

**ANKARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

YÜKSEK LİSANS TEZİ

GERÇEK ZAMANLI VİDEOLARDA ÖN PLAN VE ARKA PLAN AYRIMI

Ahmet BUĞDAY

ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

**ANKARA
2010**

Her hakkı saklıdır

TEZ ONAYI

Ahmer BUĞDAY tarafından hazırlanan “**Gerçek Zamanlı Videolarda Ön Plan ve Arka Plan Ayrımı**” adlı tez çalışması 20/05/2010 tarihinde aşağıdaki jüri tarafından oy birliği ile Ankara Üniversitesi Fen Bilimleri Enstitüsü Elektronik Mühendisliği Anabilim Dalı'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Danışman : Yrd.Doç.Dr. Hakkı Alparslan ILGIN

Jüri Üyeleri

Başkan :Yrd.Doç.Dr. Süleyman TOSUN
Ankara Üniversitesi
Bilgisayar Mühendisliği Anabilim Dalı

Üye :Yrd. Doç. Dr. Fikret ARI
Ankara Üniversitesi
Elektronik Mühendisliği Anabilim Dalı

Üye : Yrd. Doç. Dr. Hakkı Alparslan ILGIN
Ankara Üniversitesi
Elektronik Mühendisliği Anabilim Dalı

Yukarıdaki sonucu onaylarım

Prof. Dr.Orhan ATAKOL

Enstitü Müdürü

ÖZET

Yüksek Lisans Tezi

GERÇEK ZAMANLI VİDEOLARDA ÖN PLAN VE ARKA PLAN AYRIMI

Ahmet BUĞDAY

Ankara Üniversitesi

Fen Bilimleri Enstitüsü

Elektronik Mühendisliği Ana Bilim Dalı

Danışman : Yrd. Doç. Dr. Hakkı Alparslan ILGIN

Görsel video uygulamalarının temel ve kritik çalışma alanlarından biri video dizilerindeki hareket eden ön plan nesnelere tanınması ve belirlenmesidir. Hareket eden ön plan nesnelere belirlemek için arka plan çıkarımı en çok kullanılan yöntemlerin başında gelir. Arka plan çıkarımı yönteminin kullanılabilmesi için arka plan modelinin bilinmesi ya da değişen arka plan görüntüleri için arka plan modelinin dinamik olarak oluşturulması gerekir. Bu yüksek lisans tez çalışmasında ön plan arka plan ayırımında kullanılan yöntemler anlatılmıştır. Yöntemler anlatılırken bellek kullanımı, gerçek zamanlı uygulanabilirliği ve doğru sonuç üretmeleri açısından karşılaştırılmışlardır. Diğer yöntemlere nazaran kod çizelgesi üzerinde daha fazla durulmuştur. Kod çizelgesi yönteminin tercih edilmesinin sebebi etkin bellek kullanımı, hızlı sonuç üretimi, dış ortam şartlarında çalışabilmesi gibi özellikleridir. Bunların dışında kod çizelgesi yöntemi değişen arka plan şartlarına göre arka plan modelini güncelleyerek doğru sonuçlar elde edilmesini sağlar. Kod çizelgesi yöntemindeki işlem sayısı azaltılarak geliştirilen kutu tabanlı kod çizelgesi yöntemine bağlı bileşenler analizi, gölge ve ışık maskeleri eklenerek bu yöntem geliştirilmiştir. Deneysel çalışmalar sonucunda geliştirilen kutu tabanlı kod çizelgesi yönteminin ön plan arka plan ayırımını başarılı ve hızlı bir şekilde yaptığı görülmüştür.

Mayıs 2010, 76 sayfa

Anahtar Kelimeler: Ön plan, Arka plan, Arka plan çıkarımı, Ön plan bulunması, Kod çizelgesi

ABSTRACT

Master Thesis

SEGMENTATION OF FOREGROUND AND BACKGROUND OF REAL TIME VIDEO STREAMS

Ahmet BUĞDAY

Ankara University

Graduate School of Natural and Applied Sciences

Department of Electronic Engineering

Supervisor : Asst. Prof. Dr. Hakkı Alparslan ILGIN

One of the basic and critical work area are of video surveillance is, determining and identifying the moving foreground objects in video sequences. Background subtraction is the most used method identifying the moving foreground objects. Background subtraction method can be used if only background model is known or background model must be constructed dynamically. Fore, this study describes the methods used in the differentiation of the background and foreground in videos. While the methods are being described, methods' memory usage, real-time applicability and to produce accurate results have been compared. Compared to other methods more focused on the codebook method. Codebook is preferred because of the the features such as efficient memory usage, faster production of results, work in outdoor conditions. In addition, the codebook method updates the background model according to changing background conditions to produce accurate results. A new box-based codebook method was developed by reducing the computation complexity. Box-based codebook model has been improved by adding connected component analysis, shadow and highlight detection masks. Experimental results showed that improved box-based codebook method is fast and successful in foreground and background segmentation.

May 2010, 76 pages

Key Words: Foreground, Background, Background Subtraction, Foreground Detection, Codebook

TEŐEKKÜR

Çalıőmalarımı yönlendiren, araőtırmalarımın her aőamasında bilgi, öneri ve yardımları ile bana destek veren danıőmanım Yrd. Doç. Dr. Hakkı Alparslan ILGIN'a (Ankara Üniversitesi Elektronik Mühendisliđi Bölümü), sadece yüksek lisans eđitimimle kalmayıp tüm eđitim hayatım boyunca desteklerini yanımda hissettiđim aileme, iő hayatım süresince yüksek lisans eđitimimi destekleyen çalıőma arkadaşlarıma teőekkür ederim.

Ayrıca tez çalıőmalarım sırasında destek olan eőim Zülal BUĐDAY'a teőekkür ederim.

Ahmet BUĐDAY
Ankara, Mayıs 2010

İÇİNDEKİLER

ÖZET.....	i
ABSTRACT.....	ii
TEŞEKKÜR	iii
SİMGELER DİZİNİ	vi
ŞEKİLLER DİZİNİ	vii
ÇİZELGELER DİZİNİ.....	ix
1. GİRİŞ.....	1
2. VİDEOYU OLUŞTURAN BİLEŞENLER.....	3
2.1 Piksel	3
2.2 Çerçeve.....	3
2.2.1 Çözünürlük	4
2.2.2 Renk kanalları	5
2.2.3 Renk uzayları.....	5
2.2.3.1 Gri ölçekli görüntüler	5
2.2.3.2 RGB renk uzayı.....	6
2.2.3.3 YUV renk uzayı.....	7
2.2.3.4 YIQ renk uzayı.....	8
2.2.3.5 YCbCr renk uzayı.....	9
2.2.3.6 HSI, HLS ve HSV renk uzayları.....	9
3. VİDEO	12
4. VİDEOLARDA ÖN PLAN VE ARKA PLAN AYRIMI	15
4.1 Ön Plan - Arka Plan Ayrımında Kullanılan Algoritmalar	15
4.1.1 Önışleme	17
4.1.2 Arka plan modellemesi	18
4.1.2.1 Özyinelemesiz yöntemler.....	20
4.1.2.1.1 Çerçeve farkı.....	20
4.1.2.1.2 Ortanca süzgeç yöntemi	21
4.1.2.1.3 Doğrusal tahmin edici süzgeç yöntemi.....	22
4.1.2.1.4 Parametrik olmayan model yöntemi.....	22
4.1.2.2 Özyinelemeli yöntemler	22
4.1.2.2.1 Yaklaşık ortanca süzgeç yöntemi	23
4.1.2.2.2 Kalman süzgeci yöntemi	23
4.1.2.2.3 Gauss fonksiyonları karışımı yöntemi	24
4.1.2.2.4 Özarkaplan yöntemi	27
4.1.3 Ön plan bulunması	27
4.1.4 Veri doğrulaması	28
5. GERÇEK ZAMANLI VİDEOLARDA KOD ÇİZELGESİ YÖNTEMİ KULLANARAK ÖN PLAN ARKA PLAN AYRIMI	30
5.1 Kod Çizelgesi Yöntemi	30
5.1.1 Arka plan modelinin oluşturulması (Kod çizelgesinin oluşturulması).....	33
5.1.2 Gereksiz kod sözcüklerinin kod çizelgesinden çıkarılması.....	35
5.1.3 Kod çizelgesi yönteminde ön plan arka plan ayrımı	36
5.1.4 Kod çizelgesi yönteminde arka plan modelinin güncellenmesi	37
5.2 Kutu Tabanlı Kod Çizelgesi Yöntemi	38

5.2.1	Kutu tabanlı kod çizelgesi yönteminde arka plan modelinin oluşturulması (Kutu tabanlı kod çizelgesinin oluşturulması)	39
5.2.2	Kutu tabanlı kod çizelgesi yönteminde gereksiz kod sözcüklerinin kod çizelgesinden çıkarılması	42
5.2.3	Kutu tabanlı kod çizelgesi yönteminde ön plan arka plan ayrımı	42
5.2.4	Kutu tabanlı kod çizelgesi yönteminde arka plan modelinin güncellenmesi	43
5.3	Geliştirilen Kutu Tabanlı Kod Çizelgesi Yöntemi	44
5.3.1	Geliştirilen kutu tabanlı kod çizelgesi yönteminde arka plan modelinin oluşturulması	45
5.3.2	Geliştirilen kutu tabanlı kod çizelgesi yönteminde gereksiz kod sözcüklerinin kod çizelgesinden atılması.....	47
5.3.3	Geliştirilen kutu tabanlı kod çizelgesi yönteminde ön plan arka plan ayrımı.....	49
5.3.3.1	Gölge maskesi	50
5.3.3.2	Işık maskesi	52
5.3.4	Geliştirilen kutu tabanlı kod çizelgesi yönteminde bağlı bileşenler analizi ..	54
6.	DENEYSEL ÇALIŞMALAR	57
7.	SONUÇLAR.....	70
	KAYNAKLAR	72
	ÖZGEÇMİŞ.....	76

SİMGELER DİZİNİ

FPS	Frame Per Second (Saniyedeki Çerçeve Sayısı)
GM	Gölge Maskesi
HLS	Hue Luminance Saturation (Renk özü, Parlaklık, Renk doygunluğu)
HSI	Hue Saturation Intensity (Renk özü, Renk doygunluğu, Işıklılık)
HSV	Hue Saturation Value (Renk özü, Renk doygunluğu, Parlaklık)
IM	Işık Maskesi
KÇ	Kod Çizelgesi
MoG	Mixture of Gaussian (Gauss Fonksiyonları Karışımı)
NTSC	National Television System Committee (Ulusal Televizyon Sistemi Komitesi)
PAL	Phase Alternating Line (Fazı Dönüşümlü Değiştiren Hat)
RGB	Red Green Blue (Kırmızı, Yeşil, Mavi)
SECAM	Sequential Color with Memory (Bellekli Sıralı Renk)
YCbCr	Luminance Chrominance1 Chrominance2 (Işıklılık, Renk işareti1, Renk işareti2)
YIQ	Luminance Chrominance1 Chrominance2 (Işıklılık, Renk işareti1, Renk işareti2)
YUV	Luminance Chrominance1 Chrominance2 (Işıklılık, Renk işareti1, Renk işareti2)

