıflandırılabilir. Bunlar setten çıkarılırsa geriye 2 tane örnek kalır.

Beşinci iterasyonda 7 no.lu örnekten iki kural çıkarılabilir. Bunlar:

1. *Eğer hacim orta ve yakıt benzin ise ivme çokiyi'dir.*
2. *Eğer yakıt benzin ve maksimum hız orta ise ivme çok-iyi'dir.*

Bu kurallardan her birisi sadece birer örnek sınıflandırabilir. İlk kural seçildiğinde 7 no.lu örnek sınıflandırılabilir. Bu da setten çıkarılırsa geriye sadece işlenmemiş 1 örnek kalır.

Altıncı iterasyonda 8 no.lu örnekten aşağıdaki kural çıkarılır:

- Eğer yakıt motorin ve maksimum hız orta ise ivme normal'dir.*

Bu kural 8 no.lu örneği sınıflandırabilir. Bu örnek te setten çıkarıldığında geriye işlenmemiş örnek kalmaz. Dolayısıyla kural çıkarma işlemi son bulmuş olur. Bu işlemler Tablo 3.2. de özetlenmiştir.

Tablo 3.2 Araba-ivmesi Problemi için Endüktif Kural Çıkarma İşleminin Özeti.

İterasyon	kullanılan örnek	çıkarılan kural sayısı	Sınıflandırılabilen örnekler	Sınıflandırılmayan örnekler
1	1	2	1	2,3,4,5,6,7,8
2	2	1	2	3,4,5,6,7,8
3	3	2	3,4	5,6,7,8
4	5	1	5,6	7,8
5	7	2	7	8
6	8	1	8	8

Bu kurallar bir araya getirildiğinde söz konusu problem için bir kurallar seti (başka bir deyişle bilgi tabanı) elde edilmiş olur. Bu problem için bilgi tabanı Tablo 3.3 te verilmiştir.

Tablo 3.3 Araba-ivmesi problemi için otomatik oluşturulan veri tabanı

1. *Eğer hacim geniş ve yakıt motorin ise ivme iyi'dir.*
2. *Eğer yakıt propan ise ivme iyidir.*
3. *Eğer hacim kompakt ve maksimum hız yüksek ise ivme çok-iyi'dir.*
4. *Eğer maksimum hız düşük ise ivme iyi'dir.*
5. *Eğer hacim orta ve yakıt benzin ise ivme çokiyi'dir.*
6. *Eğer yakıt motorin ve maksimum hız orta ise ivme normal'dir.*

3.5 EÖ Uygulamaları

EÖ algoritmaları belirli bir alana bağımlı değil, örneklerle ifade edilebilen her alanda kullanılabilirler. Bu algoritmalar model tanıma veya sınıflandırmanın söz konusu olduğu herhangi bir işte kullanılabilir [Lui ve White, 1991].

Algoritmaların başarıyla uygulandığı birkaç faaliyet alanı şunlardır;

1. Tıbbi uygulamalar [Michalski, 1986].
2. Yatırım değerlendirme [Race, ve Thomas, 1988],
3. Uzay mekiği ön motoru için test verisi analizlerine karar verme kurallarının çıkarılması [Modesitt, 1987],
4. Eğilme kuvveti altında çelik üyelerin fikirsel tasarımı için tecrübi karar verme kurallarının üretimi [Forsyth, 1989],
5. Toprak sınıflama [Dale, 1989],
6. Stok kontrol [Thorpe, 1989],
7. Yazılım araştırma analizleri [Selby, ve Porter, 1988],
8. Metalurjik uygulamalar [Harrington, 1989],

9. Askeri uygulamalar [Lirov, 1989],
10. Dinamik sistem modelleme [Batur, 1991],
11. Motor hatasını bulma [Ke, ve Ali, 1989], ve
12. Kredi kartı uygulamaları [Carter, ve Catlett, 1987].

3.6 SONUÇ

EÖ , varolan bilginin genelleştirilmesi için önemli bir yardımcıdır. EÖ teknikleri, insana ait mantıklı düşünme prosedürlerine uygun ve tutarlı karar vermek için açık ve modül şeklinde bir format sağlar.

EÖ ile elde edilen kurallar olduğu gibi bir US'de kullanılmaya elverişlidir. 1987 den evvel geliştirilen iki önemli uzman sistem (BMT ve GASOİL) otomatik EÖ kullanılarak geliştirilmiştir [Lui ve White, 1991].

BÖLÜM 4. ÇİZGİKODU TANIMA İÇİN ALTERNATİF BİR TEKNİK

4.1 Giriş

Çizgikodları, basit yapıları sayesinde çok kısa zaman süresi içerisinde bilgisayarların da yardımıyla istenilen bilginin elde edilmesinde çok büyük kolaylıklar sağlamaktadır. Basit yapılarına karşın bazı pahalı ve özel elektronik ekipman ve standart hale getirilmiş çizgikodu alfabeleri ile bütünleştirilerek tanıma işlemini gerçekleştirebilmektedir. Çizgikodun okunabilmesi, yani bu şifrenin çözülebilmesi için sadece standart hale getirilmiş belli bir alfabe ile ifade edilen çizgikodu etiketi, okuyucu ve bilgisayar haricinde bazen bilgisayarlarla bağlantılı özel donanımlı cihazlar da gerekli olmaktadır. Bu özel donanımlar ve özel programlar yardımıyla işletmelerde mevcut sisteme ilave edilecek çizgikodu okuyucu terminallerinden veya mevcut terminallere bağlanacak çizgikod okuyucularından yararlanılabilir.

Standart hale getirilmiş çizgikodu alfabelerinin etiketlerinin hazırlanması ve basımı için özel cihazlar gerekmekte ve boyutları ise belli standartlara ve toleranslara uygun olmak zorundadır.

Çizgikodu bir okuyucu tarafından çizgiler boydan boya dik kesilecek şekilde belli bir hat içerisinde taranarak tanıma işlemi gerçekleştirilebilmektedir. Şayet bu hat üzerinde bir deformasyon var ise çizgikodu okunamamaktadır.

Çizgikodu alfabeleri iki grupta incelenebilirler. Birincisi EAN, UPC ve ITF gibi uluslararası kimliğe sahip ve bir ülkedeki bir firmanın mamülün kimliğini tanıtan bir numaraya sahip alfabeler ve ikincisi Code 39, Codebar gibi üretim ve stok takibi veya PDKT gibi spesifik alanlarda kullanılan çizgikodunun herhangi bir numarayı, örneğin PDKT için işgörenin sicil nosu veya stok takibi için stok nosu, ifade ettiği alfabelerdir. Aslında çizgikodunu meydana getiren çizgiler ve boşluklar bir numarayı ifade eder. Örneğin EAN13'te iki boşluk ve iki çizgi bir rakamı, tüm çizgikodunu oluşturan çizgi ve boşluklar ise ilgili mamülün EAN numarasını temsil eder.

Sistem bir bütün olarak düşünüldüğünde farklı bir alfabe kullanılmak ve YZ'nin EÖ tekniğinden faydalanılmak suretiyle daha basit, daha hızlı ve daha fazla avantajlara sahip alternatif bir çizgikodu tanıma tekniği oluşturmak mümkün görülmektedir.

Bu bölümde bazı dezavantajların giderildiği daha da geliştirilmiş alternatif bir teknik sunulmaktadır. Bu tekniğin en önemli özelliği, okuyucunun tüm çizgileri taramak zorunda olmaması sebebiyle gerek bilgiyi saklamak ve gerek kullanmak için hem hafıza hem de zamandan tasarruf edilmesidir.

4.2 Alternatif Çizgi Kodu Tanıma Tekniği

Alternatif teknik, yazılım ve donanım olmak üzere iki kısımda incelenebilir. Yazılım kısmının tasarımın en önemli kısmını RULES-3 EÖ algoritması ile oluşturulan bilgi tabanı teşkil etmektedir. Donanım kısmı ise bir PC bilgisayar, bir okuyucu ve çizgikodu etiketinden ibaret olacak şekilde düşünülmüştür.

EÖ algoritmaları, basit olarak örnekler arasında bir ilişki kurup örnekleri birbirinden ayıran temel farklılıkları bularak örnekleri kolayca sınıflandırma ve tanıma işlemlerini başarıyla yapabilmektedir. Bu çalışmada, ilk iş olarak EK-1'de gösterilen 30 adet kitabın her biri için benzetim tekniği kullanılmak suretiyle Şekil 4.3'te gösterilen çizgikodları üretilmiştir. Her bir çizgikodu EK-1'de verilen ilgili verileri temsil etmektedir. Daha önce RULES-3 algoritmasında anlatıldığı gibi EÖ tekniği ile bir bilgi tabanı oluşturabilmek için ilk önce örnekler setinin hazırlanması gerekmektedir. Örnekler seti, benzetim tekniğiyle üretilen çizgikodlarından oluşturulmuştur. Her bir örneğin karakteristik değerlerini çizgikodunu oluşturan çizgilerin piksel olarak kalınlık değerleri, temel karakteristiklerini çizgikodunu oluşturan her bir çizgi, ve sınıfları ise çizgikodları temsil etmektedir. Bu örnekler seti RULES-3 EÖ algoritmasında proses edilmek suretiyle Tablo 4.3'te gösterilen her bir çizgikodunun tanınmasında çok büyük kolaylıklar sağlayacak olan bilgi tabanı elde edilir. EÖ Yapay Zeka'nın (YZ) dalı olması sebebiyle, YZ'nin EÖ dalını dolayısıyla da RULES-3'ü ilgilendiren bazı temel kavramlarını açıklamakta yarar vardır. EÖ algoritmaları muhakame mekanizması olarak endüktif muhakame mekanizmasını kullanmaktadır. Yani spesifik örneklerden tümevarılmak suretiyle genel hükümler çıkarılmaktadır. Öğrenme stratejisi olarak öğretmenli öğrenme stratejisini kullanır. Bu stratejide bir öğretmene ihtiyaç duyulur. Öğretmen, istenilen sonuç çıktısını bilir ve çıkarılan kurallar tüm test setinin tamamını belirli bir hata payı ile tanıyınca kadar sisteme müdahale ederek girdileri değiştirir. Daha doğrusu istenilen sonuç elde edilinceye kadar deneme yanılma yöntemi ile öğretmen sisteme müdahale eder. Bir diğer kavram kural çıkarım mekanizması olarak ileriye doğru zincir mekanizmasını kullanırlar. Bu çıkarım mekanizmasında gerçekler ile şartlar uyumlu olduğunda şartlar tatmin edilince kural çıkarılmaktadır.

Teknik bilgi tabanı oluşturulduktan sonra, bir okuyucu ile bütünleştirilir ve bilgisayar ekranı üzerinde benzetilmiş çizgikodlarını tanıma işlemi yapılabilir hale gelmiş olur. Bunun için optik okuyucu benzetilmiş çizgikodlarını enine tarayarak her bir çizgikodunun kalınlık bilgileri sayısal olarak bilgisayara iletilir. Bilgisayar bu bilgileri bir kayıt dosyasında saklar. Bu kayıt dosyasındaki bilgiler bilgi tabanındaki kurallara karşılaştırılarak hangi çizgikodu olduğu tanınır.

Mevcut sistemlerde okuyucu çizgikodu tanımak için çizgikoduna ait tüm çizgileri ve boşlukların tamamını tarar ve algılanan çizgiler ve boşluklar bir elektronik devre ile dijital sinyal olarak ilgili bilgiyi bilgisayarlara iletir. Bilgisayara iletilen tüm çizgikodu bilgileri bilgisayar hafızasında saklanır. Satış mağazasındaki gibi bilgisayar belleğinden ilgili bilgiyi arar ve kasaya iletir. Alternatif teknik tüm çizgikodunu baştan sona taramaz sadece çizgileri tarar boşlukları taramaz. Alternatif teknik hem zaman, hem de hafıza problemini nispeten azaltacaktır.

Okuyucuların mevcut çizgikodu etiketlerini okuyabilmesi için bazı şartların sağlanması mecburidir. Birincisi, Çizgikodu etiketlerinin anma boyutları standart ölçülerde, belli toleranslar içinde ve belli bir alfabe ile hazırlanması gereklidir, ikincisi ise okuyucuların şifresi çözülecek alfabe veya alfabelere programlı olmaları gereklidir. Yani her okuyucu her alfabeyi okuyamaz sadece programlandığı alfabe veya alfabeleri okuyabilir. Alternatif teknik alfabesinin basitliği sayesinde okuyucu, her alfabeyi okuyabilme imkanına sahip olacaktır.

Çizgikodu bir okuyucu tarafından çizgiler boydan boya dik kesilecek şekilde belli bir hat içerisinde taranarak tanıma işlemi gerçekleştirilebilmektedir. Şayet bu hat üzerinde bir deformasyon var ise Çizgikodu okunamamaktadır. Alternatif teknikte tüm çizgiler taranmayacağı için Çizgikodun tanıma ihtimali daha fazla olacaktır.

Mevcutlarından farklı olarak yazılım kısmı iki önemli özelliğe sahiptir. Bu EÖ tekniğinin özelliğinden kaynaklanmaktadır. Birincisi, örneklerden faydalanılarak bilgi tabanı oluşturulduğu için yazılım çok daha kolay ve basit, ve çok daha kısa süre içerisinde yazılabilmektedir. İkincisi ise, örnek sayısının artırılması veya azaltılması durumunda bilgi tabanının yeniden oluşturulması zorunluluğu sözkonusudur.

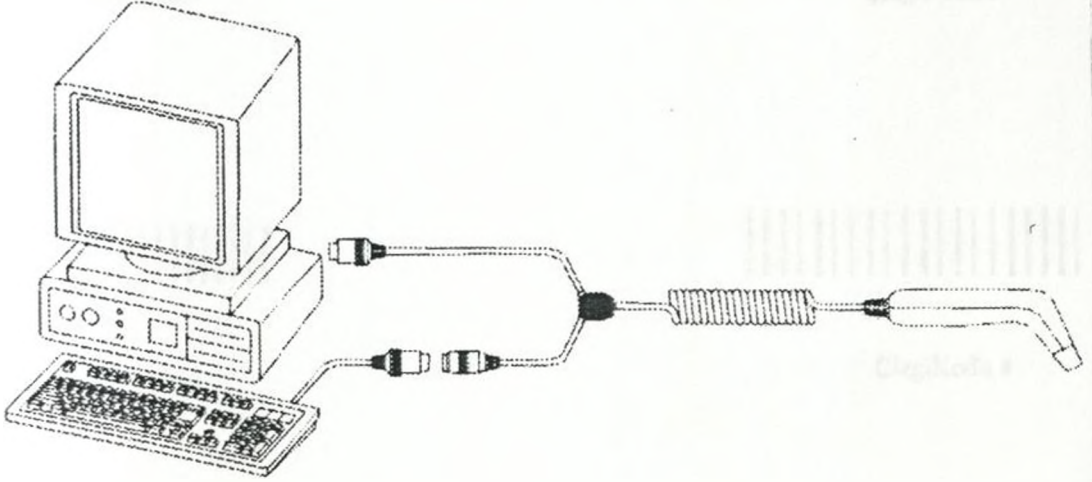
4.2.1 Alternatif Çizgi Kodu Tanıma Tekniğinin Donanımı

Tekniğin donanım kısmı iki seçenektir. Birinci seçenek Şekil 4.1’de görüldüğü üzere bir PC bilgisayar veya yazarkasa, bir okuyucu ve bir çizgikodu etiketinden ibarettir. Okuyucu, sistem içinde klavye görevi görmektedir. İkinci seçenek ise Şekil 4.2’de görüldüğü üzere bu sisteme bir de klavye eklenmesidir. Bu sayede bilgi girişleri hem okuyucu hem de klavyeden faydalanılarak gerçekleştirme seçeneğinede sahip olabiliriz. Okuyucu çizgikodunu okuyarak ilgili kitap kodu ve verilerini verebildiği gibi şayet alternatif tekniğin alfabetini karakterize eden piksel olarak çizgikodu kalınlık değerleri klavye ile girilmek suretiyle yine ilgili kitap kodu ve verileri elde edilebilir.

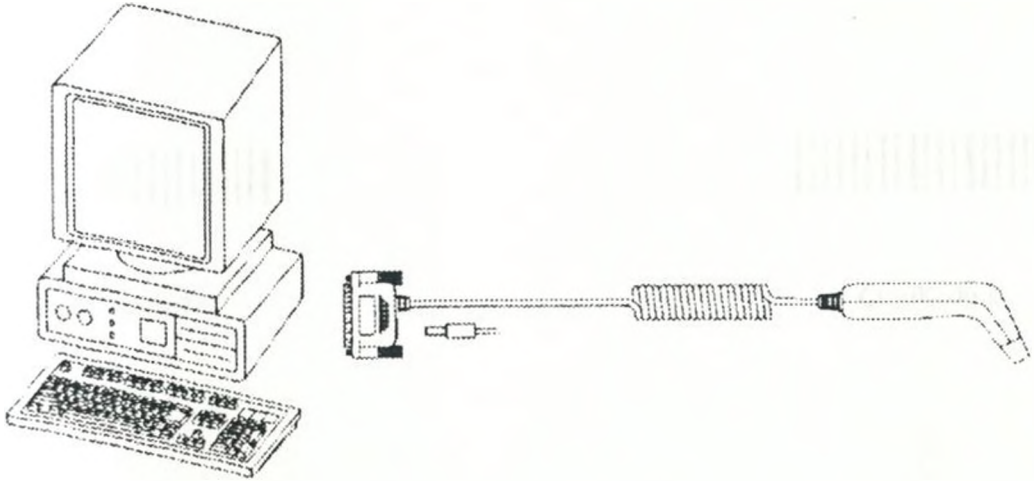
4.2.2 Alternatif Çizgi Kodu Tanıma Tekniğinin Anma boyutları, Etiket Toleransları ve basımı

Mevcut sistemlerdeki gibi okuyucunun şifreyi çözebilmesi için çizgikodu etiketinin anma boyutlarının ve çizgi kalınlıklarının belli standartlar dahilinde olması zorunluluğu yoktur. Okuma esnasında boşluk değerleri bir şey ifade etmediğinden çizgikodları sadece çizgilerden oluşur. Çizgi sayısı bu çalışmada yirmi olarak alınmıştır. Bu sayı ihtiyaca binaen arttırılabilir. Çizgikodunun anma boyutları, uygulama alanına bağlı olarak nasıl uygun ise öyle alınabilir. Bu çalışmada çizgilerin kalınlık değerleri ise 1 ila 5 piksel arasında rastgele olarak değişmektedir. Bu değerler de uygulama alanına bağlı olarak ihtiyaca binaen değiştirilebilir. Bu çalışmada herhangi bir standarta ve çeşitli toleranslara uyulmasına gerek kalınmamasının sebebi, bu çalışmada kullanılan alfabenin basitliği ve çizgikodlarını oluşturan çizgilerin kalınlık değerlerinin piksel olarak ifade edilmesidir. Mevcut alfabelerde çizgikodunu oluşturan çizgi ve boşluk grupları bir numaraya takabül ediyordu diğer bir deyişle bir numara çizgikodu haline getirilip okuyucu çizgikodunu tanıyabiliyordu. Alternatif tekniğin kalınlığı 1 ila 5 piksel arasında rastgele olarak değişen 20 adet çizgiden meydana gelen çizgikodu etiketi bir numaraya tekabül etmez veya bu kalınlık değerleri bir numarayı ifade etmez, sadece ilişkilendirildiği kitap kodunu ve verilerini temsil eder.

Bu çalışmada 30 adet kitabın kod numarasını ve verilerini temsil eden kalınlığı 1 ila 5 piksel arasında değişen 20 adet çizgiden oluşan 30 adet çizgikodu rastgele olarak üretilmiştir. Üretilen bu çizgikodları Şekil 4.3’te gösterildiği üzere bir lazer yazıcıdan çıktıları alınarak etiket olarak kullanılmaya hazır hale gelebilir.



Şekil 4.1 Alternatif sistemin donanımı (Seçenek 1)



Şekil 4.2 Alternatif Sistemin donanımı (Seçenek 2)



ÇizgiKodu 1



ÇizgiKodu 2



ÇizgiKodu 3



ÇizgiKodu 4



ÇizgiKodu 5



ÇizgiKodu 6



ÇizgiKodu 7



ÇizgiKodu 8

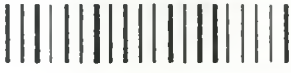


ÇizgiKodu 9



ÇizgiKodu 10

Şekil 4.3 Alternatif Çizgikodu Tanıma Sisteminin Simülasyon Tekniği ile üretilmiş Çizgikodları (Çizgikodu1-10)



ÇizgiKodu 11



ÇizgiKodu 12



ÇizgiKodu 13



ÇizgiKodu 14



ÇizgiKodu 15



ÇizgiKodu 16



ÇizgiKodu 17



ÇizgiKodu 18



ÇizgiKodu 19



ÇizgiKodu 20

Şekil 4.3 Alternatif Çizgikodu Tanıma Sisteminin Simülasyon Tekniği ile üretilmiş Çizgikodları (Çizgikodu11-20)



ÇizgiKodu 21



ÇizgiKodu 22



ÇizgiKodu 23



ÇizgiKodu 24



ÇizgiKodu 25



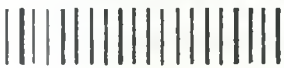
ÇizgiKodu 26



ÇizgiKodu 27



ÇizgiKodu 28



ÇizgiKodu 29



ÇizgiKodu 30

Şekil 4.3 Alternatif Çizgikodu Tanıma Sisteminin Simülasyon Tekniği ile üretilmiş Çizgikodları (Çizgikodu21-30)

4.2.3 Alternatif Çizgi Kodu Tanıma Tekniğinin Alfabeti

Bu çalışmada anma boyutlarında ve çizgi kalınlıklarında herhangi bir standarta ve toleransa gerek kalmamasının sebebi bu çalışmada kullanılan alfabenin basitliğinden kaynaklanmaktadır. Tekniğin alfabeti çizgikodlarını oluşturan çizgilerin sadece kalınlık değerlerinden oluşan bir vektördür.

4.2.4 Alternatif Çizgi Kodu Tanıma Tekniğinin Okuyucusu

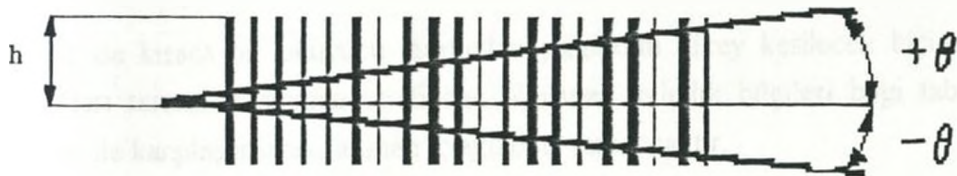
Şekil 4.4'te gösterildiği üzere çizgikodları, çizgileri dikey kesilecek şekilde artı ve eksi olarak bir θ açısı arasında kalan bölge okuyucu tarafından taranarak okunacaktır. Şekil 4.4'te gösterilen h yüksekliğinin yeri önemli değildir. Ancak θ açısı arasında kalan bölgenin okuyucunun çizgikodunu okuma ihtimalini azaltacak yükseklikte olmamasına dikkat edilmelidir.

Çizgikodunu okuyan okuyucunun belli bir alfabe veya alfabeleri okumak üzere programlanmasına gerek yoktur. Aynı şekilde çizgikodunun anma boyutları belli standartlarda ve belli bir alfabe ile ifade edilmesine gerek kalmaksızın okuyucu, artı ve eksi θ açısı arasında kalan bölge sadece çizgikodunu oluşturan kalınlıkların piksel ve renk yoğunluğu değerlerini okumak suretiyle tarar. Çizgikodu tanıma işlemi çizgikodunu oluşturan tüm çizgileri taramaz. Tarama esnasında her bir çizginin kalınlık piksel değerlerine bağlı olarak bilgi tabanına bakılır ve bir kural bulunursa çizgikodu tanınır. Örneğin Okuyucu h yüksekliğinde tarama yaparken Çizgi1=4, Çizgi2=3, Çizgi3=3 ve Çizgi4=2 bulunursa bu bilgi tabanındaki Çizgi1=4 VE Çizgi4=2 ise Çizgikodu=Kod1 kuralı bulunur. Böylece çizgikodu tanıma işlemi sona erer ve geri kalan çizgiler taranmaz.

Şayet θ açısı arasında kalan tanıma alanı içinde bulunan çizgikodu çizgileri üzerinde bir deformasyon varsa, çizgikodu tanıma işlemi gerçekleşmeyebilir. Alternatif teknikte, Şekil 4.4'e dikkat edersek okuyucunun tanıma alanı içinde bulunan sadece Çizgi1 ve/veya Çizgi4'te bir deformasyon söz konusuysa okuyucu çizgikodunu tanıyamayacaktır. Halbuki mevcut sistemlerde, tarama alanı içinde bulunan her hangi bir çizgide deformasyon varsa, çizgikodu okunamayacaktır.

4.3 Alternatif Çizgi Kodu Tanıma Tekniğinin Yazılımı

Bilgi tabanı otomatik oluşturulduğu için yazılım çok daha kolay ve basit , ve çok daha kısa süre içerisinde yazılabilmektedir.



Şekil 4.4 Çizgikodlu için alternatif sistemin okuma alanı

EK-1'de verilen 30 adet kitabın herbirinin kitap kodu, yazarı, kitap adı, sayfa adedi, basım yılı, basım yeri, hangi ülkede basıldığı verileri programımızın veritabanını oluşturmaktadır. Bu verileri rastgele olarak üretilmiş 1 ila 5 birim kalınlığa sahip yirmi adet çizgi temsil etmektedir. Tablo 4.1'de bu veriler ve temsil ettikleri çizgikodları örnekler seti olarak gösterilmektedir.

Yazılımın adımları çok kısa olarak şöyledir; *Barkod* ana menüsü altındaki *BarkodÜret* menüsü seçilerek benzetim tekniği kullanılarak herbirinin kalınlıkları rastgele olarak değişen yirmi adet çizgiden oluşan 30 adet çizgikodu üretilerek bir kayıt dosyasında üretilen bu çizgikodlarının herbirinin kalınlık değerleri kaydedilir. RULES-3 EÖ algoritması ile tekniğin bilgi tabanı oluşturulur. Benzetilmiş sistemde *Barkod* ana menüsü altındaki *BarkodTanı* menüsü seçilerek rastgele herhangi bir çizgikodunun kalınlık bilgileri dosyadan okutularak bilgi tabanı taranır ve rastgele çağrılan bu çizgikodunun hangi numaralı çizgikodu olduğu tanınır. Ayrıca bu kod numaralı kitabın tüm verileri veritabanı okunarak ekrana getirilir. Yazılım detaylı olarak EK-2'de liste halinde verilmektedir.

Pratikte ise kısaca bir okuyucu tarafından çizgikodu dikey kesilecek biçimde çizgi kalınlıkları taranır. Okuyucu tarafından algılanan kalınlık bilgileri bilgi tabanındaki kurallar ile karşılaştırılarak aranan çizgikodu tanınacaktır.

Tekniğin yazılımının tasarımı, birincisi ve en önemlisi RULES-3 EÖ algoritması ile bilgi tabanının oluşturulması, ve ikincisi ise menülerinde yardımıyla bilgi tabanı ile etkileşimli olarak çizgikodlarının tanınması olmak üzere iki kısımda incelenebilir.

4.3.1 Uygulamada Kullanılan Örnekler Setinin Elde Edilmesi

Daha önce de belirtildiği gibi bir EÖ algoritmasının girdisi belirli bir formattaki bir örnekler setidir. Dolayısıyla bilgi tabanını oluşturabilmek için kullanılacak karakteristikler, Karakteristik değerleri ve sınıfların belirlenmesi gerekir. Bu uygulamada bir çizgikodunu temsil eden her bir çizgi, bir karakteristik olarak kabul edilmiştir. Çizgilerin kalınlıkları ise karakteristiklerin değerleridir. Sınıf ise her bir çizgikoduna karşılık gelen KOD'tur.

Çalışmamızda her biri 1-5 piksel kalınlığında rastgele değişen 20 çizgiden oluşan 30 adet çizgikodu rastgele olarak üretilmiştir. Buna göre her bir örnek 20 karakteristik değeri ve bunların ait olduğu sınıfı (KOD'u) gösteren bir vektördür.