ŞEKİLLER DİZİNİ

Şekil 2.1 256x256 piksel	4
Şekil 2.2 128x128 piksel	4
Şekil 2.3 64x64 piksel	4
Şekil 2.4 32x32 piksel.....	4
Şekil 2.5 RGB renk uzayında görüntünün ayrı ayrı görünümü	7
Şekil 2.6 UV renk bileşenleri	8
Şekil 2.7 I ve Q renk bileşenleri.....	8
Şekil 2.8 HLS renk uzayı	9
Şekil 2.9 Altıgen huni şeklinde HSV renk uzayı	10
Şekil 2.10 çift altıgen huni şeklinde HSI renk modeli	11
Şekil 3.1 Video, bir grup hareketsiz görüntü	13
Şekil 3.2 Video oluşturan bileşenler	14
Şekil 4.1 Ön plan, arka plan ayrıştırılmasında kullanılan algoritmaların genel gösterimi	16
Şekil 5.1 Kod çizelgesinde renk değişimi ve parlaklık hesaplamasında kullanılan şekil (Kim vd. 2005)	34
Şekil 5.2 Kod çizelgesi yönteminin çalışma prensibinin gösterimi	42
Şekil 5.3 Kutu tabanlı kod çizelgesinin mantıksal bellek gösterimi	47
Şekil 5.4 Kod çizelgesindeki gereksiz kod sözcüklerinin çıkarılması	48
Şekil 5.5 Morfolojik süzgeç örnekleri.....	56
Şekil 6.1 Video1 için çerçeve farkı yöntemiyle elde edilen sonuçlar	59
Şekil 6.2 Video1 için Gauss karışımları yöntemiyle elde edilen sonuçlar.....	60
Şekil 6.3 Video1 için kutu tabanlı kod çizelgesi yöntemiyle elde edilen sonuçlar.....	61
Şekil 6.4 Video1 için geliştirilen kutu tabanlı kod çizelgesi yöntemiyle elde edilen sonuçlar.....	62
Şekil 6.5 Video2 için çerçeve farkı yöntemiyle elde edilen sonuçlar	63
Şekil 6.6 Video2 için Gauss karışımları yöntemiyle elde edilen sonuçlar.....	64
Şekil 6.7 Video2 için kutu tabanlı kod çizelgesi yöntemiyle elde edilen sonuçlar.....	65
Şekil 6.8 Video2 için geliştirilen kutu tabanlı kod çizelgesi yöntemiyle elde edilen sonuçlar.....	66
Şekil 6.9 Video3 için çerçeve farkı yöntemiyle elde edilen sonuçlar	67
Şekil 6.10 Video3 için Gauss karışımları yöntemiyle elde edilen sonuçlar.....	67

Şekil 6.11 Video3 için kutu tabanlı kod çizelgesi yöntemiyle elde edilen sonuçlar.....	68
Şekil 6.12 Video3 için geliştirilen kutu tabanlı kod çizelgesi yöntemiyle elde edilen sonuçlar	68
Şekil 6.13 Elle bölütlenmiş ön plan maskesiyle diğer ön plan maskeleri.....	69

ÇİZELGELER DİZİNİ

Çizelge 2.1 Küp köşegenlerinde elde edilen renk tonları	6
Çizelge 6.1 Arka plan modeli oluşturma aşamasında yöntemlerin videolara göre hızları	58
Çizelge 6.2 Ön plan arka plan ayrımı aşamasında yöntemlerin videolara göre hızları	58

1. GİRİŞ

Ön plan ve arka plan ayrımı; video çerçevelerinin oluşturduğu dizilerde hareketli ön plan nesnelere hareketli arka plan nesnelere ayrılması olarak tanımlanır. Bu işlem hareketli nesne takibi, yüz bulma, yüz tanıma gibi görsel bilgisayar uygulamalarının ilk aşamasını oluşturan temel kısımdır. Ön plan, arka plan ayrımının kullanıldığı birçok alan vardır. Video denetimi ve trafik görüntülemesi gibi video uygulamalarında kullanılır. Ayrıca video konferans görüşmelerinde öncelikle ön plan ve arka plan ayrıştırılarak arka plan herhangi bir görüntüyle değiştirilir. Böylece konferans görüşmelerindeki kişilerin bulunduğu ve görünmesi istenilmeyen arka plan görüntüleri değiştirilerek konuşmacıların bulunduğu ortam gizlenir. Bölütlenen videoda nesne tabanlı video kodlama kullanılarak ön plan sıkıştırılır ve video gönderim hızında etkin bir artış sağlanır (Meier ve Ngan, 1999). Gelişmiş video düzenleme araçları kullanılarak ön plan ve arka plan ayrımı yardımıyla değişik birçok video nesnesi birleştirilerek farklı görüntüler elde edilebilir. Askeri uygulamalarda uyarı sistemini otomatize etmek için kullanılabilir. Örneğin girilmesi yasak bir bölgeden görüntü alan kamerada ön plan ve arka plan ayrımı yaparak hareket eden nesne belirlendiğinde alarm sisteminin devreye girmesi sağlanabilir.

Yukarıda da bahsedildiği üzere birçok görsel bilgisayar uygulamalarının temel ve kritik çalışma alanlarından biri videolarda hareket eden nesnelere tanımlanması ve belirlenmesidir. Arka plan çıkarımı en çok kullanılan genel yaklaşımdır. Bu yaklaşımda arka plan modelinden belirli bir şekilde farklılık gösteren ve hareket eden nesnelere arka plandan ayrılarak ön plan olarak tanımlanırlar. İyi bir arka plan çıkarımı yapan algoritma gerçekleştirebilmek için çözülmesi gereken bazı sorunlar vardır. Bunlar:

1. Öncelikle ani ışık değişikliklerine karşı sağlam olmalıdır.
2. Sallanan yapraklar, yağmur, kar ve hareket eden nesnelere oluşturduğu gölgeler gibi durgun olmayan arka plan nesnelere ön plan nesnelere olarak bulmamalıdır.
3. Trafikteki araçların durup hareket etmesi gibi hareketlilik durumu değişken olan nesnelere belirleyebilecek bir arka plan modeline sahip olmalıdır.
4. Gece gündüz gibi geniş zamanlı ışık değişikliklerine karşı sağlam olmalıdır.

Videolarda ön plan ve arka plan ayrımı (arka plan çıkarımı) uygulamalarında her bir video çerçevesi bir referansla ya da arka plan modeliyle kıyaslanır (Bobick ve Davis 2001, Ching vd. 2005, Deane 2007). Arka plandan kesin bir şekilde farklılık gösteren karşılaştırılan çerçevedeki pikseller hareket eden nesnelere (ön plan) olarak kabul edilir. Ön plan olarak kabul edilen bu pikseller daha sonra nesnenin yerinin belirlenmesinde ve iz sürmede kullanılır. Birçok görsel bilgisayar uygulamasında arka plan çıkarımı ilk basamağı oluşturduğundan önemlidir. Karmaşık arka plana sahip videolarda hareket eden nesnelere belirlenmesi çok zordur ve üzerinde çalışılmaya devam edilmektedir.

İyi bir arka plan çıkarımı algoritmasının doğru olarak çözmesi gereken yukarıda da belirtildiği üzere birçok sorun vardır. Trafik akışının ve yaya geçişlerinin olduğu yol kenarında yayaların bulunduğu bir caddeyi çeken durağan bir kameranın olduğunu ve bu kameradan alınan görüntüde ön plan ve arka plan ayrımı yapıldığını varsayalım. Bu durumda kullanacağımız arka plan çıkarımı algoritmasının gün içindeki ışık değişimlerinden etkilenmemesi gerekir. Yol kenarındaki ağaçların yapraklarının rüzgârdan sallanmasından etkilenerek yapraklar ön plan olarak belirlenmemelidir. Hareket eden araçların ve yayaların gölgeleri de ayrı bir ön plan nesnesi olarak bulunmamalıdır. Kırmızı ışıkta durup yeşil ışıkta hareket eden araçları belirlerken ilk durumda arka plan ikinci durumda ön plan olarak belirlenebilmelidir. Gerçek zamanlı uygulamalarda kullanılan arka plan çıkarımı algoritması az bellek gerektirmeli ve hızlı sonuç üretebilmelidir.

Bu çalışmada öncelikle videoyu oluşturan temel bileşenlerden kısaca bahsedilecektir. Daha sonra ön plan ve arka plan ayrımında en çok kullanılan yöntemler kısaca anlatılacaktır. Bu yöntemlerden sonra kod çizelgesi yöntemini oluşturan algoritma adımları, kutu tabanlı kod çizelgesini oluşturan algoritma adımları ve geliştirilen kutu tabanlı kod çizelgesi yöntemini oluşturan algoritma adımları verilecektir. Tez çalışması kapsamında geliştirilen yazılımdan bahsedilecektir. Son olarak farklı yöntemlerin farklı video çerçevelerine uygulanmasıyla elde edilen sonuçlar ve süreler karşılaştırılacaktır.

2. VİDEOYU OLUŞTURAN BİLEŞENLER

Aşağıdaki bölümde videoyu oluşturan temel bileşenler parçadan bütüne doğru sırasıyla anlatılacaktır.

2.1 Piksel

Tüm sayısal görüntülerin temel yapı taşı olan noktacıklara "piksel" denir. İngilizcede **Picture Element** (Resim elemanı) kelimelerinin ilk hecelerinden oluşan ve bir grafik veya resimdeki tek bir nokta veya karedir. Grafik monitörler; resim, grafik veya fotoğrafları binlerce veya milyonlarca piksele bölerek gösterirler. Bunlar satır ve sütun olarak dizilmişlerdir. Pikseller birbirlerine o kadar yakındır ki göze bitişik gibi görünürler.

Sayısal görüntüler, yatay ve dikey biçimde yan yana sıralanmış piksellerden oluşur. Sayısal görüntü, görüntü eninde ve boyunda bulunan piksel sayısı ile tanımlanır. Pikselin kendi başına en ve boy değerleri yoktur. Yani dikdörtgen biçimindeki tek bir piksel 1*1 mm, 1*1 cm hatta 3*2 m bile olabilir. Piksel sayısı arttıkça görüntü kalitesi de artmaktadır.

2.2 Çerçeve

Video dizilerindeki çerçevenin karşılığı sayısal görüntülerdir. Diğer bir ifadeyle sayısal görüntülerin zamana yayılması sonucunda videolar oluşur. Video konusunda daha geniş açıklamaya ilerleyen bölümlerde yer verilecektir. En basit ifadesiyle sayısal görüntüler piksellerin birleşmesi sonucu elde edilir.

Görüntü, iki boyutlu $f(x,y)$ fonksiyonu olarak tanımlanabilir ve herhangi bir (x,y) koordinatı için f fonksiyonunun genliği, o noktada, görüntünü yoğunluğu (intensity ya da gray level) olarak adlandırılır. Bütün x , y ve fonksiyon genlikleri sonlu ve tamsayı ise bu görüntü, sayısal görüntü olarak adlandırılır.

2.2.1 Çözünürlük

Çözünürlük kavramının birbirinden farklı birçok tanımı bulunmaktadır. En basit ve yalın ifadeyle birim uzunluktaki (inç, cm, ...vb) piksel sayısıdır. Diğer bir genel tanım ise; görüntüdeki piksellerin birbirine olan uzaklıklarının ölçüsüne çözünürlük denir, şeklindedir.

Aynı fiziksel boyuta sahip görüntüler aynı çözünürlükte olmayabilirler. Yukarıda da belirtildiği üzere çözünürlüğü belirleyen birim uzunluktaki piksel sayısıdır. Şekil 2.1-2.4'deki görüntüleri buna örnek olarak verebiliriz (Ertürk 2009).



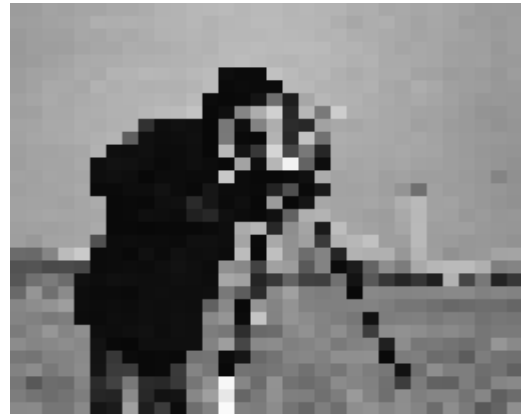
Şekil 2.1 256x256 piksel



Şekil 2.2 128x128 piksel



Şekil 2.3 64x64 piksel



Şekil 2.4 32x32 piksel

2.2.2 Renk kanalları

Sayısal görüntünü piksellerden oluştuğunu ve piksellerinde renklerden oluştuğunu söylemiştik. Renk kanalları bir pikselin içerebileceği farklı renk bilgilerini gösteren yapıdır.

Renk kanalları sayısal bir görüntüdeki ana renk seviyelerinin ayrı ayrı tutulduğu kanallardır. Bu renk kanallarının birleşmesi sonucu renkli görüntüler elde edilir. Renk kanalları her bir renk seviyesinin tek başına sağlayamadığı fonksiyonelliği sağlayabilir; böylece renk kanalları görüntüde tonlama yapabilmek için kullanılabilecek çok faydalı bir araçtır (Mirza vd. 2008).

2.2.3 Renk uzayları

Renk uzayları, renkleri tanımlamak için kullanılan matematiksel modellerdir. Renk uzayları, bütün renkleri temsil edecek şekilde oluşturulmaktadır. Renkmetri biliminin temelini oluşturan Grassmann'ın birinci kuralına göre, bir rengi belirlemek için birbirinden bağımsız 3 değişkene ihtiyaç vardır. Bu yüzden renk uzayları 3 boyutlu olarak tasarlanmaktadır. En popüler üç renk modeli bilgisayar grafiklerinde kullanılan RGB, video sistemlerinde kullanılan YCbCr ve renkli dökümlerde kullanılan CMYK'dır. Tüm renk uzayları, kamera ve tarayıcı gibi aygıt kaynaklı RGB bilgisi kullanılarak elde edilmektedir (Yılmaz 2002).

2.2.3.1 Gri ölçekli görüntüler

Gri ölçekli görüntüler, siyah beyaz görüntüler olarak bilinirler. Bu görüntülerde her bir piksel tek bir değerden oluşur ve bu değer o pikselin ışık açısından yoğunluğunu (açıklık koyuluk değerini) gösterir. Diğer bir ifadeyle gri ölçekli görüntüler tek bir renk kanalı içerir. Siyah-beyaz görüntülerde piksel değerleri, en düşük yoğunluk siyahı, en yüksek yoğunluk beyazı gösterecek şekilde grinin tonlarından oluşmaktadır. Gri ölçekli görüntüler 8 bit derinliğe sahiptir ve pikselin değeri 0 ile 255 arasında değişir ve 0 siyahı 255 ise beyazı ifade eder.

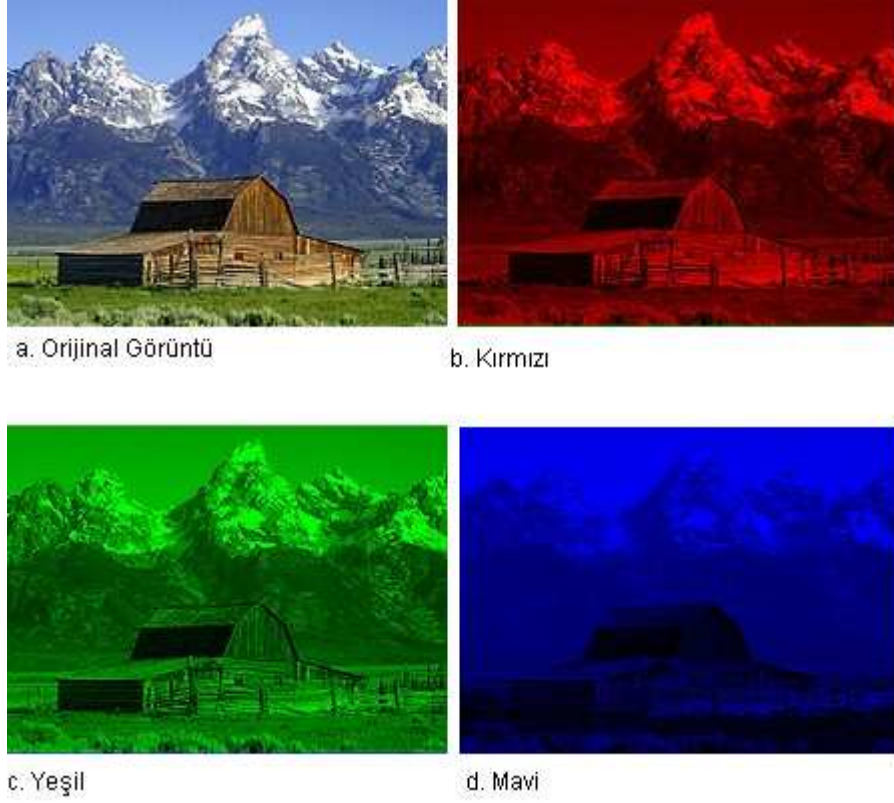
2.2.3.2 RGB renk uzayı

RGB renk uzayı bilgisayar grafikleri üzerinde yoğun bir kullanım alanına sahiptir. Kırmızı(R), yeşil(G) ve mavi(B) birbirine karıştırılabilen üç temel renktir. Diğer renkler ise, bu üç renk karıştırılarak elde edilebilmektedir. Çizelge 2.1’de bu seviyelere karşılık ortaya çıkan renkler gösterilmektedir.

Çizelge 2.1 Küp köşegenlerinde elde edilen renk tonları

	Aralık	Beyaz	Sarı	Camgöbeği	Yeşil	Eflatun	Kırmızı	Mavi	Siyah
R	0-255	255	255	0	0	255	255	0	0
G	0-255	255	255	255	255	0	0	0	0
B	0-255	255	0	255	0	255	0	255	0

RGB renk uzayındaki bir resmi işlemek etkili bir yol değildir. Örneğin verilen bir pikselin renk yoğunluğu değiştirilmek istendiğinde, çerçeve tamponundan üç RGB değeri de okunmak zorundadır. Ardından renk yoğunluğu hesaplanmakta ve istenilen değişiklik yapıldıktan sonra yeni RGB değeri elde edilerek çerçeve tampona geri yazılmaktadır. Farklı bir renk uzayı kullanılarak sistemin, saklanan resmin yoğunluk ve renk biçimine doğrudan erişim hakkı olduğu takdirde bazı işlem basamakları daha kolay ve hızlı yapılabilir. Bu ve bunun gibi birkaç nedenden dolayı birçok video standardı parlaklık ve iki farklı renk sinyali kullanmaktadır. Bunlardan en genel olanları YUV, YIQ, YCbCr ve HSV renk uzaylarıdır. Bu renk uzaylarının birçok benzerliği olmasına rağmen bir takım farklılıkları da bulunmaktadır.

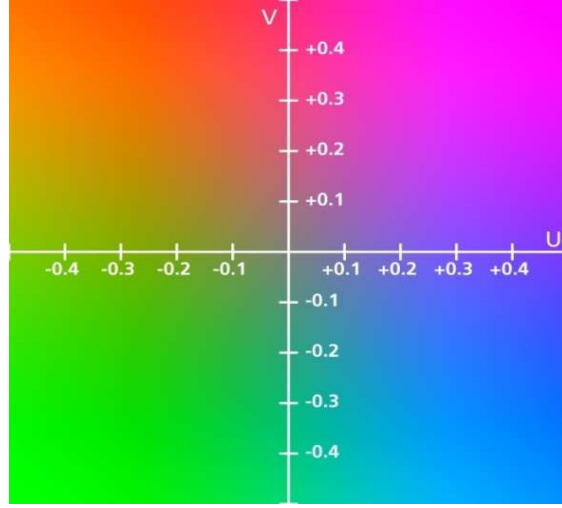


Şekil 2.5 RGB renk uzayında görüntünün ayrı ayrı görünümü

Şekil 2.5’te orijinal görüntü ve bu görüntünün R, G, B bileşenlerinin görüntüleri verilmiştir. Kar güçlü kırmızı, mavi ve yeşilden oluştuğu için her üç bileşende parlak bir şekilde görülmektedir. Kahverengi ahır, güçlü kırmızı ve yeşil, zayıf maviden oluştuğu için R, G bileşenlerinde parlak B bileşeninde parlak değildir (Ertürk 2009).

2.2.3.3 YUV renk uzayı

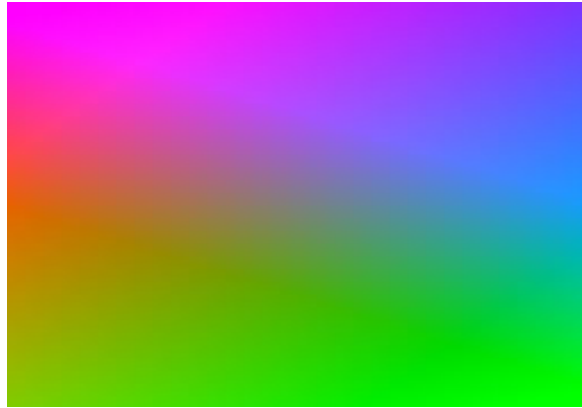
YUV renk uzayı PAL, NTSC ve SECAM bileşik ve renkli video standartlarında kullanılmaktadır. Siyah-beyaz sistem parlaklık bilgisini Y bileşeni ile tutmaktadır. U ve V Renk bilgisi bir siyah-beyaz televizyon alıcısının görüntüyü hala siyah-beyaz gösterebileceği şekilde sinyale dâhil edilmektedir. Bu renk uzayının geliştirilmesindeki temel amaç siyah-beyaz alıcıların, renkli video sinyalini siyah-beyaz şeklinde gösterebilmeleridir. Renkli televizyon alıcıları ise ilave renk bilgisini görüntüyü renklendirmek için kullanmaktadırlar. Şekil 2.6 U ve V renk bileşenlerini göstermektedir.



Şekil 2.6 UV renk bileşenleri

2.2.3.4 YIQ renk uzayı

YIQ renk uzayı, YUV renk uzayından türetilmiş ve NTSC bileşik video standardında seçimlilik olarak kullanılan bir renk uzayıdır. Burada Y parlaklık değerini göstermekte, I ve Q değerleri ise renk bilgisini transfer etmek için kullanılmaktadır. Şekil 2.7 I ve Q renk bileşenlerini göstermektedir.



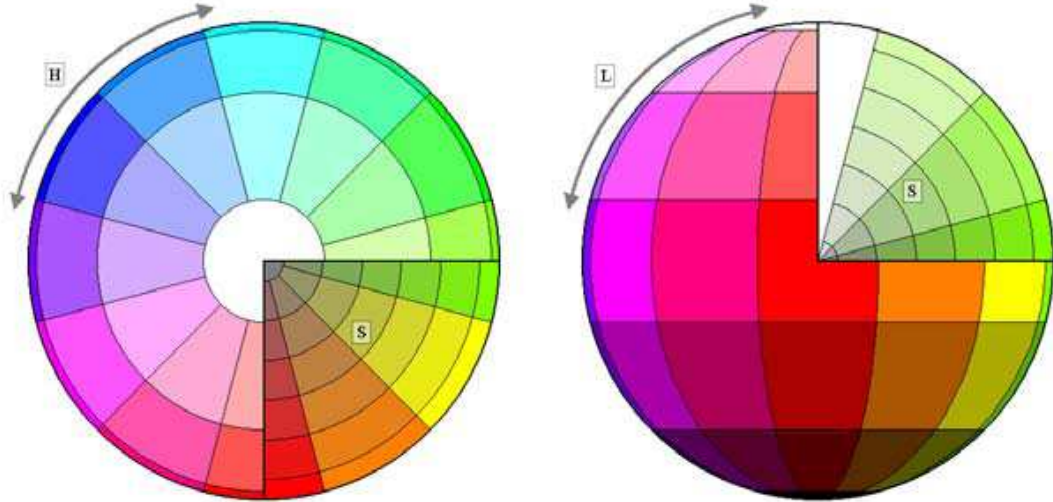
Şekil 2.7 I ve Q renk bileşenleri

2.2.3.5 YCbCr renk uzayı

YCbCr renk uzayı, dünya çapında sayısal video standardı oluşturma çabaları sırasında ortaya çıkmıştır. YCbCr, YUV renk uzayının boyutlandırılmış ve kaydırılmış bir şeklidir. Y , 8 bitlik 16-235 aralığında tanımlanmaktadır. Cb ve Cr ise de 16-240 arasında tanımlanmaktadır. 4:4:4, 4:2:2, 4:1:1, 4:2:0 gibi birçok YCbCr örnekleme biçimi bulunmaktadır (Jack 1995).