Örneğin:

4,3,3,2,2,4,1,4,5,4,1,3,5,4,2,5,5,1,5,2,KOD1

Bir örnek vektörüdür. Bu vektörün nasıl elde edildiğini şöyle açıklayabiliriz:

Önce bilgisayarda rastgele çizgi kalınlıklarına sahip aşağıdaki çizgikod çizdirilmiştir.



Daha sonra bu KOD'un her bir çizgisinin kalınlığı bulunmuştur. Bu kalınlıklar soldan sağa her çizginin kaç piksel olduğunun test edilmesi demektir. Her bir kalınlık bir karakteristiğin değeri olarak kabul edilmiştir. Sınıf ise KOD1'dir.

Benzer şekilde 30 çizgikodu üretilmiş ve 30 adet örnek elde edilmiştir. Bu örnekler Tablo 4.1'de gösterilmektedir.

4.3.2 Bilgi tabanının Oluşturulması

Daha öncede bahsedildiği üzere Şekil 4.3'de görülen örnek çizgikodları için 30 tane vektör elde edilmiştir. Tablo 4.1'de gösterilmekte olan bu vektörler RULES-3 için girdidir.

Bu örnekler setine RULES-3 şu şekilde uygulanmıştır:

Çizgi kalınlıkları her kod için sabit olduğundan karakteristikler için aralık tanımlanmamıştır.

Her kural için minimum şart sayısı 1 alınmıştır. Her örnek için şayet birden fazla kural elde edildiye bunlardan en fazla örneği sınıflandıranı seçilmiştir. Adım 3-8; Birinci örnekten rastgele başlanır. İlk olarak birinci örneğin birinci kalınlığı yani Çizgi1 ele alınır. Bu karakterin değeri Çizgi1=4 tür ve diğer örneklerin birinci kalınlığı ile karşılaştırılır. Dokuzuncu, onaltıncı, yirmibeşinci, yirmialtıncı, yirmisekizinci ve otuzuncu örneklerin birinci temel karakteristik değerleri de 4 yani Çizgi1=4

Tablo 4.1 Çizgikodu Tanıma Tekniği Uygulaması için Örnekler Seti.

Ör. No	Ç1	Ç2	Ç3	Ç4	Ç5	Ç6	Ç7	Ç8	Ç9	Ç10	Ç11	Ç12	Ç13	Ç14	Ç15	Ç16	Ç17	Ç18	Ç19	Ç20	Kodu
1	4	3	3	2	2	4	1	4	5	4	1	3	5	4	2	5	5	1	5	2	Kod1
2	3	4	1	3	3	2	4	4	2	2	5	5	3	5	5	2	4	5	2	3	Kod2
3	1	5	4	1	3	1	1	4	2	1	2	2	2	5	5	3	2	1	1	4	Kod3
4	3	3	4	2	4	2	1	3	1	3	5	2	4	2	2	5	4	4	3	1	Kod4
5	3	4	5	5	1	3	5	3	4	3	3	3	2	3	2	1	2	5	1	2	Kod5
6	2	3	1	3	2	4	3	1	5	4	3	2	1	4	3	4	3	5	1	2	Kod6
7	1	1	2	1	1	3	4	3	5	1	1	4	3	2	1	4	5	3	1	4	Kod7
8	3	3	3	2	2	1	3	1	5	1	3	2	5	4	2	4	2	1	1	2	Kod8
9	4	2	2	3	2	2	1	3	2	5	3	4	5	2	3	1	4	3	5	1	Kod9
10	5	4	2	1	3	2	5	1	1	2	3	5	3	3	5	5	4	4	5	2	Kod10
11	3	3	4	1	3	3	5	3	3	3	2	4	3	4	5	2	2	2	1	3	Kod11
12	2	3	2	5	3	1	4	4	4	4	1	5	2	5	1	1	4	2	3	1	Kod12
13	1	1	1	4	3	3	2	3	4	4	2	5	4	1	4	4	1	3	3	2	Kod13
14	5	5	4	3	4	2	2	3	5	1	4	2	1	1	1	3	5	3	3	5	Kod14
15	2	2	2	2	3	1	3	5	3	5	4	3	4	1	4	4	3	1	2	2	Kod15
16	4	1	3	4	5	2	5	5	4	5	5	3	1	5	4	4	3	1	1	1	Kod16
17	3	3	1	2	4	5	3	1	5	2	2	4	2	3	2	5	4	2	2	5	Kod17
18	3	5	3	2	5	2	2	1	2	1	4	5	2	4	3	5	4	3	1	3	Kod18
19	2	4	2	4	1	3	5	2	5	1	5	4	3	5	3	2	5	1	3	4	Kod19
20	2	2	2	3	2	3	2	1	5	1	4	5	2	5	5	3	3	4	5	2	Kod20
21	1	2	3	1	4	1	2	4	3	1	5	3	1	3	2	4	5	5	1	1	Kod21
22	1	1	2	1	5	5	3	4	1	2	1	5	4	4	2	3	3	5	3	2	Kod22
23	2	2	3	2	3	3	3	1	3	2	5	5	5	5	3	5	5	2	1	4	Kod23
24	1	2	4	1	1	2	2	3	2	2	5	2	2	3	5	3	4	4	3	5	Kod24
25	4	3	2	3	4	3	5	2	4	5	2	2	1	2	5	4	3	4	2	3	Kod25
26	4	5	1	3	3	1	3	4	3	3	5	5	3	3	4	1	1	2	1	1	Kod26
27	3	3	3	4	1	3	1	2	4	3	3	2	5	4	3	3	4	1	1	2	Kod27
28	4	5	2	4	4	1	4	3	2	4	5	5	4	1	4	5	1	5	1	2	Kod28
29	3	5	1	1	3	2	3	3	4	3	2	1	3	3	4	3	4	5	3	4	Kod29
30	4	4	4	1	4	1	2	5	5	2	1	2	5	2	2	4	4	4	4	2	Kod30

olduğundan birinci örneğin birinci temel karakteristik değeri birinci örneği sınıflandıramaz. Aynı şekilde Birinci örneğin ikinciden başlayarak yirminci temel karakteristik değerlerine kadar teker teker alınarak herhangi bir sınıflama yapılamaz. Şart sayısı bir artırılarak 2 yapılır. Birinci örneğin kalınlık değerleri Çizgi1=4 ile Çizgi2=3 yani (4,3) kombinezonu ele alınarak diğer örneklerdeki aynı temel karakteristik değerleri ile karşılaştırıldığında görülür ki yirmibeşinci örneğin ilgili değerleri Çizgi1=4 ile Çizgi2=3 yani (4,3) olduğundan birinci örneğin bir sonraki ikili kombinezonu Çizgi1=4 ile Çizgi3=3 yani (4,3) ele alınır. Bu kombinezon yine diğer örneklerdeki aynı temel karakteristik değerleri ile karşılaştırıldığında onaltıncı örneğin ilgili değerleri Çizgi1=4 ile Çizgi3=3 yani (4,3) olduğu görülür ve birinci örneğin bir sonraki ikili kombinezonu Çizgi1=4 ile Çizgi4=2 yani (4,2) ele alınır ve görülür ki geri kalan 29 örneğin hiçbirinin ilgili temel karakteristik değerleri Çizgi1=4 ve Çizgi4=2 değildir. Böylece 1 nolu örnek sınıflanmış olur. Buradan, Kural-1, "EĞER Çizgi1=4 VE Çizgi4=2 İSE KitapKodu=KOD1" dir. Bu örnek setten çıkarılarak aynı işlemler geriye kalan örnekler içinde uygulanır. 30 iterasyon sonucunda tüm örnekler için herbiri için bir adet olmak üzere toplam 30 adet kural çıkarılmış olur ve sınıflanmamış örnek kalmamış olur. Bu prosesin nasıl işlediği Tablo 4.2'de gösterilmiştir. Buna göre elde edilen kurallar seti Tablo 4.3'te gösterilmektedir. Bu, aynı zamanda bilgi tabanıdır.

Dikkat edilirse her KOD için 20 çizgi yani yirmi karakteristik değeri kullanıldığı halde, elde edilen kurallara göre karakteristiklerden en az 18 tanesinin fazla bilgi olduğu ortaya çıkarılmıştır. Böylece gerekli olan bilginin her KOD için en fazla 2 karakteristik değer olduğu ortaya çıkarılmıştır. Bu ise gerek bilgiyi saklamak, gerekse kullanmak (burada KOD'ları tanımak) için hem hafıza hem de zamandan tasarruf etmek demektir.

Elde edilen kurallar ile tüm örnekler seti %100 tanınmıştır. Bu da öğrenmenin %100 doğru olduğunu göstermektedir.

Bu kuralların her biri bir KOD için kullanılmaktadır. Buna göre örneğin KOD1'i hafızada çizgi olarak saklamak yerine elde edilmiş 1 nolu kuralı kullanmak yeterlidir. Görüldüğü gibi KOD'u saklamak için 20 değere, elde edilmiş kural için sadece 2 değere ihtiyaç vardır.

Bu kurallar bir KOD'u saklamak için kullanıldığı gibi tanımak için de kullanılabilir. Tanıma işlemi ise sadece en fazla iki karakteristik değer test edilmesi ile gerçekleşebilmektedir. Dolayısıyla işlem hızlıdır.

Tablo 4.2. Alternatif Çizgikodu Tanıma Tekniği Uygulanması için Endüktif Kural Çıkarma İşleminin Özeti.

İter.	Kullanılan Örnekler	Çıkarılan Kural Sayısı	Tanıma Tekniği Sınıflandırılan Örnekler	Sınıflandırılmayan Örnekler	
				Çıkarılan Kural Sayısı	Örnekler
1	1	1	1	2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30	
2	2	1	2	3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30	
3	3	1	3	4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30	
4	4	1	4	5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30	
5	5	1	5	6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30	
6	6	1	6	7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30	
7	7	1	7	8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30	
8	8	1	8	9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30	
9	9	1	9	10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30	
10	10	1	10	11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30	
11	11	1	11	12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30	
12	12	1	12	13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30	
13	13	1	13	14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30	
14	14	1	14	15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30	
15	15	1	15	16,17,18,19,20,21,22,23,24,25,26,27,28,29,30	
16	16	1	16	17,18,19,20,21,22,23,24,25,26,27,28,29,30	
17	17	1	17	18,19,20,21,22,23,24,25,26,27,28,29,30	
18	18	1	18	19,20,21,22,23,24,25,26,27,28,29,30	
19	19	1	19	20,21,22,23,24,25,26,27,28,29,30	
20	20	1	20	21,22,23,24,25,26,27,28,29,30	
21	21	1	21	22,23,24,25,26,27,28,29,30	
22	22	1	22	23,24,25,26,27,28,29,30	
23	23	1	23	24,25,26,27,28,29,30	
24	24	1	24	25,26,27,28,29,30	
25	25	1	25	26,27,28,29,30	
26	26	1	26	27,28,29,30	
27	27	1	27	28,29,30	
28	28	1	28	29,30	
29	29	1	29	30	
30	30	1	30	-	

Tablo 4.3 Alternatif Çizgikodu Tanıma Tekniğinin Bilgi Tabanı.

-
1. EĞER Çizgi(1) = 4 VE Çizgi(4) = 2 İSE Çizgikodu = KOD1 dir.
 2. EĞER Çizgi(1) = 3 VE Çizgi(4) = 3 İSE Çizgikodu = KOD2 dir.
 3. EĞER Çizgi(1) = 1 VE Çizgi(2) = 5 İSE Çizgikodu = KOD3 dir.
 4. EĞER Çizgi(1) = 3 VE Çizgi(9) = 1 İSE Çizgikodu = KOD4 dir.
 5. EĞER Çizgi(3) = 5 İSE Çizgikodu = KOD5 dir.
 6. EĞER Çizgi(1) = 2 VE Çizgi(3) = 1 İSE Çizgikodu = KOD6 dir
 7. EĞER Çizgi(1) = 1 VE Çizgi(7) = 4 İSE Çizgikodu = KOD7 dir
 8. EĞER Çizgi(1) = 3 VE Çizgi(5) = 2 İSE Çizgikodu = KOD8 dir.
 9. EĞER Çizgi(1) = 4 VE Çizgi(2) = 2 İSE Çizgikodu = KOD9 dir.
 10. EĞER Çizgi(1) = 5 VE Çizgi(2) = 4 İSE Çizgikodu = KOD10 dir.
 11. EĞER Çizgi(1) = 3 VE Çizgi(9) = 3 İSE Çizgikodu = KOD11 dir.
 12. EĞER Çizgi(1) = 2 VE Çizgi(4) = 5 İSE Çizgikodu = KOD12 dir.
 13. EĞER Çizgi(1) = 1 VE Çizgi(3) = 1 İSE Çizgikodu = KOD13 dir.
 14. EĞER Çizgi(1) = 5 VE Çizgi(2) = 5 İSE Çizgikodu = KOD14 dir.
 15. EĞER Çizgi(1) = 2 VE Çizgi(8) = 5 İSE Çizgikodu = KOD15 dir.
 16. EĞER Çizgi(1) = 4 VE Çizgi(2) = 1 İSE Çizgikodu = KOD16 dir.
 17. EĞER Çizgi(1) = 3 VE Çizgi(6) = 5 İSE Çizgikodu = KOD17 dir.
 18. EĞER Çizgi(1) = 3 VE Çizgi(5) = 5 İSE Çizgikodu = KOD18 dir.
 19. EĞER Çizgi(1) = 2 VE Çizgi(2) = 4 İSE Çizgikodu = KOD19 dir.
 20. EĞER Çizgi(1) = 2 VE Çizgi(7) = 2 İSE Çizgikodu = KOD20 dir.
 21. EĞER Çizgi(1) = 1 VE Çizgi(3) = 3 İSE Çizgikodu = KOD21 dir.
 22. EĞER Çizgi(1) = 1 VE Çizgi(5) = 5 İSE Çizgikodu = KOD22 dir.
 23. EĞER Çizgi(1) = 2 VE Çizgi(3) = 3 İSE Çizgikodu = KOD23 dir.
 24. EĞER Çizgi(1) = 1 VE Çizgi(6) = 2 İSE Çizgikodu = KOD24 dir.
 25. EĞER Çizgi(1) = 4 VE Çizgi(6) = 3 İSE Çizgikodu = KOD25 dir.
 26. EĞER Çizgi(1) = 4 VE Çizgi(3) = 1 İSE Çizgikodu = KOD26 dir.
 27. EĞER Çizgi(1) = 3 VE Çizgi(4) = 4 İSE Çizgikodu = KOD27 dir.
 28. EĞER Çizgi(1) = 4 VE Çizgi(7) = 4 İSE Çizgikodu = KOD28 dir.
 29. EĞER Çizgi(12) = 1 İSE Çizgikodu = KOD29 dir.
 30. EĞER Çizgi(19) = 4 İSE Çizgikodu = KOD30 dur.
-

4.3.3 Programda kullanılan Menüler

Program sırasıyla *Barkod*, *Veri/Bilgi* ve *Renkler* olmak üzere üç ana menü altında toplanmıştır. Bu üç ana menü altında ise alt menüler bulunmaktadır. *Barkod* ana menüsü altında olan alt menüler sırasıyla *BarkodÜret*, *BarkodTanı*, *DosShell*, *Yazdır* ve *Çıkış*; *Veri/Bilgi* ana menüsü altında sırasıyla *EkrandanTanıma*, *KodNosuna GöreTanıma*, *Bilgi tabanıGöster* ve *VeriTabanınıListeleme*; *Renkler* ana menüsünün altında ise dört çeşit ekran arka ve ön zemin rengi alt menüler şeklinde donatılmıştır.