2.2.3.6 HSI, HLS ve HSV renk uzayları

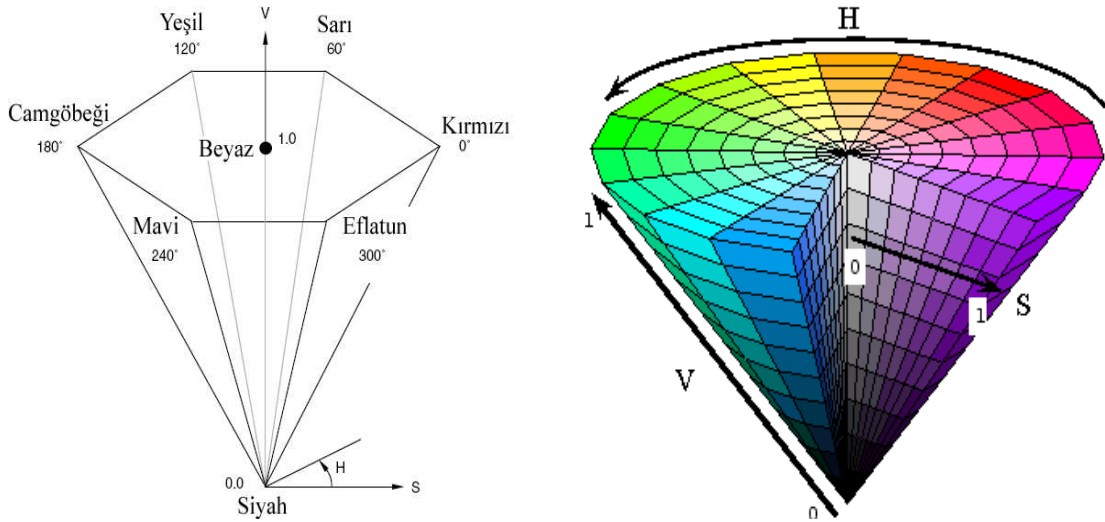
HSI ve HSV renk uzayları, insanın sezgisel olarak ve daha kolay renk seçimi yapabilmesi amacıyla geliştirilmişlerdir (Jack 1995). Renklerin el ile gösterilmeleri gerektiğinde ve kullanıcıların renkleri görerek seçmeleri gerektiği durumlarda idealdirler (Jack 1995). HLS ve HSI birbirine çok benzerdirler. Parlaklık bileşeni L yerine yoğunluk bileşeni I kullanılmıştır. HSI ile HSV arasındaki fark ise parlaklık bileşeninin hesaplanma şekli, I yada V renk doyumunun dağılım ve dinamik aralığın değişmesiyle gerçekleşmektedir. Şekil 2.8 HLS renk uzayını göstermektedir.



Şekil 2.8 HLS renk uzayı

HSI renk uzayı, R , G , B değerlerine eşit oranda bağlı olan parlaklık değerleri ile doğrudan alakalı katsayı, eşitleme, histogram gibi geleneksel resim işleme metotları için en iyi renk uzayıdır. HSV renk uzayı ise renk doyumu açısından büyük bir dinamik aralığa sahip olduğu için, renkleri değiştirme ya da renk yoğunluğu ayarlama kullanılmaktadır.

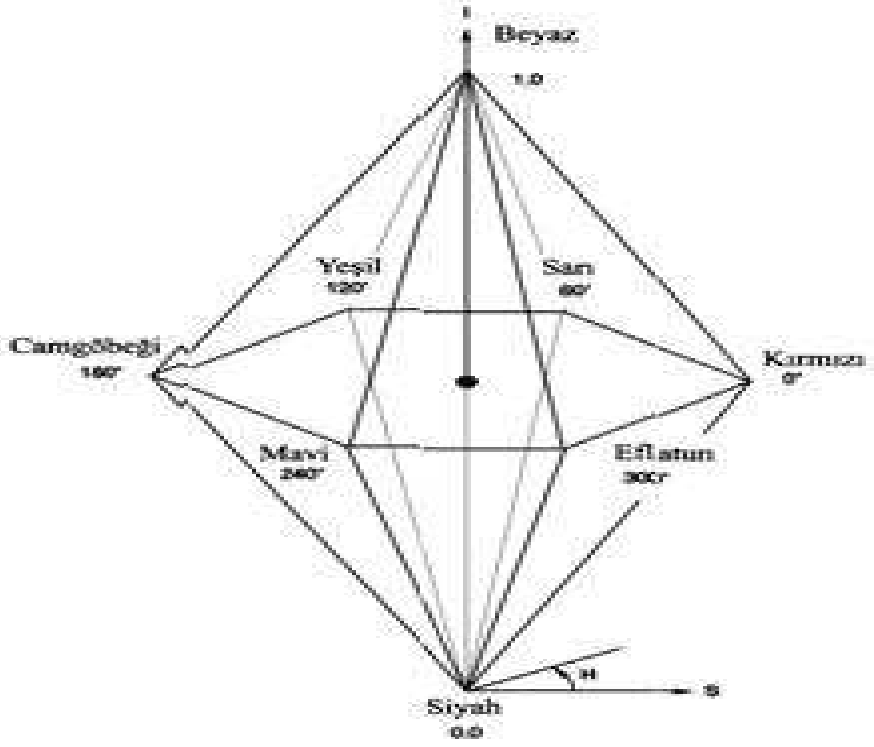
Şekil 2.9 altıgen huni şeklinde HSV renk uzayını göstermektedir. Altıgen huninin en alt kısmı siyah ve $V=0$ 'dır, tepesi ise $V=1$ değerine karşılık gelmektedir. Bu konumda en yoğun renkler elde edilmektedir. Kırmızı 0° olmak üzere tamamlayıcı renkler H 'ye göre 180° karşılıklıdır. S ile gösterilen ve 0 ile 1 arasındaki değere *oran* denilmektedir. $S=0$ iken ve $V=1$ beyaz rengi ve V 'nin diğer değerleri ise grileri göstermektedir. $S=0$ iken H değerinin önemsiz olduğu görülmektedir. $V=1$ ve $S=1$ olduğunda da katıksız renkler görülmektedir. V değerini değiştirmeden S 'i azaltarak renge beyaz ekleyebilmek, S 'i değiştirmeden V 'yi azaltarak renge siyah eklemek mümkündür. Tonlar ise hem S , hem de V 'yi azaltarak elde edilebilmektedir (Jack,1995).



Şekil 2.9 Altıgen huni şeklinde HSV renk uzayı

Şekil 2.10 çift altıgen huni şeklinde HSI renk modelini göstermektedir. Altıgen huninin tepesi $I=1$ 'e karşılık gelmektedir ve beyazı göstermektedir. Altıgen huninin tabanı $I=0$ ile gösterilmektedir ve siyah renge karşılık gelmektedir. Birbirini tamamlayan renkler

180° karşılıklıdır ve H ile ifade edilmektedirler. Dikey eksen üzerindeki S değerleri 0 ile 1 arasındadır. Gri seviyeler $S=0$ iken elde edilmektedir. Tonların en yüksek değeri $S=1$ ve $I=0.5$ iken elde edilmektedir.



Şekil 2.10 çift altıgen huni şeklinde HSI renk modeli

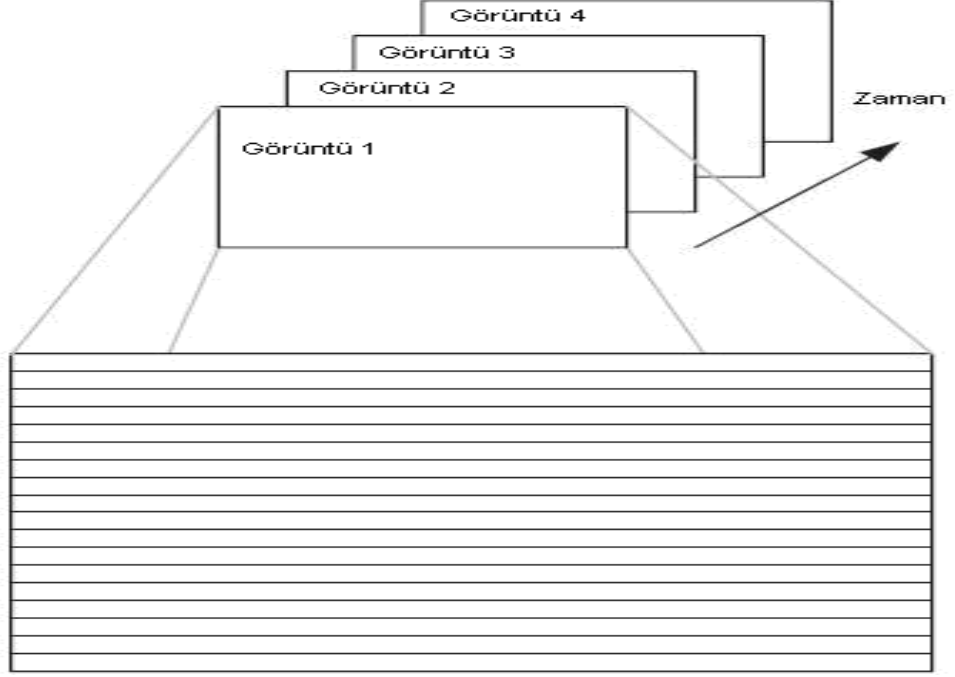
3. VIDEO

Birçok çeşidi ve uygulama şekli olmasına rağmen video, temelde görsel bir bilginin bir noktadan diğer bir noktaya transfer edilmesinde kullanılan sinyale verilen isimdir (Jack, 1995). Genellikle kamera tarafından görsel bilgiler video sinyaline dönüştürülmekte, dağıtım kanalları yoluyla alıcıya kadar iletilmekte ve alıcı tarafından tekrar görsel hale dönüştürülmektedir. Kelime köken olarak Latince Videre (görmek)'den gelmektedir ve video *görüyorum* demektir. Video teknolojisi ilk olarak televizyon yayınlarında kullanılmıştır. Gelişen teknolojiyle birlikte birçok kişisel kullanım alanı bulmuştur.

Başlangıçta video, sadece renksiz (siyah, beyaz ve gri tonlar) analog bilgi içermektedir. Renkli yayınlar geliştirildiğinde renkler, analog RGB verisi şeklinde ifade edilmiştir. Bununla birlikte bu teknik siyah-beyaz çözümün 3 katı bant genişliği gerektirdiğinden renk bilgisini YIQ ya da YUV şeklinde ifade eden alternatif yöntemler geliştirilmiştir (Cücioğlu 2003). YIQ ya da YUV analog bilgiyi üç farklı sinyal yerine orijinal siyah-beyaz video sinyali ile aynı bant genişliğinde tek bir sinyal olarak ileten bir yöntemdir. Bu bileşik sinyal günümüzde hala kullanılan NTSC, PAL ve SECAM video standartlarının (analog renkli televizyon sistemleri) temelini oluşturmaktadır (Cücioğlu 2003). Bugün videoyu ifade etmenin birçok yolu bulunmaktadır ve bunlar matematiksel olarak hala RGB'ye bağlıdır.

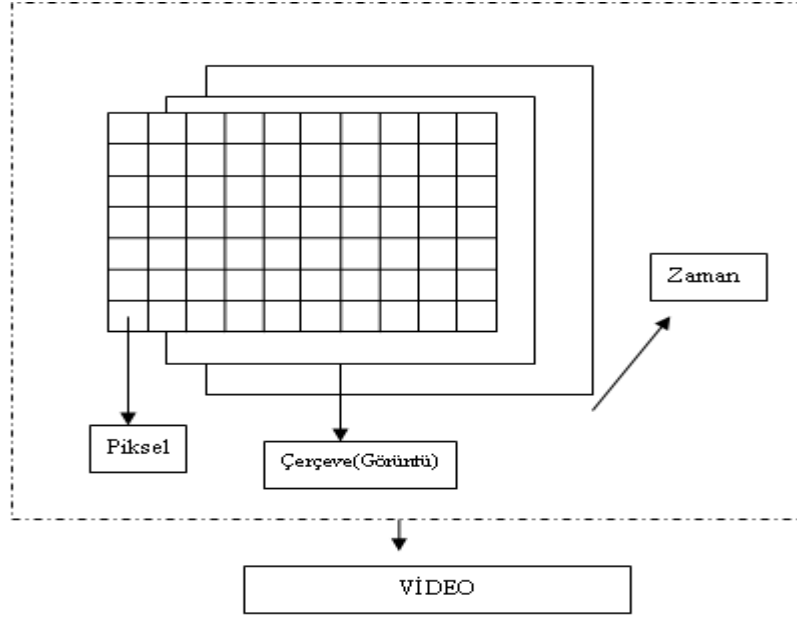
Analog video sinyali, görüntüleri elektrik akımı gibi sürekli değişen sinyallerden oluştururken, sayısal videoda sadece 0 ve 1'lerden oluşan bit akımı vardır. Analog sinyaller iletim ve depolama esnasında oluşan gürültü ve parazitlere karşı çok duyarlıdır. Bu yüzden orijinal videodan alınan kopyaların görüntü kalitesi düşmektedir. Analog sinyaller sıkıştırılamayacağından (satır atlamalı tarama bir tür kayıplı analog sıkıştırma; ancak sınırlı bir sıkıştırma sağlar), artan yayın sayısına sınırlama getirmektedir. Günümüzde *sayısal video* adında hızlı bir dönüşüm başlamıştır ve analog yayınların yerini almaktadır. Sayısal video gürültü ve parazitlerden daha az etkilenmekte ve görüntü kalitesi hiç bozulmadan kopyalanabilmektedir. Sıkıştırma işlemine tabi tutulabilen sayısal video sayesinde aynı bant genişliğinden daha fazla yayın iletmek mümkün olmaktadır.

Video, devamlı hareket eden bir görüntü olarak görülmesine rağmen, şekil 3.1’de gösterildiği gibi hareketli gözükecek kadar hızlı değişen bir grup hareketsiz görüntüden oluşmaktadır (Jack 2002).



Şekil 3.1 Video, bir grup hareketsiz görüntü

Videolarda belli bir zaman aralığında ardı ardına gösterilen görüntülere çerçeve denir.



Şekil 3.2 Video oluşturan bileşenler

Şekil 3.2’de videoyu oluşturan bileşenler ve videonun nasıl oluştuğu gösterilmiştir. Şekil 3.2’ye göre piksellerin birleşmesiyle hareketsiz görüntüler oluşur, bu hareketsiz görüntülerin zamanla değişmesiyle hareketli görünen videolar oluşur.

4. VİDEOLARDA ÖN PLAN VE ARKA PLAN AYRIMI

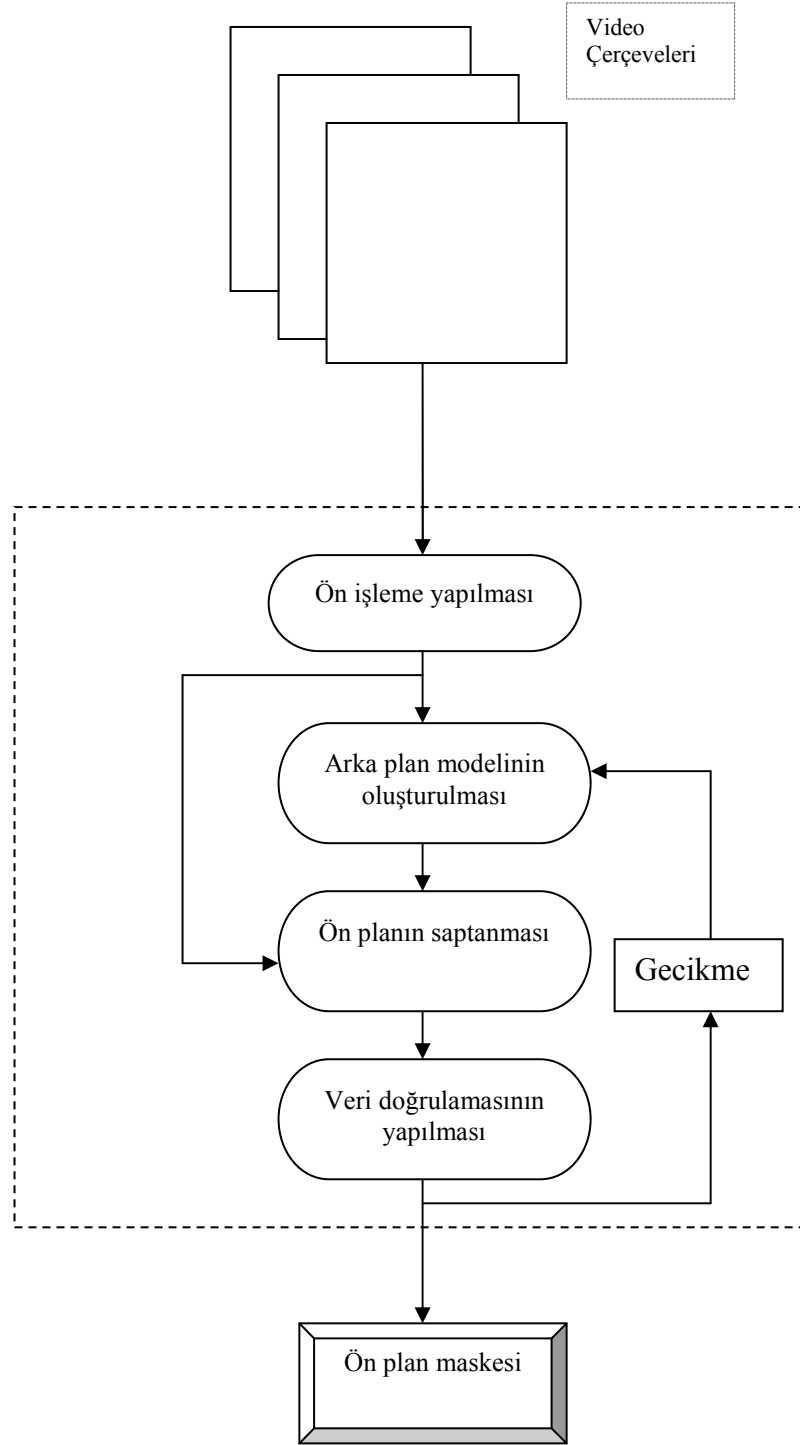
Modern video sistemlerinde sayısal video analiz teknikleri önemli bir rol oynamaktadır. Video analizindeki önemli aşamalardan biri videolardaki hareketli nesnelerin tespit edilmesidir (Zaibi 1999). Videolarda hareketli nesnelerin bulunması için videolar ön plan arka plan ayrımı gerçekleştirilir. Hareketli nesnelerin bulunması savunma ve güvenlik sistemlerinden olduğu gibi trafik kontrolü, otopark kontrolü gibi uygulamalarda da kullanılmaktadır. Yoldaki araçların hareketleri ve bu hareketlerin ürettiği alarmlar; sollama, yakın takip, kaza, aşırı sürat ... v.b analiz için ön plan arka plan ayrımı kullanılmaktadır.

Hareket halindeki bir nesnenin takip edilmesi kontrol uygulamaları açısından önemlidir. Örneğin seyir halindeki araçların otomatik yönlendirilmesi, hareketli nesnenin takibi ve gözetleme gibi uygulamalar hareket parametrelerinin (hareket yönü, hareket hızı gibi) bulunmasına ihtiyaç duyar. Hareket nesnelere doğru ulaşabilmek ise görüntüdeki nesnenin şekil ve konum bilgilerinin minimum hatayla tespit edilmesini gerektirir.

Daha öncede belirtildiği gibi videolarda ön ve arka plan ayrımı görsel bilgisayar uygulamalarının (bilgisayarlı görü uygulamaları) ilk basamağını oluşturur. Bu yüzden gerçekleştirilen uygulamaya göre değişen yöntemler geliştirilmiştir. İlerleyen bölümlerde bu yöntemlerin genel işleyişinden ve ön plan arka plan ayrımında kullanılan temel yöntemlerden bahsedilecektir.

4.1 Ön Plan - Arka Plan Ayrımında Kullanılan Algoritmalar

Literatürde çok sayıda arka plan çıkarımı algoritması olmasına rağmen hemen hemen hepsi şekil 4.1'deki gibi bir işleyişe sahiptir.



Şekil 4.1 Ön plan, arka plan ayrıştırılmasında kullanılan algoritmaların genel gösterimi

Arka plan çıkarımı algoritmalarındaki dört temel adım; ön işleme (preprocessing), arka plan modelinin oluşturulması (background modelling), ön planın ortaya çıkarılması (foreground detection) ve veri doğrulamasının (data validation) yapılmasından oluşur

(Wren vd. 1997, Elgammal vd. 2000, Kaewtrakulpong ve Bowden 2001, Ching vd. 2005). Önışleme aşamasında gelen video verisinin bir dizi işlemden geçirilerek sonraki alt adımlarda kullanılabilir hale getirilmesi işlemleri gerçekleştirilir. Yani ham video verisini belirli bir formata sokulur. Arka plan modellemesi aşamasında formatlanmış video verileri kullanılarak arka plan hesaplanır, arka plan modeli oluşturulur ve güncellenir. Bu arka plan modellemesi tüm arka plan sahnelerinin istatistiksel tanımını yapar. Ön planın saptanması aşamasında ise video çerçevelerinde arka plan modeliyle tam olarak açıklanamayan pikseller bulunur ve ön plan maskesini oluşturacak bu pikseller ikili (x, y koordinatlarına göre) bileşenler halinde üretilir. Son olarak veri doğrulaması aşamasında ise; önceki aşamalarda oluşturulan ön plan maskesi ele alınır. Bu maskede hareket eden gerçek nesnelere uyuşmayan pikseller ayrıştırılır ve ön plan ayrımında kullanılacak maske elde edilir.

Gerçekleştirilecek uygulamaya göre her bir aşamada çok çeşitli yöntemler kullanılmaktadır. Çalışmanın ilerleyen bölümlerinde her bir aşamadan ayrı ayrı bahsedilecek ve bu aşamalarda en çok kullanılan yöntemler anlatılacaktır.

4.1.1 Önışleme

Çoğu bilgisayar görüntüleme sistemlerinde, video çerçeveleri işlenmeye başlamadan önce kamera gürültüsünü azaltmak için basit zamansal ve/veya uzaysal yumuşatma kullanılır. Yumuşatma işlemi, geçici çevresel gürültüleri azaltmak veya yok etmek için de kullanılır. Yumuşatma yapabilmek için süzgeçlerden yararlanır ve ortanca süzgeç yüksek geçirgen süzgeç gibi süzgeçlerin birçoğu bu işi görür (Shireen vd. 2007).

Hareketli bir kameradan veya değişik noktalardan çekim yapan birden fazla kameradan görüntü alınıyorsa, arka plan modellemesi yapılmadan önce görüntüdeki birbirini takip eden video çerçevelerinin birleştirilmesi gerekir (Mittal ve Huttenlocher 2000, Kang vd. 2003).

Gerçek zamanlı video uygulaması yapılıyorsa, algoritmanın kısa zamanda cevap verebilmesi için ön işleme aşamasında çerçeve sıklığının ve çerçeve boyunun azaltılması en çok kullanılan yöntemlerdendir (Shireen vd. 2007).

Uygulamayı hızlandırmak için ve video çerçevelerinin sadece ilgili kısmında işlem yapmak için maskeleye kullanılabilir. Örneğin sadece yol ile ilgileniyorsak çevredeki binaları maskeleyerek görüntüden çıkarılabiliriz veya arka plan modeli oluştururken zaman kaybı olmaması için binaları, dikkate alınmaması için işaretleyebiliriz.

Önişlem aşamasında yapılması gereken önemli işlemlerden biri de verinin (video çerçevelerinin) arka plan çıkarımı algoritmasının kullanabileceği hale getirilmesi işlemidir. Algoritmaların çoğu her bir piksel için yönsüz gerçel bir sayı olan parlaklık şiddetini kullanır (Wren vd. 1997). Bununla birlikte renkli görüntüler ve RGB (Red-Green-Blue) (Kırmızı-Yeşil-Mavi renk bileşenleri), HSV (Hue-Saturation-Value) (Renk Tonu-Rengin grilikten uzaklaşma ölçüsü-Parlaklık) gibi renk uzayları arka plan çıkarımı algoritmalarında gittikçe artan bir öneme sahip olmaya başlamıştır (Wren vd. 1997, Cuchiara vd. 2003). Çünkü renkli görüntüler video üzerindeki düşük parlaklıktaki alanlarda daha fazla bilgi içerir. Renkli görüntü kullanıldığında daha fazla bilgi elde edilmesine karşın işlem yükünü artırır. Bu yük sadece parlaklık değerinin kullanıldığı uygulamalardaki işlem yükünün dört katı kadardır.