Bu menülerin işlevleri aşağıda anlatılmaktadır.

Barkod Ana Menüsü;

1. ***BarkodÜret:*** Burada program çizgikalınlığı rastgele olarak bir ila beş arasında, çizgi kalınlıkları arasındaki boşluklar ise rastgele olarak bir ila yedi arasında değişen 30 adet barkod benzetim tekniği ile üretilmekte ve aynı zamanda bu üretilen çizgikodlarının sadece kalınlık değerleri bir kayıt dosyası içinde saklanması sağlanmaktadır.
2. ***BarkodTanı:*** *BarkodÜret* alt menüsünde oluşturulan çizgikodu kalınlık değerlerini dosyadan okuyarak bilgi tabanındaki kuralları tarayarak ilgili çizgikod tanınır.
3. ***Yazdır :*** Veritabanının çıktısı alınır.
4. ***DosShell:*** Programdan geçici süreli olarak DOS'a geçilir.
5. ***Çıkış:*** Menüden çıkılarak programın Yazılımına geçilir.

Veri/Bilgi Ana Menüsü;

1. ***EkrandanTanıma:*** Ekrandan girilen kalınlık bilgilerine bakarak bu kalınlık bilgilerinin hangi kitap koduna ait olduğunu ve ilgili kitap kodu verilerini verir.
2. ***KodaGöreTanıma:*** Girilen Kitap Kodu numarasına göre Veri Tabanındaki bilgileri okuyarak sadece o kod numarasına ait verileri verir.
3. ***BilgitabanıGöster:*** RULES-3 tarafından çıkarılmış olan kural setini gösterir.
4. ***VeriTabanınıGöster:*** Mevcut tüm kitap bilgilerini KitapKodu, Yazarı, Kitapadı, BasımYılı, BasımYeri, Fiyatı, Basıldığı Şehir ve Ülke verilerini gösterir.

Renkler;

Daha öncede bahsedildiği üzere dört çeşit renk sözkonusudur.

1. *SiyahVeBeyaz*: Arka zemin rengi siyah ve ön zemin rengi ise beyaz renktedir.
2. *MaviVeAçıkYeşil*: Arka zemin rengi mavi ve ön zemin rengi ise AçıkYeşil renktedir.
3. *AçıkYeşilVeMavi*: Arka zemin rengi açıkyeşil ve ön zemin rengi ise mavi renktedir.
4. *GriVeKırmızı*: Arka zemin rengi gri ve ön zemin rengi ise kırmızı renktedir.

Sistem baştan sona şu adımlardan aşarak tanıma işlemini gerçekleştirir.

- Benzetim tekniği kullanılarak herbirinin kalınlıkları rastgele olarak değişen yirmi adet çizgiden oluşan 30 adet çizgikodu üretilir. Mevcut sistemlerdeki gibi okuyucunun şifreyi çözebilmesi için çizgikodu etiketinin anma boyutlarının ve çizgi kalınlıklarının belli standartlar dahilinde olması zorunluluğu yoktur. Okuma esnasında boşluk değerleri bir şey ifade etmediğinden çizgikodları sadece çizgilerden oluşur. Çizgi sayısı bu çalışmada yirmi olarak alınmıştır. Bu sayı ihtiyaca binaen arttırılabilir. Çizgikodunun anma boyutları, uygulama alanına bağlı olarak nasıl uygun ise öyle alınabilir. Bu çalışmada çizgilerin kalınlık değerleri ise 1 ila 5 piksel arasında rastgele olarak değişmektedir. Bu değerler de uygulama alanına bağlı olarak ihtiyaca binaen değiştirilebilir. Çizgikodu sayısı veri tabanı ile sınırlı olmak zorundadır. EK-1'de de verildiği üzere 30 adet kitabın herbirinin kitap kodu, yazarı, kitap adı, sayfa adedi, basım yılı, basım yeri, hangi ülkede basıldığı verileri veriler.kod ismindeki bir veri kayıt dosyasına önceden kaydedilir.

- Benzetilmiş sistemde, birinci üretilen çizgikodunun kalınlık bilgileri Kod1.dat, ikincisi ise Kod2.dat ve otuzuncusu ise Kod30.dat dosyasında olmak üzere benzetim tekniği ile üretilmiş her bir çizgikodunun kalınlık bilgileri bir kayıt dosyasına kaydedilir. Daha sonra bu kayıtlardaki kalınlık bilgileri okutularak üretilen çizgikodları ekrana getirilir. Böylece üretilen çizgikodları bir okuyucu tarafından okunmaya hazır hale getirilmiş olur. Pratikte ise benzetim tekniği kullanılarak üretilen çizgikodlarının çıktıları laser yazıcıdan alınır. Daha sonra bu çıktılar veriler.kod dosyasına kaydedilen her bir kitabın üzerine yerleştirilir.

- Çizgikodu etiketlerinin okuyucudan taranması benzetilmiş sistemde ekrana getirilen çizgikodları dik kesilecek şekilde renk yoğunluklarından istifade edilerek her hangi bir h yüksekliğinde taranır. Tarama esnasında çizgikodunu oluşturan her bir çizginin kalınlık değerleri bulunur.

- Bilgi tabanının oluşturulması ; 4.3.2'de detaylıca anlatılmıştır.

- Kalınlık bilgileri dosyadan okutulur ve rastgele olarak ekrana getirilen çizgikodları taranarak çizgikodunu oluşturan her bir çizginin kalınlık değeri bulunur. Bu kalınlık değeri bilgi tabanını oluşturan kurallar ile karşılaştırılır. Çizgi kalınlık değeri bilgi tabanındaki bir kurala uyuyorsa, bu kurala uyan çizgikodu tanınır ve o koda ait veriler kod dosyasındaki kitap kodu, yazarı, kitap adı, sayfa sayısı, basım yılı, basım yeri, hangi ülkede basıldığı verileri ekrandan listelenir.

Diğer bir seçenek ise, alternatif tekniğin alfabesini karakterize eden piksel olarak Çizgikodu kalınlık değeri klavye ile girmek suretiyle ilgili kitap kodu ve verilerini elde edilebilmesidir.

4.4 Sonuç

Alternatif teknik uluslararası standart bir sistem olmaktan ziyade spesifik kullanımlara uygundur.

Pratikte alternatif tekniğin donanım kısmının bir PC bilgisayar veya yazarkasa, bir okuyucu ve bir çizgikodu etiketinden ibaret olacak şekilde tasarlanabileceği düşünülmektedir. Bu tasarıma tercihe bağlı olarak klavye ilave edilebilecektir. Bu donanım spesifik kullanımlar için özel donanımlı cihazlara gerek kalmadığı için çok uygun ve ucuz olabilecektir.

Tekniğin çizgikodu etiketlerinin standart boyut ve belli toleranslar dahilinde olması şartı yoktur. Okuyucuların mevcut çizgikodu etiketlerini okuyabilmesi için bazı şartların sağlanması mecburiyeti yoktur. Çizgikodları özel basım teknikleri kullanılmadan hazırlanması hem maliyet hem zaman hem de basitlik açısından avantaj olacaktır.

Tekniğin alfabesi çizgilerin sadece kalınlık değerlerinden oluşturulan vektörlerdir. Okuyucuların mevcut çizgikodu etiketlerini okuyabilmesi için bazı şartların sağlanması gerekmektedir. Çizgikodu etiketlerinin anma boyutlarının standart ölçülerde, belli toleranslar ve belli bir alfabe ile hazırlanması zorunda olmasının yanında bir de okuyucuların şifresi çözülecek alfabe veya alfabelere programlı olmaları gereklidir. Her okuyucu her alfabe okuyamamakta olduğu, sadece programlandığı alfabe veya alfabeleri okuyabilmekte olduğu unutulmamalıdır. Alternatif teknikte ise tüm çizgikodunu baştan sona taranması ve sadece çizgilerin taranması, boşlukların

taranmaması sayesinde hem hafıza, hem de zamandan tasarruf edilmektedir. Alternatif tekniğin alfabesinin basitliği, okuyucunun programlanmasına gerek duyulması zorunluluğunu ortadan kaldırabileceği düşünülmektedir. Bunun için çizgikodu alfabesini oluşturan çizgilerin kalınlık değerlerinin vektörler ile ifade edilip bir bilgi tabanı oluşturulması yeterlidir.

Mevcutlarından farklı olarak yazılım kısmı iki önemli özelliğe sahiptir. Bu YZ'nin EÖ tekniğinin özelliğinden kaynaklanmaktadır. Birincisi ki bu avantaj sağlar örneklerden faydalanılarak bilgi tabanı oluşturulduğu için yazılımın çok daha kolay ve basit olarak ve çok daha kısa süre içerisinde yazılabilme imkanı doğmuştur. İkincisi ise örnek sayısının artırılması veya azaltılması durumunda bilgi tabanı en baştan yeniden oluşturulma zorunluluğu sözkonusudur ki bu da dezavantaj sayılabilir.

BÖLÜM 5. SONUÇ

Alternatif çizgikodu tanıma tekniği mevcut tanıma sistemleri ile karşılaştırıldığı zaman bazı avantajlara sahiptir. Aşağıda bu avantajlarından genişçe bahsedilmektedir.

Tekniğin bilgi tabanı için üretilen kurallar ile simülasyon tekniği ile üretilmiş 30 adet çizgikodunun 30'unu da teknik %100 tanımıştır. Örnekler seti ne kadar çok olursa olsun bilgi tabanında muhakkak ilgili çizgi kodu için bir kural üretilecektir. 486 DX 33 donanımında 33 MgHz'lik bir hıza sahip bir bilgisayarda 30 örnekten oluşan örnek seti için bilgi tabanı 33 saniyede, 486 DX 2x66 donanımında 2x66 MgHz'lik bir hıza sahip bir diğer bilgisayarda ise 9 saniyede oluşmuştur.

Mevcut sistemlerde çizgikodun okunabilmesi, yani bu şifrenin çözülebilmesi için sadece çizgikodu etiketi, okuyucu ve bilgisayar haricinde ışıklı kalem, sabit ışık üstü cam, okunan çizgi kodu gösterecek ekran ve bilgisayarlarla bağlantılı özel donanımlı cihazlar gerekli olabilmektedir. Alternatif teknik tüm çizgikodunu baştan sona taramaz, sadece çizgileri tarar boşlukları taramaz. Bu sebeple gerek bilgi tabanında bilgi saklamak, gerekse bilgiyi kullanmak için hem hafıza hem de zamandan tasarruf edilir. Tanıma işlemi esnasında tüm çizgiler taranmadığı için hızlıdır.

Her KOD için 20 çizgi yani 20 karakteristik değeri kullanıldığı halde, elde edilen kurallara göre karakteristiklerden en az 18 tanesinin fazla bilgi olduğu ortaya çıkarılmıştır. Böylece gerekli olan bilginin her KOD için en fazla 2 karakteristik değer olduğu ortaya çıkarılmıştır. Bu ise gerek bilgiyi saklamak, gerekse kullanmak (burada KOD'ları tanımak) için hem hafıza hem de zamandan tasarruf etmek demektir.

Bu kuralların her biri bir KOD için kullanılmaktadır. Buna göre örneğin KOD1'i hafızada çizgi olarak saklamak yerine elde edilmiş 1 nolu kuralı kullanmak yeterlidir. Görüldüğü gibi KOD'u saklamak için 20 değere, elde edilmiş kural için sadece 2 değere ihtiyaç vardır.

Bu kurallar bir KOD'u saklamak için kullanıldığı gibi tanımak için de kullanılabilir. Tanıma işlemi ise sadece en fazla iki karakteristik değer test edilmesi ile gerçekleşebilmektedir. Dolayısıyla işlem hızlıdır.

İRTİSMA VE ÖNEMLER

Çizgi kod işaretlerinden perakende satış dışında toptan satışlarda, ambar kontrolünde, muhasebe kayıtlarında ve daha birçok alanda yararlanabilmek için genel amaçlı bilgisayarlar ve onların programlarına ek olarak özel donanıma ve özel programlara gerek duyulmaktadır. Ancak son zamanlarda bir PC veya yazarkasa bir okuyucu ve standart bir çizgikodu alfabesi ile ifade edilmiş bir çizgi kodu etiketinden ibaret sistemler yaygın olarak kullanılmaya başlanmıştır. Alternatif tekniğin donanım kısmına arzu edilirse klavye eklemek mümkündür. Bu sayede kalınlık bilgileri klavyeden girilerek bilgiye ulaşma darboğazının aşılmasında destek sağlanabileceği umulmuştur. Alternatif teknik daha basit, daha ucuz olmasının yanında klavyeden yararlanabilme imkanı sağlayabilmektedir.