4.1.2 Arka plan modellemesi

Otomatik gözetleme sistemleri konusunda geçen yıllarda pek çok araştırma çalışmaları başlatılmıştır (Haritaoğlu vd. 2000, Bobick ve Davis 2001, Collins vd. 2001, Javed ve Shah 2002). IBM ve Microsoft gibi yazılım şirketleri de hareket algılamaya yönelik çalışmalar yürütmektedir (Haritaoğlu vd. 2000, Perez vd. 2002).

Gözetleme amaçlı kullanılan sistemlerin çoğu sabit bir kameranın belirli alanın görüntüsünü almasından oluşur. Her ne kadar bir kamerayı gözetlenecek alana monte etmek ucuz ve kolay bir iş olsa da bu kameradan alınan görüntüyü işleyecek insan gücü

oldukça maliyetli olmaktadır (Ekinci ve Gedikli 2005). Günümüzde kullanılan gözetleme sistemleri, sabit kamera ve bu görüntüyü kaydeden bir kayıt cihazından oluşmaktadır. Bu sistemden elde edilen veri ancak olay sonrası inceleme amaçlı kullanılmaktadır. Gözetleme Otomasyonlarından elde edilmek istenilen görüntünün sadece kaydedilmesi değil, görüntüden gerçek zamanlı içeriğin elde edilebilmesidir. Bu sistemlerden istenen en temel veri; alana giren, hareket eden nesnelerin tespitidir. Hareket eden nesnelerin tespitinin en kolay yolu, var olan kare ile bir önceki karenin karşılaştırılmasıdır. Bu karşılaştırma sadece iki kare arasındaki farkı bulmayı sağlar. Hareket eden nesnelerin tutarlı ve tam şekilde bulunması işlemi ise ancak arka planın oluşturulması, bu modelin ani ışık değişimi, arka planda oluşan değişiklikler gibi etmenlere karşı modelin güncellenebilmesiyle mümkündür. Arka plan modelleme konusunda uygulanabilen en temel varsayım, hareket etmeyen cisimler haricinde kalan görüntünün istatistiksel verileriyle modellenebilecek düzgün bir davranış sergilemesidir. Eğer bunu sağlayan bir model oluşturulursa, modele uymayan parçalar; sahne alanına giren, hareket eden nesnelere belirtir. Bu işlem **Arka Plan Modelleme** ya da **Arka Plan** olarak adlandırılmaktadır.

Arka plan modelleme için kullanılan yöntemlerin en büyük bölümünü tümevarım (videoyu oluşturan temel bileşenlerden piksellerin kullanılarak arka plan modelinin oluşturulması yani parçadan bütün elde edilmesi) yöntemler oluşturur. Bu yöntemler, görüntüyü oluşturan her bir karenin piksel değerlerinin dağılımını kullanarak arka plan modelini oluştururlar. Arka planın gerçekçi bir şekilde modellenmesi, eğitim için 10 – 30 saniye süresince video karelerinin istatistiksel değerlerin yorumlanması ile olur (Ekinci ve Gedikli 2005). Eğitim süreci içeren tekniklerde, arka planın gerçeğine en yakın şekilde modellenebilmesi için, eğitim süresince hareketli nesnelerin mümkün olduğunca az, hatta hiç olmaması gerekmektedir. Fakat gerçek durumlarda bu çoğu zaman mümkün değildir. Kalabalık alışveriş merkezlerinden, sürekli trafik akısı olan yollara kadar çok geniş bir aralıktaki sahnelerde arka plan modelleme çalışması yapılması gerekebilir. Bu yüzden modelin öğrenim süresinde bu tarz durumlara karşı çözüm getirmesi gerekmektedir.

4.1.2.1 Özyinelemesiz yöntemler

Genel olarak özyinelemesiz yöntemler kayan pencere yöntemini kullanırlar. Bu yöntem önceki N tane video çerçevesini bir arabellekte depolar ve arabellekte bulunan piksel değerlerine bakarak arka plan tahmini yapar (McIvar 2000, Ching vd. 2005, Deane 2007). Özyinelemesiz yöntemler arabellek dışındaki piksel değerlerine bağımlı olmadığı için kolaylıkla yeni arka plan değerlerine uyarlanabilirler. Fakat bu yöntemlerin bellek tüketimi fazladır. En çok kullanılan özyinelemesiz yöntemler şunlardır:

4.1.2.1.1 Çerçeve farkı

En basit arka plan modelleme yöntemidir. Çerçeve farkı yöntemi t-1 zamanındaki çerçeveyi t zamanındaki çerçeve için arka plan modeli olarak kullanır. Bu yöntemde t zamanındaki çerçeve t-1 zamanındaki çerçeveden çıkarılır ve piksel değerindeki fark eşik değerinden büyükse bu piksel arka plana aittir denir.

$$|\text{Çerçeve}_t - \text{Çerçeve}_{t-1}| > T \quad (4.1)$$

Çerçeve farkı yöntemi (4.1)'de gösterilen formülü kullanarak ön plan arka plan ayırımı gerçekleştirir. (4.1)'de; Çerçeve_t t zamanındaki çerçeveyi, Çerçeve_{t-1} t-1 zamanındaki çerçeveyi ve T'de çerçeve farklarına uygulanan eşik değerini ifade etmektedir.

Bu yöntemin uygulanması oldukça basittir. Bununla birlikte birçok sorun içermektedir (Benton 2009). Bu yöntemin en büyük kusurlarından biri düzgün bir şekilde dağılmış yoğunluk değerine sahip ve yavaş hareket eden nesnelere ait içteki pikseller arka plan olarak algılanır (açıklık problemi). Diğer bir problem ise hareket eden nesnelere arkasında kalan piksellerde ön plan nesnelere algılanır (hayalet etkisi). Bu yöntemle ilgili diğer bir problem ise ön plan nesnelere belirlenebilmesi için nesnelere durmadan hareket etmesi gerekir. Eğer bir nesne çerçeve periyodundan (1/fps) daha uzun süre durgun kalırsa nesne arka plan olarak algılanır.

Bu yöntemin iki temel avantajı vardır (Ching vd. 2005, Benton 2009). Bunlardan biri işlem yükünün az olmasıdır. Diğer bir avantajı ise değişen piksel değerlerine göre kolaylıkla uyarlanabilir olmasıdır. Arka plan sadece bir önceki çerçeveye bağlı olduğu için, arka plandaki değişikliklere diğer metotlara göre daha kolay ve daha çabuk adapte olur. Ayrıca çevresel faktörlerden kaynaklanan gürültüyü diğer karmaşık metotlara nazaran daha iyi bir şekilde görüntüden çıkarır.

Bu yöntem uygulanırken üzerinde önemle durulması gereken noktalardan biri eşik değerinin seçilmesidir. Eşik değeri seçilirken deneme yanılma yöntemi kullanılabilir. İlk önce rastgele bir eşik değeri verilir ve çıkan sonuca göre, ön plan nesnelere arka plan nesnesi olarak bulundursa eşik değeri azaltılarak, arka plan nesnelere ön plan nesnelere olarak bulundursa eşik değeri artırılarak en iyi sonuç elde edilmeye çalışılır. Deneme yanılma yöntemi haricinde eşik değeri adaptif bir şekilde elde edilebilir. Görüntünün ortalama parlaklığı bulunur. Bu parlaklık eşik değeri olarak seçilebilir.

4.1.2.1.2 Ortanca süzgeç yöntemi

Ortanca süzgeç yöntemi en çok kullanılan yöntemlerden biridir. Bu yöntem arabellekteki çerçevelerin her birindeki piksellerin ortancasını bulur ve bu değeri arka plan olarak kabul eder (Cutler ve Davis 1998, Cuchiara vd. 2003). Arka plan modeli oluşturulurken arka planın arabellekteki çerçevelerin yarısından fazla görüldüğü varsayılır. Yani arka plan görüntüde daha fazla görülür, ön plan nesnelere ise değişken sürede ekranda kalır. Ortanca süzgeç yöntemi renkli video çerçeveleri üzerinde çalışırken doğru sonuç üretmez ve sorun oluşturur. Bu sorunu çözmek için medoid yöntemi geliştirilmiştir (Cuchiara vd. 2003). Renkli videolarda her rengin görüntü üzerindeki dağılımı farklı olabilmektedir. Bu durumda ortanca değer hesaplanamaz. Bunun yerine ortacı değer (medoid) hesaplanır. Medoid hesaplanırken çerçevelerdeki renkler ayrı bir küme olarak düşünülür ve kümelerin merkezleri hesaplanır. Arabelleğe yeni katılan her çerçeveden sonra kümelerin merkezleri yeniden hesaplanır. İşlem yükü ortanca süzgece göre fazladır. Arabelleğin boyu arttıkça işlem karmaşıklığı üstel olarak artar. Bu yöntemin algoritma karmaşıklığı her bir piksel için $O(L \log L)$ 'dir (L arabellekteki çerçeve sayısı). Ayrıca bu yöntemin bellek gereksinimi fazladır.

4.1.2.1.3 Doğrusal tahmin edici süzgeç yöntemi

Arabellekteki çerçevelerdeki piksellere doğrusal tahmin edici süzgeç uygulayarak arka plan modelini oluşturmaya çalışır (Toyama vd. 1999). Diğer yöntemlere göre bellek gereksinimi daha fazladır. Süzgeç katsayıları her bir çerçeve zamanında örnekleme kovaryansına göre hesaplanır (Arabellekteki çerçevelerin kovaryansı hesaplanarak süzgeç katsayıları bulunur). Bu işlem çok karmaşık olduğu için gerçek zaman uygulamalarında kullanılmaz. Arka plan çıkarımında sık rastlanan bir yöntem değildir.

4.1.2.1.4 Parametrik olmayan model yöntemi

Yukarıda bahsedilen tekniklerden farklı olarak arka plan tahmini yaparken piksel yoğunluk $f(u)$ fonksiyonunu kullanır (Elgammal vd. 2000). Piksel yoğunluk fonksiyonunun parametrik olmayan kestirimini oluşturmak için geçmiş bütün çerçevelere bakılır.

$$f(u) = \frac{1}{L} \sum_{i=1}^L K(u - u_i) \quad (4.2)$$

Eşitlik (4.2)'de L geçmiş çerçeve sayısını, $K(*)$ Gauss yoğunluk fonksiyonunu, u ise çerçevedeki x, y koordinatında bulunan pikseli gösterir. Bu yöntem video çerçevelerine uygulandığında iyi sonuç üretir; fakat arka plan modellemesinde ve ön plan belirlenmesinde işlem karmaşıklığı yüksektir. Çünkü her bir piksel için (4.2) eşitliği işlenir.

4.1.2.2 Özyinelemeli yöntemler

Özyinelemeli yöntemler arka plan modeli oluştururken arabellek kullanmazlar. Bunun yerine basit bir arka plan modelini her girdi çerçevesine göre özyinelemeli olarak güncellerler (Oliver vd. 2000, Kaewtrakulpong ve Bowden 2001, Magee 2002). Bunun bir sonucu olarak geçmiş video çerçeveleri arka plan modeli üzerinde doğrudan olmasa da dolaylı bir etkiye sahiptir. Özyinelemesiz yöntemlerle kıyaslandığında özyinelemeli

yöntemler daha az bellek gereksinimine sahiptirler; fakat arka plan modellemesinde yapılan bir hatanın düzeltilmesi daha uzun sürer. En çok kullanılan özyinelemeli yöntemler:

4.1.2.2.1 Yaklaşık ortanca süzgeç yöntemi

Özyinelemesiz ortanca süzgeç yönteminin başarısını gören McFarlane and Schofield, özyinelemesiz yöntemin avantajlarını kullanarak özyinelemeli yeni bir etkin çalışan yöntem geliştirmişlerdir. Çerçeve farkı yönteminden sonra en düşük işlem yüküne sahip yöntemdir.

Bu yöntemin çalışma prensibi; girdi olarak gelen pikselin değeri arka plan olarak kabul edilen piksel değerinden büyükse arka plan piksel değeri 1 artırılır ve aynı şekilde girdi olarak gelen pikselin değeri arka plan piksel değerinden küçükse arka plan değeri 1 azaltılır. Bu şekilde devam edilirse zamanla arka plan, girdi piksellerin yarısının değerinin arka plan değerinden büyük, yarısının da arka plan piksellerinin değerinden küçük olduğu bir tahmine yakınsar, yani arka plan yaklaşık olarak ortanca değer olur (Yakınsama süresi çerçeve sıklığına göre değişebilir).

Bu yöntemle elde edilecek sonuçlar, karmaşık yöntemlerle elde edilebilecek sonuçlara yakındır ve bu yöntemin işlemin karmaşıklığı ve bellek gereksinimi çerçeve farkı yönteminden çok farklı değildir.

4.1.2.2.2 Kalman süzgeci yöntemi

Genel Kalman süzgeci, Gauss gürültüsü altındaki doğrusal dinamik sistemleri incelemede geniş bir kullanım alanına sahiptir. Arka plan modellemesi için kullanılan çok değişik sürümleri vardır (Ridder vd. 1995, Wren vd. 1997, Heikkila ve Silven 1999). Bu sürümler izlemede kullanılacak durum değişkenlerinin kapsadığı uzay bakımından farklılık gösterirler. En basit sürümü sadece parlaklık yoğunluğu değerini kullanırken Karmann ve von Brandt hem parlaklık yoğunluğunu hem de parlaklık yoğunluğunun zamana göre türevini kullanır. Koller, Weber ve Malik ise parlaklık

yoğunluğunu ve parlaklık yoğunluğunun uzamsal türevini kullanır. Genel olarak durum değişkenleri uzayı (4.3) numaralı eşitlik yardımıyla hesaplanır.

$$x_n = Ax_{n-1} + K (u_{n-1} - HAx_{n-1}) \quad (4.3)$$

(4.3)'te A içsel dinamiklerin oluşturduğu parametre matrisini, K Kalman kazanç matrisini ve H de ölçüm matrisini gösterir. Arka plan yoğunluk değeri ve zamana göre türevi özyinelemeli olarak (4.4) numaralı eşitliği kullanılarak güncellenir.

$$\begin{bmatrix} B_t \\ B_t' \end{bmatrix} = A \cdot \begin{bmatrix} B_{t-1} \\ B_{t-1}' \end{bmatrix} + K_t \cdot \left(I_t - H \cdot A \cdot \begin{bmatrix} B_{t-1} \\ B_{t-1}' \end{bmatrix} \right) \quad (4.4)$$

(4.4)'te B_t , B_{t-1} t ve t-1 zamanındaki arka plan yoğunluk değerini, B_t , B_{t-1} t ve t-1 zamanındaki arka plan yoğunluk değerinin zamana göre türevini, A içsel dinamiklerin oluşturduğu parametre matrisini, I parlaklığı, K kalman matrisini ve H'de ölçüm matrisini göstermektedir.

4.1.2.2.3 Gauss fonksiyonları karışımı yöntemi

Yüksek karmaşıklığa sahip yöntemler arasında literatürde iki yöntem baskın olarak görülür. Bunlardan biri Kalman süzgeç yöntemi diğeri ise Gauss karışımı (MoG) yöntemidir (McIvar 2000, Kaewtrakulpong ve Bowden 2001, Sun vd. 2006). Her ikisinin de kendi avantajları vardır. Fakat Kalman süzgeci yönteminde hareket eden nesnelerin hareket izleri yok edilememektedir. Bu da birçok uygulama için sorun oluşturduğundan çok fazla tercih edilmeyen bir yöntem olmuştur. Bunların dışında MoG birden fazla çevresel faktörlerin etkisini süzebildiğinden daha sağlam, daha doğru sonuçlar üretir. Örneğin gökyüzü ve sallanan yaprağın olduğu bir görüntüde iki farklı şekil vardır, gökyüzü ve yaprak. Kalman süzgeci yöntemi tek bir Gauss fonksiyonu kullandığı için tek bir şekli ayırabilir. Ya gökyüzünü ayırır ya da yaprağı ayırabilir, ikisini birden ayıramaz. MoG yönteminde ise birden fazla Gauss fonksiyonu kullanıldığı için her iki şekilde başarılı bir şekilde ayrıştırılır.

MoG yönteminde arka plan, renklerin açıklık, koyuluk değerlerinin oluşturduğu çerçeveler olarak kabul edilmez. Bunun yerine arka plan modeli parametrikdir. Her pikselin yeri bir grup Gauss fonksiyonu tarafından belirlenir. Bu fonksiyonlar toplanarak olasılık dağılım fonksiyonu F elde edilir.

$$F(i_t = \mu) = \sum_{i=1}^k \omega_{i,t} \cdot \eta(\mu, \sigma) \quad (4.5)$$

Her bir Gauss fonksiyonunun ortalamasını, bir sonraki çerçevede gelecek olan piksel değerlerinin eğitim sonucu elde edilmiş tahmini olarak düşünülebiliriz. Bu işlem uygulanırken piksellerin genelde arka plana ait olduğu kabul edilir. Her bir piksel için genelde 3-5 Gauss fonksiyonu vardır. Fakat bu sayı bellek sınırlamasına göre değişebilir.

Bir pikselin arka plana ya da ön plana ait olduğunu belirleyebilmek için Gauss bileşenlerinin (fonksiyonlarının) ilgili pikseli izleyip izlemediğine bakarız. Eğer pikselin değeri arka plan bileşeninin standart sapmasının ölçekleme faktörüyle çarpımından küçük veya eşitse bu piksel arka plan bileşenidir, diğer durumda ise ön plan bileşenidir. MoG algoritması aşağıdaki adımlardan oluşur:

1- Girdi piksellerini, girdi piksellerine komşu piksellerin (bileşenlerin) ortalamasıyla karşılaştır. Eğer pikselin değeri bileşenlerin ortalamasına yakınsa bu bileşen uyumlu bileşen (eşleşen bileşen) olarak kabul edilir. Yani bir bileşenin eşleşen bileşen olması için piksel ile bileşen ortalamasının farkının mutlak değerinin bileşenin standart sapmasının ölçekleme değeriyle çarpımından küçük olması gerekir.

$$|i_i - \mu_{i,t-1}| \leq D \cdot \sigma \quad (4.6)$$

2- Yeni pikselin değerini yansıtmak için bileşen değerlerini (ω , u , σ) güncelle. Eşleşen bileşen için w değeri artar, σ değeri azalır ve u değeri ise pikselin değerine göre hafifçe dürtülür. Eşleşen bileşenler için parametrelerin güncellenmesi aşağıdaki

gibi olur. Bu bileşen değerlerinin değişme hızı bütün denklemlerde bulunan öğrenme hızı ρ 'ye göre değişir.

$$\omega_{i,t} = (1 - \alpha) \cdot \omega_{i,t-1} + \alpha \quad (4.7)$$

$$\mu_{i,t} = (1 - \rho) \cdot \mu_{i,t-1} + \rho I_t \quad (4.8)$$

$$\sigma_{i,t}^2 = (1 - \rho) \cdot \sigma_{i,t-1}^2 + \rho (I_t - \mu_{i,t})^2 \quad (4.9)$$

α , kullanıcı tarafından belirlenen $0 \leq \alpha \leq 1$ aralığında öğrenme hızıdır. ρ ise parametreler için öğrenme hızıdır ve yaklaşık olarak

$$\rho \approx \frac{\alpha}{\omega_{i,t}} \quad (4.10)$$

formülüyle bulunur. Eşleşmeyen bileşen bulunduğunda en düşük ağırlığa sahip bileşen, büyük varyans, düşük ağırlığa sahip bileşenle değiştirilir. Geriye kalan bileşenlerin ise ω değeri üstel olarak azalırken σ ve μ değeri sabit kalır.

3- Hangi bileşenlerin arka plana ait olduğunu belirle.

a. Bileşenleri ω/σ oranına göre büyükten küçüğe doğru sırala. İlk sıralardaki bileşenler düşük varyans değerine, yüksek ağırlık değerine (yoğunluk) sahip bileşenlerdir. Bu özelliklerde tipik arka plan özellikleridir.

b. Bileşen ağırlığı ω 'ya eşik değeri uygula.

$$\sum_{k=i}^{i_M} \omega_{k,t} \geq \Gamma \quad (4.11)$$

(4.11) eşitliğini sağlayan pikselleri arka plan pikselleri olarak belirle

4- Ön plan piksellerini belirle. Arka plan modelinde tanımlanamayan pikseller ön plan pikselleridir.

4.1.2.2.4 Özarkaplan yöntemi

Bu yöntemde video çerçevelerindeki pikselleri gruplamak için hareket kullanılır; çünkü arka plan sabit olarak kabul edilir. Hareket eden nesnelere bulunabilmesi için arka planı modelleyen özuzay oluşturulur. Özuzay gün içinde değişen ışık çeşitlerini ve hava koşullarını da tanımlar. Özuzay basit standart bilgisayar grafik teknikleri kullanılarak oluşturulur. Ana bileşenler analizi bunlardan biridir (bir veri kümesini daha az sayıdaki boyutta ifade etmenin en küçük hatalar karesi (least square error) bağlamında en az hatalı yolunu verir. Eğer veriyi k tane boyuta indirgeyeceksek, veri içindeki varyansı maksimize eden k tane birbirine dik (orthogonal) öz vektörü (eigen vector) bulmaya dayanır (Ching vd. 2005, Sun vd. 2006, Deane 2007).

Özarkaplan yönteminin temel adımları:

1. n tane çerçeve A matrisinin kolonları olarak yeniden düzenlenir.
2. Kovaryans matrisi $C=AA^t$ hesaplanır.
3. C kullanılarak özdeğerlerin köşegen matrisi L ve özvektör matrisi Φ hesaplanır.
4. Sadece ilk M tane özarkaplan değeri saklanır.
5. Yeni bir girdi çerçevesi geldiğinde M özvektörlerinin alt uzayındaki değeri bulunur ve I' olarak kabul edilir.
6. $I-I'$ farkı hesaplanır. Alt uzay durgun arka plan nesnelere düzgün olarak modellediği için bu farkın sonucu ön plan nesnesini verir.

4.1.3 Ön plan bulunması

Ön plan bulunması aşamasında girdi video çerçeveleri arka plan modeliyle karşılaştırılır ve girdi video çerçevelerindeki ön plan pikselleri bulunur. Yukarıda anlatılan yöntemlerden parametrik olmayan yöntem ve MoG haricindeki yöntemler arka plan modeli olarak tek bir yalın görüntüyü arka plan nesnesi olarak kullanırlar. En çok

kullanılan ön plan bulma yöntemi girdi çerçevesindeki piksel değerlerinin arka plan modelindeki değerlere yakın olup olmadığına bakıldığı yöntemdir (Fuentes ve Velastin 2003).

$$|I_t(x, y) - B_t(x, y)| > T \quad (4.12)$$

Diğer bir popüler yöntem de normalleştirilmiş istatistiksel verilere eşik değerinin uygulandığı yöntemdir.

$$\frac{|I_t(x, y) - B_t(x, y) - \mu_d|}{\sigma_d} > T_s \quad (4.13)$$

μ ve σ her x, y uzamsal noktası için $I_t(x, y) - B_t(x, y)$ farkının ortalaması ve standart sapma değeridir. Ön plan bulunmasında kullanılan yöntemlerin çoğu eşik değerlerini deneysel yollarla belirlerler.

4.1.4 Veri doğrulaması

Veri doğrulaması arka plan modeli kullanılarak elde edilen ön plana ait daha sağlam, daha doğru bilgilerin elde edilebilmesi için videolarda arka plan ve ön plan ayırımında uygulanan aşamadır (Heikkila ve Silven 1999, Toyama vd. 1999, Ching vd. 2005). Yukarıda bahsedilen algoritmaların tümünde görülen 3 temel problem vardır.

- Birbirine komşu iki piksel arasındaki bağıntı dikkate alınmaz.
- Arka plan adaptasyon hızı ön planda hareket eden nesnelerin hızıyla eşleşmeyebilir, adaptasyon hızı düşük olabilir.
- Hareket eden yapraklar üzerindeki durgun olmayan pikseller ve hareket eden ön plan nesnelerinin gölgeleri gerçek ön plan nesnesiymiş gibi algılanır.