Daha öncede bahsedildiği gibi, mevcut sistemlerdeki gibi okuyucunun şifreyi çözebilmesi için çizgikodu etiketinin anma boyutlarının ve çizgi kalınlıklarının belli standartlar dahilinde olma zorunluluğu yoktur. Çizgikodunu oluşturan çizgi sayısı ve anma boyutları, uygulama alanına bağlı olarak değişebilecektir. Bir anlamda çizgi kodunu oluşturan her bir çizginin piksel olarak kalınlık değerlerinden meydana gelen vektör alternatif tekniğin alfabesini teşkil ettiği farzedilebilir. Tekniğin yazılım ile elde edilebilen çizgi kodu etiketleri lazer yazıcıdan çıktı alınarak esas film hazırlanmasına gerek kalmaksızın baskıya hazır hale getirilebilmesi büyük avantaj sağlayacağı düşünülebilir. Bu sebeplerden dolayı, dolayı maliyet daha ucuz ve etiketlerin üretilmesi daha kolay ve çabuktur.

Mevcut çizgi kodları tanıma işlemi bir okuyucu tarafından tanıma alanı içerisinde taranarak gerçekleştirilir. Şayet θ açısı arasında kalan bölgede çizgikodu çizgilerinde bir deformasyon varsa çizgikodu tanınamayacaktır. Alternatif teknikte okuyucunun belli bir standart alfabe ile programlanmasına gerek kalmaksızın tanıma alanı içinde çizgikodu kalınlıklarının piksel değerleri okunmak suretiyle çizgikodu tanınır. Çizgikodu tanıma işlemi esnasında çizgi kodunu oluşturan tüm çizgiler taranmasına ihtiyaç duyulmaz. Tarama işlemi bilgi tabanında bir kural bulununca sona erecektir. Deformasyon, kuralı oluşturan çizgi veya çizgilerde ise çizgikodu tanınamayacaktır. Bu, hem zaman, hem hafıza, hem de güvenilirlik açısından avantaj sayılır.

Örnekler setinin tamamı RULES-3 EÖ tekniği kullanılarak elde edilmiş kurallar ile %100 tanınmıştır. Bu, öğrenmenin %100 ve tekniğin güvenilirliğinin ve okuma ve tanıma hızlarının yüksek olduğunu göstermektedir.

BÖLÜM 6. TARTIŞMA VE ÖNERİLER

Alternatif teknik daha ziyade uluslararası bir standart olmaktan ziyade bir işletme bünyesinde gerçekleştirilebilen üretim ve stok takibi, Personel Devam Takip Sistemi vb. spesifik alanlarda kullanılması basitliği, ucuzluğu, bilgisayarın hafızasında az yer kaplaması ve daha verimli olması sebeplerinden dolayı daha fazla avantaj sağlayacaktır. İleride bu konu üzerinde çalışmak isteyenler için tekniğin uluslararası standart bir alfabe olarak tasarlanma imkanı araştırılabilir.

Alternatif teknikte üretilen 30 adet çizgikodlarının her biri kalınlıkları random olarak değişen 1 ila 5 piksel olarak değişen 20 adet çizgiden meydana gelmektedir. Random olarak üretilen çizgikodlarının iki tanesinin aynı olma ihtimali;

$$1/5^{20} \text{ dir.}$$

Yani böyle bir durumun ortaya çıkması çok çok zayıf bir ihtimaldir. Yine de ortaya çıkarsa en baştan tekrar random olarak çizgi kodları üretilerek bu problem giderilebilecektir.

Örneklerden faydalanılarak bilgi tabanı oluşturulduğu için yazılım, çok daha kolay ve basit olarak ve çok daha kısa süre içerisinde gerçekleştirilebilmektedir. Örnek sayısının arttırılması veya azaltılması durumunda bilgi tabanının en baştan yeniden oluşturulma zorunluluğu sözkonusudur ki bu dezavantaj sayılabilir. Çünkü bu çalışma için ne kadar örnek varsa hepsi için bir kural üretilme zorunluluğu vardır.

Çizgikodu ile birlikte aynı zamanda klavyenin kullanılabilmesi, bilgiye ulaşılmada ve bilgi darboğazının aşılmasında destek rolü oynayacağı düşünülmüştür.

KAYNAKLAR

- [1] Aksoy M.S. "New algorithms for machine learning", PhD Thesis, University of Wales College of Cardiff, Cardiff, U.K., 1993.
- [2] Cordier M.O. and Reynaud C. "Knowledge acquisition techniques and second generation expert systems", Applied Artificial Intelligence, 5, pp. 209-226, 1991.
- [3] Feigenbaum E.A. "Expert systems in 1980s" in Infotech State of the Art Report on Machine Intelligence, Edited by A.Bond, Maidenhead, Pergamon, Infotech, 1981.
- [4] Forsyth R. "Machine learning principles and techniques", Ed: R. Forsyth, Chapman and Hall, London, 1989.
- [5] Hancox P.J., Mills W.J. and Reid B.J. "Artificial intelligence/expert systems", Ergosyst Associates, Lawrence, Kansas, 1990.
- [6] Pham D.T. and Pham P.T.N. "Expert systems in mechanical and manufacturing engineering", The international Journal of Advanced Manufacturing Technology, 3(3), pp.3-21, 1988.
- [7] Quinlan J.R. "Induction, knowledge and expert systems", in Artificial Intelligence Developments and Applications, Eds: J.S. Gero and R. Stanton, Amsterdam, North-Holland, pp.239-266, 1988.
- [8] Waterman D.A. "A guide to expert systems", Addison-wesley, California, 1986.
- [9] Wu X. "Inductive learning: Algorithms and frontiers", Artificial Intelligence Review, 7, pp. 93-108, 1993.
- [10] Yasdi R. "Learning classification rules from database in the context of knowledge acquisition and representation", IEEE Trans. on Knowledge and Data Engineering, 3(3), pp.177-183, 1991.

[11] Hunt,E.B., Marin,J., ve Stone,P.J., "Experiments in Induction", Academic Press, New York, London, 1966.

[12] Michalski, R.S. et al., "The Multi-purpose Incremental Learning System AQ15 and its testing Application to three Medical Domains", Proc. of The National Conference on AI, Philadelphia, PA, 1986.

[13] Liu W.Z. and White A.P., "A Review of Inductive Learning", in I.M. Graham and R.W. Milne, Research and Development in Expert Systems VIII, Proc. of Expert Systems 91, pp. 112-126. Cambridge Univ. Press, London, 1991.

[14] TÜRKEKEL S. "Uluslararası Mal Numaralama Birliği (EAN) nedir, TOBB'nin EAN'a Üyeliği, Milli Mal Numaralama Merkezi nedir, Faaliyetleri nelerdir", TOBB EAN İhtisas Toplantısı, Ankara, 27/09/1988

[15] DUMAN H. " ISBN Uluslararası Standart Kitap Numaralama Sistemi ve Türkiye'de Uygulaması", TOBB EAN İhtisas Toplantısı, Ankara, 27/09/1988

[16] İŞMEN M. " Çizgikodun Teknik Özellikleri, Master Film Hazırlanması ve Baskıya Uygulanması", TOBB EAN İhtisas Toplantısı, Ankara, 27/09/1988

[17] TS 10146/Nisan "Barkod (Çubuk Kodlama) Sembolleri - Terimleri", UDK 681.327.12:003.2 (083.73), Birinci Baskı,1992

[18] TS 10147/Nisan, "Barkod (Çubuk Kodlama) Sembolleri - EAN-13 ve EAN-8- Genel Kurallar", UDK 681.327.12:003.2 (083.73), Birinci Baskı,1992

[19] TS 10149/Nisan, "Barkod (Çubuk Kodlama) Sembolleri - EAN-13 ve EAN-8- Genel Kurallar", UDK 681.327.12:003.2 (083.73), Birinci Baskı,1992

[20] TS 10151/Nisan , "Barkod (Çubuk Kodlama) Sembolleri - ITF-14- Genel Kurallar", UDK 681.327.12:003.2 (083.73), Birinci Baskı,1992

Ek-1 Alternatif Çizgikodu Tanıma Tekniğinin Veri Tabanı.

Kod	Yazar	Bas.Y	Kitap Adı	S.Ad.	Yayınevi	Y.Ülke
Kod1	H. ERKUT	1992	Yönetimde Simülasyon Yaklaşımı	406	Bayrak Y.	İst. /TÜRKİYE
Kod2	A. UĞUR	1991	İşletme Ekonomisi	233	İTÜ SMF M.	Sak. /TÜRKİYE
Kod3	İ. M. DOĞRUER	1989	Mühendislik Ekonomisine Giriş	175	İTÜ SMF M.	Sak. /TÜRKİYE
Kod4	M. AKKURT	1991	Talaş Kaldırma Yöntemleri Ve Takım Tezgahları	224	Birsen Y.	İst. /TÜRKİYE
Kod5	M. AKKURT	1991	Bilgisayar Kontrollü Takım Tezgahları (CNC)	347	Birsen Y.	İst. /TÜRKİYE
Kod6	A. BOZACI	1989	Makina Elemanlarının Tasarımı	369	İTÜ SMF M.	Sak. /TÜRKİYE
Kod7	L. ZEREN	1985	Çözümlü Yüksek Matematik Problemleri	188	Birsen Y.	İst. /TÜRKİYE
Kod8	N. KÜRKÇÜOĞLU	1980	Isı Ve Termodinamik Problemleri	371	Matbaa Tek.	İst. /TÜRKİYE
Kod9	S. ERDOĞAN	1993	QuickBASIC 4.5	528	Beta B. Y. D.	İst. /TÜRKİYE
Kod10	S. GÜÇLÜ	1990	İş Hukuku	86	İTÜ SMF M.	Sak. /TÜRKİYE
Kod11	G. ÖZEL	1988	Basic	292	Beta B. Y. D.	İst. /TÜRKİYE
Kod12	E. E. İNAN	1978	Cisimlerin Mukavemeti Çözümlü Problemler	362	Arpaz M.	İst. /TÜRKİYE
Kod13	H.BODUROĞLU & F.DELALE	1990	Çözümlü Mukavemet Problemleri 1	368	Birsen Y.	İst. /TÜRKİYE
Kod14	O. HALAÇ	1984	Kantitatif Karar Verme Teknikleri	239	Oğul M.	İst. /TÜRKİYE
Kod15	Ö. A. SAÇLI &	1982	Fen Bilimleri İçin Matematik	416	M. E. B.	İst. /TÜRKİYE
Kod16	M. BALCI					
Kod17	T. AKINTÜRK	1989	Temel Hukuk 1. Fasikül	197	AÜ Y.	Esk. /TÜRKİYE
Kod18	İ. A. ŞENGİL	1992	Korozyon	494	İTÜ SMF M.	Sak. /TÜRKİYE
Kod19	L. H. V. Vlack	1990	Malzeme Bilimine Giriş	272	Birsen Y.	İst. /TÜRKİYE
Kod20	A. ARAN	1989	Metal Döküm Teknolojisi	165	İTÜ Mak.F.A.	İst. /TÜRKİYE
Kod21	F. Y. BOR	1988	Ekonomik Metalurji	250	İTÜ MFA.	İst. /TÜRKİYE
Kod22	E. N. ÇAVUŞOĞLU	1992	Döküm Teknolojisi	424	İTÜ MFA	İst. /TÜRKİYE
Kod23	V. AYTEKİN	1980	Metalurji Termodinamiği	313	İTÜ Met.F.A.	İst. /TÜRKİYE
Kod24	K. ONARAN	1985	Malzeme Bilimi	319	Çağlayan B.	İst. /TÜRKİYE
Kod25	V. SEVİNÇ & A.O. AYDIN	1987	Temel Kimya	177	İTÜ SMF M.	Sak. /TÜRKİYE
Kod26	E. S. TÜRKER	1990	Diferansiyel Denklemler	430	İTÜ SMF M.	Sak. /TÜRKİYE
Kod27	C. BİNBALIOĞLU	1988	Eğitime Giriş	190	Binbaşıoğlu Y.	Ank. /TÜRKİY
Kod28	İ. E. Başaran	1994	Eğitim Psikolojisi	368	Gül Y.	Ank. /TÜRKİY
Kod29	S. Şener	1994	Sosyoloji	127	Dcha Y.	Sak. /TÜRKİYE
Kod30	M. Hesapçıoğlu	1994	Öğretim İlke Ve Yöntemleri	390	Beta B. Y. D.	İst. /TÜRKİYE

EK-2 Program Listesi.

```

DECLARE SUB Yazdir ()
DECLARE SUB KayitAI ()
DECLARE SUB Listele ()
DECLARE SUB SonucuBul ()
DECLARE SUB EkrandanTani ()
DECLARE SUB DosyaKodYaz ()
DECLARE SUB VeriKodYaz ()
DECLARE SUB Veritabani ()
DECLARE SUB KuralSeti ()
DECLARE SUB DosyaAc ()
DECLARE SUB Yazim ()
DECLARE SUB DosyaYeni ()
DECLARE SUB PrintHelpLine (help$)
DECLARE FUNCTION Menu% (CurrChoiceX%, MaxChoice%, choice$, itemRow%(),
itemcol%(), help$, BarMode%)
DECLARE SUB LoadState ()
DECLARE SUB initialize ()
DECLARE SUB fancyCls (Dots%, Background%)
DECLARE SUB Center (row%, text$)
DECLARE SUB box (Row1%, Col1%, Row2%, Col2%)
DECLARE SUB SaveState ()
DECLARE FUNCTION Trim$ (x$)
DECLARE SUB menusystem ()
DEFINT A-Z
ON ERROR GOTO Hata
CONST TRUE = -1
CONST FALSE = NOT TRUE
TYPE Test
    Harf AS STRING * 1
END TYPE
DIM SHARED Test1 AS Test
DIM SHARED TestEkran AS Test
DIM SHARED S1 AS LONG
VeriN = FREEFILE
OPEN "VERILER.KOD" FOR RANDOM AS VeriN LEN = LEN(Test1)
DO UNTIL EOF(VeriN)
    S1 = S1 + 1

```

```

    GET #VeriN, S1, Test1
LOOP
CLOSE #VeriN
S1 = S1 - 1

```

'User-defined types

```

TYPE AccountType

```

```

    Title    AS STRING * 20

```

```

    AType    AS STRING * 1

```

```

    Desc     AS STRING * 50

```

```

END TYPE

```

```

TYPE veri

```

```

    Kod      AS STRING * 6

```

```

    Yazar    AS STRING * 20

```

```

    Yil      AS STRING * 4

```

```

    Ad       AS STRING * 50

```

```

    Sayfa    AS STRING * 5

```

```

    Ucret    AS STRING * 15

```

```

    Yer      AS STRING * 14

```

```

    BYer     AS STRING * 16

```

```

END TYPE

```

```

DIM SHARED Dosya AS veri

```

```

DIM SHARED DosyaKayit AS LONG

```

```

DosyaKayit = LEN(Dosya) / S1

```

```

TYPE Recordtype

```

```

    Date     AS STRING * 8

```

```

    Ref      AS STRING * 10

```

```

    Desc     AS STRING * 50

```

```

    Fig1     AS DOUBLE

```

```

    Fig2     AS DOUBLE

```

```

END TYPE

```

```

DIM SHARED account(1 TO 19) AS AccountType 'Stores the 19 account titles

```

```

DIM SHARED ColorPref 'Color Preference

```

```

DIM SHARED colors(0 TO 20, 1 TO 4) 'Different Colors

```

```

DIM SHARED ScrollUpAsm(1 TO 7) 'Assembly Language Routines

```

```

DIM SHARED ScrollDownAsm(1 TO 7)
DIM SHARED PrintErr AS INTEGER          'Printer error flag
DIM SHARED DosyaKol%, TanimKol%, RenkKol%, Bulunamadi, Aded, OkunanS
DIM SHARED Uzunluk%, Oteleme%, SutunYaz%, SatirYaz%, Adet, IstencnS, indeks
DIM SHARED itemRow(80), itemcol(80), Dots%, Background%, Cizgi(30), Sonuc
DIM SHARED x1, y1, x2, y2, Kutu(1000), veri, BitirYaz, BirOnceki, SutunVeri%
DIM SHARED YaziciHata
initialize
menusystem
END

```

Hata:

```

SELECT CASE ERR

```

```

    ' If data file not found, create and initialize a new one.

```

```

CASE 53

```

```

    CLOSE

```

```

    ColorPref = 1

```

```

    FOR a = 1 TO 19

```

```

        account(a).Title = ""

```

```

        account(a).AType = ""

```

```

        account(a).Desc = ""

```

```

    NEXT a

```

```

    SaveState

```

```

    RESUME

```

```

CASE 24, 25

```

```

    PrintErr = TRUE

```

```

    box 10, 25, 14, 55

```

```

    Center 11, "Yazıcı hazır değil..."

```

```

    Center 12, "<ENTER> Tekrar dene.."

```

```

    Center 13, "<ESC>ape Vazgeç  "

```

```

    DO

```

```

        HarfS = INKEYS

```

```

    LOOP WHILE HarfS = ""

```

```

    SELECT CASE HarfS

```

```

        CASE CHR$(27)

```

```

            YaziciHata = 1

```

```

        CASE CHR$(13)

```

```

            END SELECT

```

```

        RESUME NEXT

```

```

CASE ELSE

```

END SELECT

RESUME NEXT

'The following data defines the color schemes available via the main menu.

' scrn dots bar back title shdow choice curs cursbk shdow

DATA 0, 7, 15, 7, 0, 7, 0, 15, 0, 0

DATA 1, 9, 12, 3, 0, 1, 15, 0, 7, 0

DATA 3, 15, 13, 1, 14, 3, 15, 0, 7, 0

DATA 7, 12, 15, 4, 14, 0, 15, 15, 1, 0

'The following data is actually a machine language program to

'scroll the screen up or down very fast using a BIOS call.

DATA

&HB8,&H01,&H06,&HB9,&H01,&H04,&HBA,&H4E,&H16,&HB7,&H00,&HCD,&H10,&HCB

DATA

&HB8,&H01,&H07,&HB9,&H01,&H04,&HBA,&H4E,&H16,&HB7,&H00,&HCD,&H10,&HCB

'Box:

' Draw a box on the screen between the given coordinates.

SUB box (Row1, Col1, Row2, Col2) STATIC

BoxWidth = Col2 - Col1 + 1

LOCATE Row1, Col1

PRINT "┌"; STRINGS(BoxWidth - 2, "-"); "┐";

FOR a = Row1 + 1 TO Row2 - 1

LOCATE a, Col1

PRINT "|"; SPACES(BoxWidth - 2); "|";

NEXT a

LOCATE Row2, Col1

PRINT "└"; STRINGS(BoxWidth - 2, "-"); "┘";

END SUB

'Center:

' Center text on the given row.

```

SUB Center (row, text$)
  LOCATE row, 41 - LEN(text$) / 2
  PRINT text$;
END SUB

SUB DosyaAc

'VIEW PRINT (6) TO 23
'FILES "kod*.dat"
'DO WHILE INKEYS = "": LOOP
'VIEW PRINT
SCREEN 12
PALETTE 1, 4144959
DO
  CLS
  Numara = INT((30 - 1 + 1) * RND + 1)
  GOSUB BarCiz
  LOCATE 10, 1: PRINT "KayıtlıKod="; DosyaAdi$
  GOSUB CiziliKoduTara
  SonucuBul

VeriNo1 = FREEFILE
OPEN "veriler.kod" FOR RANDOM AS VeriNo1 LEN = LEN(Dosya)

FOR DosyaKayit = 1 TO 30
  GET #VeriNo1, DosyaKayit, Dosya
  IF Trim(UCASES(Dosya.Kod)) = Trim(UCASES(DosyaAdi$)) THEN
    CikisYok = 0
    SutunVeri% = 1
    DosyaKodYaz
    CLOSE #VeriNo1
    GOTO Son1
  ELSE
    CikisYok = 0
  END IF
  IF CikisYok = 1 THEN
    CLOSE #VeriNo1
    GOTO Son1
  END IF

```

```

NEXT
Son1:

    SonucS = "KOD" + LTRIMS(STRS(Sonuc))
    LOCATE 10, 41
    PRINT "RULES-3 ile bulunan sonuc="; SonucS

VeriNo1 = FREEFILE
OPEN "veriler.kod" FOR RANDOM AS VeriNo1 LEN = LEN(Dosya)

FOR DosyaKayit = 1 TO 30
GET #VeriNo1, DosyaKayit, Dosya
IF Trim(UCASES(Dosya.Kod)) = Trim(UCASES(DosyaAdiS)) THEN
    CikisYok = 0
    SutunVeri% = 41
    DosyaKodYaz
    CLOSE #VeriNo1
    GOTO Son2
ELSE
    CikisYok = 0
END IF
IF CikisYok = 1 THEN
    CLOSE #VeriNo1
    GOTO Son2
END IF
NEXT
Son2:

    aS = INPUT$(1)

    IF aS = CHR$(27) THEN
        SCREEN 0, 0, 0
        menusystem
    END IF
LOOP

BarCiz:
    CLOSE
    DosyaAdiS = "KOD" + LTRIMS(STRS(Numara))

```

```

OPEN "i", #1, DosyaAdi$ + ".DAT"
DO UNTIL EOF(1)
  INPUT #1, x
  LINE (x, 1)-(x, 50), 1, BF
  LINE (x + 320, 1)-(x + 320, 50), 1, BF
LOOP
RETURN

```

CiziliKoduTara:

```

BarNo = 0
FOR a = 1 TO 200
  IF POINT(a, 4) = 1 THEN
    x1 = a
    DO UNTIL POINT(a, 4) = 0
      a = a + 1
    LOOP
    x2 = a
    BarNo = BarNo + 1
    Cizgi(BarNo) = x2 - x1
  END IF
NEXT a
RETURN

```

END SUB

SUB DosyaKodYaz

```

SatirDosya% = 11
LOCATE 1 + SatirDosya%, SutunVeri%: PRINT "Yazarı"; Trim(Dosya.Yazar)
LOCATE 2 + SatirDosya%, SutunVeri%: PRINT "Yılı"; Trim(Dosya.Yil)
LOCATE 3 + SatirDosya%, SutunVeri%: PRINT "Adı"; Trim(Dosya.Ad)
LOCATE 4 + SatirDosya%, SutunVeri%: PRINT "Sayfa adedi"; Trim(Dosya.Sayfa)
LOCATE 5 + SatirDosya%, SutunVeri%: PRINT "Ücreti"; Trim(Dosya.Ucret)
LOCATE 6 + SatirDosya%, SutunVeri%: PRINT "Basımevi"; Trim(Dosya.Yer)
LOCATE 7 + SatirDosya%, SutunVeri%: PRINT "Basıldığıyer"; Trim(Dosya.BYer)

```

END SUB

```

SUB DosyaYeni
KILL "kod*.dat"
box 9, 21, 13, 41
LOCATE 11, 23: PRINT "Kitap sayısı: 30"
DO WHILE INKEY$ = "": LOOP
'Uzunluk% = 4: Oteleme% = 46: SutunYaz% = Oteleme% + 1: SatirYaz% = 11
'CALL Yazim
'Adet = VAL(Istene$)
Adet = 30

SCREEN 12
PALETTE 1, 4144959
CLOSE #1
OPEN "o", #1, "BarCode.EXM"

FOR Ornek = 1 TO Adet
  Bak$ = ""
  LOCATE 10, 1
  FOR x = 1 TO 200 STEP 10 '20 adet
    Kalinlik = INT(RND * 5)
    x1 = x
    x2 = x + Kalinlik
    y1 = 1
    y2 = 50
    LINE (x1, y1)-(x2, y2), 1, BF
  NEXT x

  Bar = 0
  FOR a = 1 TO 200
    IF POINT(a, 4) = 1 THEN
      x1 = a
      DO UNTIL POINT(a, 4) = 0
        a = a + 1
      LOOP
      x2 = a
      Bar = Bar + 1
      Cizgi(Bar) = x2 - x1
    END IF
  
```

```

NEXT a

PRINT
'ornek dosyasını oluştur
FOR i = 1 TO 20
  PRINT Cizgi(i);
  PRINT #1, LTRIMS(STRS(Cizgi(i))); ", ";
NEXT i
  PRINT #1, "KOD" + LTRIMS(STRS(Ornek))

'her kodu bir dosyaya kaydet
OPEN "o", #2, "KOD" + LTRIMS(STRS(Ornek)) + ".DAT"
FOR i = 1 TO 200
  IF POINT(i, 4) = 1 THEN PRINT #2, LTRIMS(STRS(i)); ", ";
NEXT i
CLOSE #2

CLS
NEXT Ornek
CLOSE
SCREEN 0, 0, 0
menusystem

END SUB

SUB EkrandanTani
ERASE Cizgi
  fancyCls colors(7, ColorPref), colors(1, ColorPref)
  COLOR colors(7, ColorPref), colors(0, ColorPref)

box 5, 11, 10, 40
LOCATE 6, 12: INPUT "Çubuk adedi      : ", CubukAdedi

FOR Aded = 1 TO CubukAdedi

  Bulunamadi = 0
  LOCATE 8, 12: PRINT LTRIMS(STRS(Aded)); ". Çubuğun kalınlığı:  "

  LOCATE 8, 35: INPUT "", Cizgi(Aded)

```

SonucuBul

```

IF Bulunamadi = 0 THEN
    fancyCls colors(7, ColorPref), colors(1, ColorPref)
    box 9, 8, 19, 60
    KodNoS = "KOD" + RTRIMS(LTRIMS(STRS(Sonuc)))
    LOCATE 10, 12: PRINT "Kod Numarası :"; KodNoS
    LOCATE 11, 12: PRINT "_____ "
VeriNo1 = FREEFILE
OPEN "veriler.kod" FOR RANDOM AS VeriNo1 LEN = LEN(Dosya)

FOR DosyaKayit = 1 TO 30
    GET #VeriNo1, DosyaKayit, Dosya
    IF Trim(UCASES(Dosya.Kod)) = Trim(UCASES(KodNoS)) THEN
        CikisYok = 0

        SutunVeri% = 12
        DosyaKodYaz

        CLOSE #VeriNo1
        GOTO Son3
    END IF
NEXT
Son3:
    CLOSE #VeriNo1
    DO WHILE INKEYS = "": LOOP
    EXIT FOR
    END IF
NEXT
IF Bulunamadi = 1 THEN
    fancyCls colors(2, ColorPref), colors(1, ColorPref)
    box 5, 11, 10, 40
    LOCATE 8, 12
    COLOR 24, colors(1, ColorPref)
    PRINT "Bulunamadi...": BEEP
    DO WHILE INKEYS = "": LOOP
    END IF
END SUB

```

'FancyCls:

' Clears screen in the right color, and draws nice dots.

SUB fancyCls (Dots, Background)

VIEW PRINT 2 TO 24

COLOR Dots, Background

CLS 2

FOR a = 95 TO 1820 STEP 45

row = a / 80 + 1

col = a MOD 80 + 1

LOCATE row, col

' PRINT CHR\$(250);

NEXT a

VIEW PRINT

END SUB

'Initialize:

' Read colors in and set up assembly routines

SUB initialize

WIDTH , 25

VIEW PRINT

FOR ColorSet = 1 TO 4

FOR x = 1 TO 10

READ colors(x, ColorSet)

NEXT x

NEXT ColorSet

LoadState

P = VARPTR(ScrollUpAsm(1))

DEF SEG = VARSEG(ScrollUpAsm(1))

FOR i = 0 TO 13

READ J

POKE (P + i), J

```
NEXT i
```

```
P = VARPTR(ScrollDownAsm(1))
```

```
DEF SEG = VARSEG(ScrollDownAsm(1))
```

```
FOR i = 0 TO 13
```

```
  READ J
```

```
  POKE (P + i), J
```

```
NEXT i
```

```
DEF SEG
```

```
END SUB
```

```
SUB KayıtAl
```

```
OkunanS = SPACES(76)
```

```
SELECT CASE indeks
```

```
CASE 1: MIDS(OkunanS, 1) = "Kod Numarası : " + Dosya.Kod
```

```
CASE 2: MIDS(OkunanS, 1) = "Yazarı      : " + Dosya.Yazar
```

```
CASE 3: MIDS(OkunanS, 1) = "Basıldığı yıl : " + Dosya.Yil
```

```
CASE 4: MIDS(OkunanS, 1) = "Kitabın adı  : " + Dosya.Ad
```

```
CASE 5: MIDS(OkunanS, 1) = "Sayfa adedi  : " + Dosya.Sayfa
```

```
CASE 6: MIDS(OkunanS, 1) = "Fiyatı      : " + Dosya.Ucret
```

```
CASE 7: MIDS(OkunanS, 1) = "Basıldığı yer : " + Dosya.Yer
```

```
CASE 8: MIDS(OkunanS, 1) = "Ülkesi      : " + Dosya.BYer
```

```
END SELECT
```

```
END SUB
```

```
SUB KuralSeti
```

```
fancyCls colors(7, ColorPref), colors(1, ColorPref)
```

```
VIEW PRINT (3) TO 24
```

```
PRINT "IF Cizgi(1) = 4 AND Cizgi(4) = 2 THEN Sonuc = 1"
```

```
PRINT "IF Cizgi(1) = 3 AND Cizgi(4) = 3 THEN Sonuc = 2"
```

```
PRINT "IF Cizgi(1) = 1 AND Cizgi(2) = 5 THEN Sonuc = 3"
```

```
PRINT "IF Cizgi(1) = 3 AND Cizgi(9) = 1 THEN Sonuc = 4"
```

```
PRINT "IF Cizgi(3) = 5 THEN Sonuc = 5"
```

```
PRINT "IF Cizgi(1) = 2 AND Cizgi(3) = 1 THEN Sonuc = 6"
```

```
PRINT "IF Cizgi(1) = 1 AND Cizgi(7) = 4 THEN Sonuc = 7"
```

```

PRINT "IF Cizgi(1) = 3 AND Cizgi(5) = 2 THEN Sonuc = 8"
PRINT "IF Cizgi(1) = 4 AND Cizgi(2) = 2 THEN Sonuc = 9"
PRINT "IF Cizgi(1) = 5 AND Cizgi(2) = 4 THEN Sonuc = 10"
PRINT "IF Cizgi(1) = 3 AND Cizgi(9) = 3 THEN Sonuc = 11"
PRINT "IF Cizgi(1) = 2 AND Cizgi(4) = 5 THEN Sonuc = 12"
PRINT "IF Cizgi(1) = 1 AND Cizgi(3) = 1 THEN Sonuc = 13"
PRINT "IF Cizgi(1) = 5 AND Cizgi(2) = 5 THEN Sonuc = 14"
PRINT "IF Cizgi(1) = 2 AND Cizgi(8) = 5 THEN Sonuc = 15"
PRINT "IF Cizgi(1) = 4 AND Cizgi(2) = 1 THEN Sonuc = 16"
PRINT "IF Cizgi(1) = 3 AND Cizgi(6) = 5 THEN Sonuc = 17"
PRINT "IF Cizgi(1) = 3 AND Cizgi(5) = 5 THEN Sonuc = 18"
DO WHILE INKEYS = "": LOOP
PRINT "IF Cizgi(1) = 2 AND Cizgi(2) = 4 THEN Sonuc = 19"
PRINT "IF Cizgi(1) = 2 AND Cizgi(7) = 2 THEN Sonuc = 20"
PRINT "IF Cizgi(1) = 1 AND Cizgi(3) = 3 THEN Sonuc = 21"
PRINT "IF Cizgi(1) = 1 AND Cizgi(5) = 5 THEN Sonuc = 22"
PRINT "IF Cizgi(1) = 2 AND Cizgi(3) = 3 THEN Sonuc = 23"

```

```

PRINT "IF Cizgi(1) = 1 AND Cizgi(6) = 2 THEN Sonuc = 24"
PRINT "IF Cizgi(1) = 4 AND Cizgi(6) = 3 THEN Sonuc = 25"
PRINT "IF Cizgi(1) = 4 AND Cizgi(3) = 1 THEN Sonuc = 26"
PRINT "IF Cizgi(1) = 3 AND Cizgi(4) = 4 THEN Sonuc = 27"
PRINT "IF Cizgi(1) = 4 AND Cizgi(7) = 4 THEN Sonuc = 28"
PRINT "IF Cizgi(12) = 1 THEN Sonuc = 29"
PRINT "IF Cizgi(19) = 4 THEN Sonuc = 30"

```

END SUB

SUB Listele

```

fancyCls colors(7, ColorPref), colors(1, ColorPref)
VIEW PRINT (2) TO 24
DosyaNumara = FREEFILE
OPEN "Veriler.kod" FOR RANDOM AS #DosyaNumara LEN = LEN(Dosya)

```

Satirlar = 1: indcks = 0: Sutunlar = 1

```

FOR KayitNumara = 1 TO 30
    GET #DosyaNumara, KayitNumara, Dosya
    FOR indcks = 1 TO 9

```

```

Satirlar = Satirlar + 1
    IF Satirlar = 24 THEN
        Satirlar = 1
        DO WHILE INKEYS = "": LOOP
    END IF

    KayitAJ
    LOCATE Satirlar, Sutunlar

    PRINT OkunanS
    NEXT
    'DO WHILE INKEYS = "": LOOP
NEXT
CLOSE #DosyaNumara
END SUB

'LoadState:
' Load color preferences and account info from MONEY.DAT
SUB LoadState

    OPEN "money.dat" FOR INPUT AS #1
    INPUT #1, ColorPref

    FOR a = 1 TO 19
        LINE INPUT #1, account(a).Title
        LINE INPUT #1, account(a).AType
        LINE INPUT #1, account(a).Desc
    NEXT a

    CLOSE

END SUB

' currChoiceX : Number of current choice
' maxChoice   : Number of choices in the list
' choiceS()   : Array with the text of the choices
' itemRow()   : Array with the row of the choices
' itemCol()   : Array with the col of the choices
' helpS()     : Array with the help text for each choice

```

```

' barMode : Boolean: TRUE = menu bar style, FALSE = drop down style
'
FUNCTION Menu (CurrChoiceX, MaxChoice, choiceS(), itemRow(), itemcol(), helpS(), BarMode)
    currChoice = CurrChoiceX
    IF BarMode THEN
        COLOR colors(7, ColorPref), colors(4, ColorPref)
        LOCATE 1, 1
        PRINT SPACES(80);
    ELSE
        fancyCIs colors(2, ColorPref), colors(1, ColorPref)
        COLOR colors(7, ColorPref), colors(4, ColorPref)
        box itemRow(1) - 1, itemcol(1) - 1, itemRow(MaxChoice) + 1, itemcol(1) + LEN(choiceS(1)) +
1

        COLOR colors(10, ColorPref), colors(6, ColorPref)
        FOR a = 1 TO MaxChoice + 1
            LOCATE itemRow(1) + a - 1, itemcol(1) + LEN(choiceS(1)) + 2
            PRINT CHR$(178); CHR$(178);
        NEXT a
        LOCATE itemRow(MaxChoice) + 2, itemcol(MaxChoice) + 2
        PRINT STRINGS(LEN(choice$(MaxChoice)) + 2, 178);
    END IF

    'print the choices
    COLOR colors(7, ColorPref), colors(4, ColorPref)
    FOR a = 1 TO MaxChoice
        LOCATE itemRow(a), itemcol(a)
        PRINT choiceS(a);
    NEXT a

    finished = FALSE

    WHILE NOT finished

        GOSUB MenuShowCursor
        GOSUB MenuGetKey
        GOSUB MenuHideCursor

```

```

SELECT CASE Kbd$
  CASE CHR$(0) + "H": GOSUB MenuUp
  CASE CHR$(0) + "P": GOSUB MenuDown
  CASE CHR$(0) + "K": GOSUB MenuLeft
  CASE CHR$(0) + "M": GOSUB MenuRight
  CASE CHR$(13): GOSUB MenuEnter
  CASE CHR$(27): GOSUB MenuEscape
  CASE ELSE: BEEP
END SELECT
WEND

```

```
Menu = currChoice
```

```
EXIT FUNCTION
```

```
MenuEnter:
```

```

finished = TRUE
RETURN

```

```
MenuEscape:
```

```

currChoice = 0
finished = TRUE
RETURN

```

```
MenuUp:
```

```

IF BarMode THEN
  BEEP
ELSE
  currChoice = (currChoice + MaxChoice - 2) MOD MaxChoice + 1
END IF
RETURN

```

```
MenuLeft:
```

```

IF BarMode THEN
  currChoice = (currChoice + MaxChoice - 2) MOD MaxChoice + 1
ELSE
  currChoice = -2
  finished = TRUE
END IF

```

```
RETURN
```

```
MenuRight:
```

```
IF BarMode THEN
```

```
    currChoice = (currChoice) MOD MaxChoice + 1
```

```
ELSE
```

```
    currChoice = -3
```

```
    finished = TRUE
```

```
END IF
```

```
RETURN
```

```
MenuDown:
```

```
IF BarMode THEN
```

```
    finished = TRUE
```

```
ELSE
```

```
    currChoice = (currChoice) MOD MaxChoice + 1
```

```
END IF
```

```
RETURN
```

```
MenuShowCursor:
```

```
COLOR colors(8, ColorPref), colors(9, ColorPref)
```

```
LOCATE itemRow(currChoice), itemcol(currChoice)
```

```
PRINT choiceS(currChoice);
```

```
PrintHelpLine helpS(currChoice)
```

```
RETURN
```

```
MenuGetKey:
```

```
KbdS = ""
```

```
WHILE KbdS = ""
```

```
    KbdS = INKEYS
```

```
WEND
```

```
RETURN
```

```
MenuHideCursor:
```

```
COLOR colors(7, ColorPref), colors(4, ColorPref)
```

```
LOCATE itemRow(currChoice), itemcol(currChoice)
```

```
PRINT choiceS(currChoice);
```

```
RETURN
```

END FUNCTION

SUB menuSystem

DIM choiceS(20), menuRow(20), menuCol(20), helpS(20)

LOCATE , , 0

choice = 1

finished = FALSE

WHILE NOT finished

GOSUB MenuSystemMain

subchoice = -1

WHILE subchoice < 0

SELECT CASE choice

CASE 1: GOSUB MenuSystemFile

CASE 2: GOSUB MenuSystemEdit

CASE 3: GOSUB MenuSystemColors

END SELECT

fancyCls colors(2, ColorPref), colors(1, ColorPref)

SELECT CASE subchoice

CASE -2: choice = (choice + 1) MOD 3 + 1

CASE -3: choice = (choice) MOD 3 + 1

END SELECT

WEND

WEND

EXIT SUB

MenuSystemMain:

fancyCls colors(2, ColorPref), colors(1, ColorPref)

COLOR colors(7, ColorPref), colors(4, ColorPref)

choiceS(1) = " Barkod "

choiceS(2) = " Veri/Bilgi"

choiceS(3) = " Renkler "

DosyaKol% = 2: TanimaKol% = 12: RenkKol% = 22

menuRow(1) = 1: menuCol(1) = DosyaKol%

menuRow(2) = 1: menuCol(2) = TanimaKol%

menuRow(3) = 1: menuCol(3) = RenkKol%

```
helpS(1) = "Barkod Üretir ve İlgili Bilgileri bir dosyaya kaydeder"
```

```
helpS(2) = " Veri/Bilgitabanlarınıgösterir "
```

```
helpS(3) = " Ekran renkleri gösterir "
```

```
DO
```

```
    NewChoice = Menu((choice). 3, choiceS(), menuRow(), menuCol(), helpS(), TRUE)
```

```
LOOP WHILE NewChoice = 0
```

```
choice = NewChoice
```

```
RETURN
```

```
MenuSystemFile:
```

```
choiceS(1) = " BarkodÜret "
```

```
choiceS(2) = " BarkodTanı "
```

```
choiceS(3) = " _____ "
```

```
choiceS(4) = " Yazdır "
```

```
choiceS(5) = " DOS Shell "
```

```
choiceS(6) = " _____ "
```

```
choiceS(7) = " Çıkış "
```

```
menuRow(1) = 3: menuCol(1) = DosyaKol%
```

```
menuRow(2) = 4: menuCol(2) = DosyaKol%
```

```
menuRow(3) = 5: menuCol(3) = DosyaKol%
```

```
menuRow(4) = 6: menuCol(4) = DosyaKol%
```

```
menuRow(5) = 7: menuCol(5) = DosyaKol%
```

```
menuRow(6) = 8: menuCol(6) = DosyaKol%
```

```
menuRow(7) = 9: menuCol(7) = DosyaKol%
```

```
helpS(1) = " Barkodlar Üretilir ve Üretilen barkodların İlgili bilgileri bir dosyaya kaydedilir"
```

```
helpS(2) = "Kayıtlı dosyadaki bilgiler Bilgi tabanı ile karşılatırılarak Barkod tanınır"
```

```
helpS(3) = " "
```

```
helpS(4) = "Veritabanının yazıcıdan çıkışını verir"
```

```
helpS(5) = "Geçici olarak DOS'a çıkılır"
```

```
helpS(6) = " "
```

```
helpS(7) = "Programdan çıkılır"
```

```
subchoice = Menu(1, 7. choiceS(), menuRow(), menuCol(), helpS(). FALSE)
```

```
SELECT CASE subchoice
```

CASE 1

DosyaYeni

CASE 2

DosyaAc

CASE 4

Yazdir

CASE 5

COLOR 7, 0: CLS

SHELL "command.com"

CASE 7

finished = TRUE

END

CASE ELSE

END SELECT

RETURN

MenuSystemEdit:

choiceS(1) = " Ekrandan tanıma "

choiceS(2) = " KodNosuna göre tanıma"

choiceS(3) = " Bilgi tabanı Göster "

choiceS(4) = " Veritabanını Listeleme "

menuRow(1) = 3: menuCol(1) = TanimaKol%

menuRow(2) = 4: menuCol(2) = TanimaKol%

menuRow(3) = 5: menuCol(3) = TanimaKol%

menuRow(4) = 6: menuCol(4) = TanimaKol%

helpS(1) = "Çizgi kalınlıklarıEkrandan girilerek ilgili çizgikodunu ve bilgilerini verir"

helpS(2) = "Kod nosu girilen Kaydın çizgikodu ve veritabanınıgösterir"

helpS(3) = "RULES3 Algoritmasıile çıkarılan Kural setini gösterir"

helpS(4) = "Tüm Kitapların veritabanı çizgikodlarıyla beraber listelenir"

subchoice = Menu(1, 4, choiceS(), menuRow(), menuCol(), helpS(), FALSE)

SELECT CASE subchoice

```

CASE 1
EkrandanTani
CASE 2
Veritabani
CASE 3
KuralSeti
DO WHILE INKEYS = "": LOOP
CASE 4
Listele

END SELECT
RETURN

```

MenuSystemColors:

```

choiceS(1) = " Siyah/Beyaz Şema "
choiceS(2) = " AçıkYeşil/Mavi Şema"
choiceS(3) = " Mavi/AçıkYeşil Şema"
choiceS(4) = " Kırmızı/Gri Şema "

menuRow(1) = 3: menuCol(1) = RenkKol%
menuRow(2) = 4: menuCol(2) = RenkKol%
menuRow(3) = 5: menuCol(3) = RenkKol%
menuRow(4) = 6: menuCol(4) = RenkKol%

helpS(1) = "Monokrom ve LCD ekranlar için Siyah ve Beyaz"
helpS(2) = "Mavi Renk zemin üzerine AçıkYeşil"
helpS(3) = "AçıkYeşil Renk Zemin üzerine Mavi "
helpS(4) = "Gri Renk üzerine Kırmızı"

subchoice = Menu(1, 4, choiceS(), menuRow(), menuCol(), helpS(), FALSE)

SELECT CASE subchoice
CASE 1 TO 4
ColorPref = subchoice
SaveState
CASE ELSE
END SELECT
RETURN

```

END SUB

'PrintHelpLine:

' Prints help text on the bottom row in the proper color

SUB PrintHelpLine (help\$)

 COLOR colors(5, ColorPref), colors(4, ColorPref)

 LOCATE 25, 1

 PRINT SPACES(80);

 Center 25, help\$

END SUB

'SaveState:

' Save color preference and account information to "MONEY.DAT" data file.

SUB SaveState

 OPEN "money.dat" FOR OUTPUT AS #2

 PRINT #2, ColorPref

 FOR a = 1 TO 19

 PRINT #2, account(a).Title

 PRINT #2, account(a).AType

 PRINT #2, account(a).Desc

 NEXT a

 CLOSE #2

END SUB

SUB SonucuBul

 IF Cizgi(1) = 4 AND Cizgi(4) = 2 THEN Sonuc = 1: EXIT SUB

 IF Cizgi(1) = 3 AND Cizgi(4) = 3 THEN Sonuc = 2: EXIT SUB

 IF Cizgi(1) = 1 AND Cizgi(2) = 5 THEN Sonuc = 3: EXIT SUB

 IF Cizgi(1) = 3 AND Cizgi(9) = 1 THEN Sonuc = 4: EXIT SUB

 IF Cizgi(3) = 5 THEN Sonuc = 5: EXIT SUB

 IF Cizgi(1) = 2 AND Cizgi(3) = 1 THEN Sonuc = 6: EXIT SUB

 IF Cizgi(1) = 1 AND Cizgi(7) = 4 THEN Sonuc = 7: EXIT SUB

 IF Cizgi(1) = 3 AND Cizgi(5) = 2 THEN Sonuc = 8: EXIT SUB

 IF Cizgi(1) = 4 AND Cizgi(2) = 2 THEN Sonuc = 9: EXIT SUB

 IF Cizgi(1) = 5 AND Cizgi(2) = 4 THEN Sonuc = 10: EXIT SUB

```

IF Cizgi(1) = 3 AND Cizgi(9) = 3 THEN Sonuc = 11: EXIT SUB
IF Cizgi(1) = 2 AND Cizgi(4) = 5 THEN Sonuc = 12: EXIT SUB
IF Cizgi(1) = 1 AND Cizgi(3) = 1 THEN Sonuc = 13: EXIT SUB
IF Cizgi(1) = 5 AND Cizgi(2) = 5 THEN Sonuc = 14: EXIT SUB
IF Cizgi(1) = 2 AND Cizgi(8) = 5 THEN Sonuc = 15: EXIT SUB
IF Cizgi(1) = 4 AND Cizgi(2) = 1 THEN Sonuc = 16: EXIT SUB
IF Cizgi(1) = 3 AND Cizgi(6) = 5 THEN Sonuc = 17: EXIT SUB
IF Cizgi(1) = 3 AND Cizgi(5) = 5 THEN Sonuc = 18: EXIT SUB
IF Cizgi(1) = 2 AND Cizgi(2) = 4 THEN Sonuc = 19: EXIT SUB
IF Cizgi(1) = 2 AND Cizgi(7) = 2 THEN Sonuc = 20: EXIT SUB
IF Cizgi(1) = 1 AND Cizgi(3) = 3 THEN Sonuc = 21: EXIT SUB
IF Cizgi(1) = 1 AND Cizgi(5) = 5 THEN Sonuc = 22: EXIT SUB
IF Cizgi(1) = 2 AND Cizgi(3) = 3 THEN Sonuc = 23: EXIT SUB
IF Cizgi(1) = 1 AND Cizgi(6) = 2 THEN Sonuc = 24: EXIT SUB
IF Cizgi(1) = 4 AND Cizgi(6) = 3 THEN Sonuc = 25: EXIT SUB
IF Cizgi(1) = 4 AND Cizgi(3) = 1 THEN Sonuc = 26: EXIT SUB
IF Cizgi(1) = 3 AND Cizgi(4) = 4 THEN Sonuc = 27: EXIT SUB
IF Cizgi(1) = 4 AND Cizgi(7) = 4 THEN Sonuc = 28: EXIT SUB
IF Cizgi(12) = 1 THEN Sonuc = 29: EXIT SUB
IF Cizgi(19) = 4 THEN Sonuc = 30: EXIT SUB
Bulunamadi = 1
END SUB

```

'Trim\$:

' Remove null and spaces from the end of a string.

FUNCTION TrimS (x\$)

```

IF x$ = "" THEN
    Trim$ = ""
ELSE
    lastChar = 0
    FOR a = 1 TO LEN(x$)
        y$ = MIDS(x$, a, 1)
        IF y$ <> CHR$(0) AND y$ <> " " THEN
            lastChar = a
        END IF
    NEXT a
    Trim$ = LEFTS(x$, lastChar)
END IF

```

END FUNCTION

SUB VeriKodYaz

SutunVeri% = 21

LOCATE 8, SutunVeri%: PRINT Dosya.Yazar

LOCATE 9, SutunVeri%: PRINT Dosya.Yil

LOCATE 10, SutunVeri%: PRINT Dosya.Ad

LOCATE 11, SutunVeri%: PRINT Dosya.Sayfa

LOCATE 12, SutunVeri%: PRINT Dosya.Ucret

LOCATE 13, SutunVeri%: PRINT Dosya.Yer

LOCATE 14, SutunVeri%: PRINT Dosya.BYer

DO WHILE INKEYS = "": LOOP

END SUB

SUB Veritabani

fancyCls colors(2, ColorPref), colors(1, ColorPref)

box 6, 6, 15, 72

LOCATE 7, 7: PRINT "Kod :"

LOCATE 8, 7: PRINT "Yazar :"

LOCATE 9, 7: PRINT "Basım yılı :"

LOCATE 10, 7: PRINT "Kitabın adı:"

LOCATE 11, 7: PRINT "Sayfa sayısı:"

LOCATE 12, 7: PRINT "Ücreti :"

LOCATE 13, 7: PRINT "Basım yeri :"

LOCATE 14, 7: PRINT "Ülkesi :"

VeriNo = FREEFILE

OPEN "veriler.kod" FOR RANDOM AS VeriNo LEN = LEN(Dosya)

KodBas:

BirOnceki = 0: BitirYaz = 0

Uzunluk% = 6: Oteleme% = 20: SutunYaz% = Oteleme% + 1: SatirYaz% = 7

CALL Yazim

IF BitirYaz = 1 THEN GOTO Son

IF BirOnceki = 1 THEN GOTO KodBas

'Dosya Kod = Istenen\$

DosyaKayitTMP = DosyaKayit 'Şu anda kullanılmıyor

```

FOR DosyaKayit = 1 TO 30
GET #VeriNo, DosyaKayit, Dosya
IF Trim(UCASES(Dosya Kod)) = Trim(UCASES(IsteneS)) THEN
  CikisYok = 0
  VeriKodYaz
  CLOSE #VeriNo
  GOTO Son
ELSE
  CikisYok = 0
END IF
IF CikisYok = 1 THEN
  CLOSE #VeriNo
  GOTO Son
END IF
NEXT

```

YazarBas:

```

BirOnceki = 0: BitirYaz = 0
Uzunluk% = 20: Oteleme% = 20: SutunYaz% = Oteleme% + 1: SatirYaz% = 8
CALL Yazim
IF BitirYaz = 1 THEN GOTO Son
IF BirOnceki = 1 THEN GOTO KodBas
Dosya.Yazar = IsteneS

```

YilBas:

```

BirOnceki = 0: BitirYaz = 0
Uzunluk% = 4: Oteleme% = 20: SutunYaz% = Oteleme% + 1: SatirYaz% = 9
CALL Yazim
IF BitirYaz = 1 THEN GOTO Son
IF BirOnceki = 1 THEN GOTO YazarBas
Dosya.Yil = IsteneS

```

AdBas:

```

BirOnceki = 0: BitirYaz = 0
Uzunluk% = 50: Oteleme% = 20: SutunYaz% = Oteleme% + 1: SatirYaz% = 10
CALL Yazim
IF BitirYaz = 1 THEN GOTO Son
IF BirOnceki = 1 THEN GOTO YilBas
Dosya.Ad = IsteneS

```

SayfaBas:

BirOnceki = 0: BitirYaz = 0

Uzunluk% = 5: Oteleme% = 20: SutunYaz% = Oteleme% + 1: SatirYaz% = 11

CALL Yazim

IF BitirYaz = 1 THEN GOTO Son

IF BirOnceki = 1 THEN GOTO AdBas

Dosya.Sayfa = IstenenS

UcretBas:

BirOnceki = 0: BitirYaz = 0

Uzunluk% = 15: Oteleme% = 20: SutunYaz% = Oteleme% + 1: SatirYaz% = 12

CALL Yazim

IF BitirYaz = 1 THEN GOTO Son

IF BirOnceki = 1 THEN GOTO SayfaBas

Dosya.Ucret = IstenenS

YerBas:

BirOnceki = 0: BitirYaz = 0

Uzunluk% = 15: Oteleme% = 20: SutunYaz% = Oteleme% + 1: SatirYaz% = 13

CALL Yazim

IF BitirYaz = 1 THEN GOTO Son

IF BirOnceki = 1 THEN GOTO UcretBas

Dosya.Yer = IstenenS

BYerBas:

BirOnceki = 0: BitirYaz = 0

Uzunluk% = 15: Oteleme% = 20: SutunYaz% = Oteleme% + 1: SatirYaz% = 14

CALL Yazim

IF BitirYaz = 1 THEN GOTO Son

IF BirOnceki = 1 THEN GOTO YerBas

Dosya.BYer = IstenenS

Son:

IF Bitir = 0 THEN

'DosyaKayit = DosyaKayit + 1

'PUT #VeriNo, DosyaKayit, Dosya

'END IF

END SUB

```

SUB Yazdir
YaziciHata = 0
DosyaNumara = FREEFILE
OPEN "Veriler.kod" FOR RANDOM AS #DosyaNumara LEN = LEN(Dosya)

FOR KayitNumara = 1 TO 30
    GET #DosyaNumara, KayitNumara, Dosya
    FOR indeks = 1 TO 9
        KayitAl
        LPRINT Okunan$
        IF YaziciHata THEN EXIT SUB
    NEXT
NEXT

CLOSE #DosyaNumara

END SUB

SUB Yazim
'COLOR 7, 0
DIM Secilmis%(20)
HazirAlan$ = SPACES(Uzunluk%): Istenen$ = HazirAlan$
COLOR colors(7, ColorPref), colors(4, ColorPref)

LOCATE SatirYaz%, SutunYaz%
PRINT HazirAlan$

LOCATE SatirYaz%, SutunYaz%, 1

Satira% = CSRLIN
Sutun% = POS(0)

Harf% = 0: Sutun% = Harf% + Oteleme%
Tekrar:
    Harf% = Harf% + 1: Sutun% = Harf% + Oteleme%
LOCATE Satira%, Sutun%
DO
    Tus$ = INKEYS

```

```

IF Tus$ = "" THEN Tus$ = ""
LOOP WHILE Tus$ = ""
SELECT CASE Tus$
CASE CHR$(0) + "M"           'Sağ
  IF Harf% = Uzunluk% THEN GOTO Bit
  GOTO Tekrar
CASE CHR$(0) + "S"           'DEL
  FOR N% = Harf% TO Uzunluk%
    MIDS(IsteneS, N%, 1) = MIDS(IsteneS, N% + 1, 1)
    Sutun% = N% + Oteleme%
    LOCATE Satira%, Sutun%
    PRINT MIDS(IsteneS, N%, 1)
  NEXT
  Sutun% = Harf% + Oteleme%
  LOCATE Satira%, Sutun%
  PRINT MIDS(IsteneS, N%, 1)
  Harf% = Harf% - 1
  GOTO Tekrar

CASE " _"                     'Backspace
  IF Harf% > 1 THEN
    FOR N% = Harf% - 1 TO Uzunluk%
      MIDS(IsteneS, N%, 1) = MIDS(IsteneS, N% + 1, 1)
      Sutun% = N% + Oteleme%
      LOCATE Satira%, Sutun%
      PRINT MIDS(IsteneS, N%, 1)
    NEXT
    Sutun% = Harf% + Oteleme%
    LOCATE Satira%, Sutun%
    PRINT MIDS(IsteneS, N%, 1)
    Harf% = Harf% - 2
  ELSEIF Harf% = 1 THEN
    Harf% = Harf% - 1
  END IF
  GOTO Tekrar

CASE CHR$(0) + "-"           'Caps Lock
Harf = Harf - 1
GOTO Tekrar

```

```

CASE CHR$(0) + "H"           'Yukarı
  BirOnceki = 1
  GOTO Cikis

CASE CHR$(0) + "K"           'Sol
  IF Harf% > 0 THEN Harf% = Harf% - 2
  IF Harf% < 0 THEN Harf% = 0
  GOTO Tekrar

CASE CHR$(27)                'ESCAPE
  BitirYaz = 1
  GOTO Cikis

CASE CHR$(0) + "P"           'Aağı
'COLOR 7, 6
LOCATE SatirYaz%, SutunYaz%: PRINT IstenenS
Sonraki% = 1

CASE CHR$(9)                 'TAB
'COLOR 8, 6
LOCATE SatirYaz%, SutunYaz%: PRINT IstenenS

CASE CHR$(13)
Sonraki% = 1

CASE ELSE
  PRINT Tus$
  MIDS(IstenenS, Harf%, 1) = Tus$
  IF Harf% = Uzunluk% THEN
    GOTO Bit
  ELSE
    GOTO Tekrar
  END IF
Bit:
END SELECT
Cikis:

END SUB

```