Ön planın bulunmasında kullanılacak ön plan maskesi üzerinde rastgele dağılmış fazladan alanların oluşması ya da eksik alanların olması yukarıda bahsedilen ilk problemden kaynaklanan bir durumdur. Oluşan bu sorunu ortadan kaldırmak için uygulanan yöntemlerin başında bağlı bileşenleri gruplama yöntemleri (biçimsel süzgeçler) gelir (Heikkila ve Silven 1999, Toyama vd. 1999). Morfolojik süzgeçler kullanılarak ayırık halde bulunan ön plan pikselleri ortadan kaldırılır ve arada bağ olmayan birbirine yakın ön plan alanlarını birleştirir. Uygulamaların çoğu ön plan nesnelere belirli bir büyüklükte olması gerektiğini kabul ederler. Bağlı bileşenleri gruplama yapılarak bağlı ön plan nesnelere bulunması ve hareket eden gerçek ön plan nesnesi olamayacak kadar küçük nesnelere ayıklanması sağlanır.

Arka plan adaptasyon hızı ön plan nesnelere hızından küçükse ön plana ait olmayan hatalı ön plan nesnelere oluşmasına neden olur. Bu durum hayalet (ghost) etkisi olarak bilinir. Eğer arka plan uyumu çok hızlı olursa bu durumda da hareket eden nesnelere arka plan nesnesi olarak bulunur. Değişik adaptasyon hızına sahip birden fazla arka plan modeli kullanmak ve performansı artırmak için periyodik olarak arka plan modelleri arasında çapraz doğrulama yapmak yukarıda bahsedilen problemlere çözümünde kullanılan basit yaklaşımlardır.

Veri doğrulanmasında optiksel akış, renk bölütlemesi gibi daha karmaşık yöntemlerde kullanılabilir. Farklı açılardan aynı zamanda aynı yeri çeken birden fazla kamera tarafından kaydedilmiş bir görüntü üzerinde çalışılıyorsa kameralar arasındaki tamamen farklı bilgiler kullanılarak derinlik bilgisi elde edilir. Derinlik bilgisi ön plan bulunmasında çok yardımcı olur; çünkü ön plan nesnelere arka plan nesnelere göre kameraya her zaman daha yakındır.

Hareket eden yaprakların ön plan olarak algılanması problemini MoG gibi karmaşık arka plan modelleme algoritmaları kullanarak ortadan kaldırılabiliriz. Fakat hareket eden nesnelere gölgelerinin de gerçek ön plan nesnesi olarak algılanması problemini çözmek için bu iş için geliştirilmiş yöntemlerin kullanılması gerekir.

5. GERÇEK ZAMANLI VİDEOLARDA KOD ÇİZELGESİ YÖNTEMİ KULLANARAK ÖN PLAN ARKA PLAN AYRIMI

Durgun bir kameradan alınan görüntülerdeki hareket eden nesnelerin ayrılması görsel uygulamaların temelini oluşturmaktadır. Hareket eden nesnelerin arka plandan ayrılmasında uygulanan en temel yaklaşım arka plan çıkarımı yönteminin kullanılmasıdır. Bu yöntemin arkasındaki temel fikir gelen video çerçevelerini belirli bir arka plan modelinden çıkarılmasıdır. Bu çıkarım sonucunda arka plan nesnelere hareket eden ön plan nesnelere ayrılmış olur.

Hareket eden ön plan nesnelere belirlenebilmesi için öncelikle arka plan modelinin oluşturulması gerekir. Arka plan modelini en basit şekilde her bir pikselin parlaklık değerine bakarak oluşturabiliriz. Fakat bu yöntem, daha geniş bir ifadeyle bir pikselin arka plan pikseli olduğunu söylemek için tek bir değer kullanan yöntemler sallanan ağaçlar gibi karmaşık arka plan nesnelere içeren görüntülerde kullanılamaz. Bu gibi görüntülerde, karmaşık, hareket eden arka plan nesnelere içeren görüntülerde pikselin alabileceği bütün değerleri içeren arka plan modelinin oluşturulması gerekir. Arka plan modeli oluşturulurken bazı algoritmalar ön plan nesnelere olmadığını varsayarak arka plan modeli oluştururlar. Fakat bu dış ortam için pek mümkün görünmemektedir. Algoritma tasarlanırken bu durum da göz önünde bulundurulmalıdır. Ön plan, arka plan ayrımını yaparken kullandığımız yöntemin kabul edilebilir sürede, sınırlı bellek imkânlarıyla tutarlı ve iyi sonuç üretmesi gerekir.

Bu bölümde kod çizelgesi yönteminden, kutu tabanlı kod çizelgesi yönteminden ve kutu tabanlı kod çizelgesi yönteminden bahsedilecektir. Her iki yöntemin kullandığı algoritmalar ve bu algoritmaların farklılıklarından bahsedilecektir.

5.1 Kod Çizelgesi Yöntemi

Kod çizelgesi yöntemi ön plan arka plan ayrımı yaparken arka plan modeli kullanan bir yöntemdir. Arka plan modeli olarak kod çizelgesini kullanır. Kod çizelgesi yöntemi arka plan modeli kullanan diğer yöntemlerle kıyaslandığında bellek kullanımı ve hız

bakımından daha verimlidir (Kim vd. 2005). Temel kod çizelgesi algoritmasına katmanlı modelleme ve uyarlanabilir kod çizelgesi güncellemesi eklenerek daha iyi sonuçlar elde edilebilir (Kim vd. 2005).

Kod çizelgesi yöntemi herhangi bir parametrik tahmin yürütmeden zamana göre değişen ve değişmeyen piksel değerlerini kullanarak kod çizelgesini oluşturur. Kod çizelgesinin temel yapıtaşı kod sözcükleridir. Karmaşık arka plan nesnelere birden fazla kod sözcüğüyle ifade edilebilir. Kod çizelgesi yöntemini belirleyen en önemli özellikleri şöyledir:

1. Uzun zaman periyodu boyunca hareketli ve hareketsiz arka plan nesnelere sınırlı bellek altında modelleyebilen uyarlanır (adaptive) ve derli toplu (compact) arka plan modelidir. Böylece değişen arka planları iyi modelleyebilir.
2. Arka plan modelini güncelleyerek yerel ve genel aydınlanma değişiklikleriyle başa çıkabilme kabiliyetine sahiptir.
3. Arka plan modeli oluşturulma aşamasında görüntülerde ön plan nesnelere izin verir.

Kod çizelgesi yöntemi Kohonen (1988) tarafından ifade edilen nicemleme/gruplandırma tekniğini kullanarak arka plan modelini oluşturur. Her bir piksel bir ya da daha fazla kod sözcüğü içeren kod sözcükleri kümesi oluşturur. Kod sözcüklerini kümelerken renk değişimini ve parlaklık sınırlarını kullanır. Her bir piksel için oluşturulan kod sözcükleri kümesindeki kod sözcüğü sayısı aynı olmayabilir. Kod çizelgesi algoritmasında kullanılan çerçevelerdeki pikselin alabileceği değişik durumlara göre kod sözcüğü kümesindeki kod sözcüğü sayısı değişir. Bu oluşturulan kod sözcükleri kümeleri Gauss dağılımına ya da diğer parametrik dağılımlardan birine uymak zorunda değildir. Bir pikseldeki dağılım tek bir Gauss fonksiyonuyla ifade edilebilir ve normal bir dağılım olsa bile, o piksel için birden fazla kod sözcüğü olabilir (Kim vd. 2005).

Ön plan arka plan ayrımı aşamasında da renk değişimi ve parlaklık sınırlarına bakılarak ilgili pikselin arka plan ya da ön plan pikseli olduğuna karar verilir. Arka plan modelindeki kod sözcüklerinden belirgin şekilde farklı renk değişimine sahip ve parlaklık sınırları dışında kalan pikseller ön plan piksellerini içerir. Ön plan pikselleri dışındaki piksellerde arka plan pikselleridir (Kim vd. 2005).

Beklenen performansı elde edebilmek için arka plandaki belirgin durumları tutan bir kod çizelgesi oluşturulur. Daha sonra ilgili pikselin yeni değerleri daha önce kaydedilmiş piksel değeriyle karşılaştırılır. Karşılaştırılan değer ilk değere yeterince yakınsa aynı piksel değeri olarak kabul edilir. Eğer yakın bir değer değilse o pikselle ilişkili yeni bir kod sözcüğü oluşturulur. Elde edilen kod sözcükleri arka plan modelini oluşturan piksel olmaya aday pikselleri barındırır.

Kod çizelgesi yöntemi arka plan modeli olarak video çerçevelerindeki pikseller için oluşturulan kod çizelgesini kullanır. Belirli sayıdaki çerçeve için arka plan modelini oluşturur. Arka plan modelini oluştururken ön plan arka plan ayrımını gerçekleştirmez. Kod çizelgesini (arka plan modeli) oluşturduktan sonra gereksiz kod sözcüklerini kod çizelgesinden çıkarır. Elde edilen kod çizelgesi sadece arka plan nesnelere ait bilgileri içerir. Arka plan nesnelere ait bilgileri içeren kod çizelgesi oluşturulduktan sonra ön plan arka plan ayrımını gerçekleştirir. Ön plan arka plan ayrımı gerçekleştirilirken aynı zamanda arka plan modelinin güncellenmesi de gerçekleştirilir. Böylece değişen arka plan şartlarına uyumlu arka plan modeli elde edilmiş olur.

Kod çizelgesini oluşturan temel bileşenlere kod sözcükleri denir. Kod çizelgesi oluşturulurken kod sözcüklerinden faydalanılır. $X=\{x_1, x_2, \dots, x_N\}$ bir pikselin arka plan modeli oluşturulması aşaması boyunca alabileceği RGB değerleri olsun. N arka plan modeli oluşturulmasında kullanılan çerçeve sayısını gösterecek şekilde $C=\{c_1, c_2, \dots, c_L\}$ bir piksel için oluşturulan L tane kod sözcüğü içeren kod sözcüğü kümesi olsun. Herbir piksel için L sayısı farklı olabilir.

Her bir kod sözcüğü c_i , $i=1,2,\dots,L$ ortalama RGB değerlerini içeren v_i vektörlerinden, maksimum ve minimum parlaklık bilgileri olan 2 değişkenden $\text{auxb}=\{\hat{I}, \check{I}\}$ ve 4 geçici $\text{auxt}=\{f, \lambda, p, q\}$ değişkenden oluşur.

v_i : $\bar{R}_i, \bar{G}_i, \bar{B}_i$ ortalama değerleri içeren vektörü gösteren değişken

\hat{I} : Kod sözcüğünün aldığı maksimum parlaklık değerini gösteren değişken

\check{I} : Kod sözcüğünün aldığı minimum parlaklık değerini gösteren değişken

f : Kod sözcüğüne erişim sıklığını gösteren değişken

λ : mnrl (maximum negative run length) Kod sözcüğüne erişilmeyen en uzun süreyi gösteren değişken. Bu değişken kullanılarak kod çizelgesindeki gereksiz kod sözcükleri atılır.

p : Kod sözcüğüne ilk erişim zamanı gösteren değişken

q : Kod sözcüğüne son erişim zamanını gösteren değişken

5.1.1 Arka plan modelinin oluşturulması (Kod çizelgesinin oluşturulması)

Kod sözcüğünün yapısından bahsettikten sonra kod çizelgesi yönteminin kullandığı algorithmadan bahsedebiliriz. Kod çizelgesi algoritması Kim vd. (2005)'de verildiği şekliyle şöyledir:

I. $L \leftarrow 0, C \leftarrow \emptyset$,

II. for t=1 den N'ye tekrarla

(i). $x_t = (R, G, B), I \leftarrow \sqrt{R^2 + G^2 + B^2}$

(ii). (a) ve (b) şartlarını x_t 'ye göre sağlayan $C = \{c_i \mid 1 \leq i \leq L\}$ içinde c_m kod sözcüğünü bul

(a) $\delta(x_t, v_m) = \text{renkdeğişimi}(x_t, v_m) \leq \varepsilon_1$

(b) $\text{parlaklık}(I, (\hat{I}, \check{I})) = \text{doğru}$

(iii). Eğer $C = \emptyset$ veya eşleşen kod sözcüğü bulunamazsa, $L \leftarrow L + 1$ yap ve değerleri aşağıdaki gibi olacak şekilde yeni bir c_L kod sözcüğü oluştur

- $v_L \leftarrow (R, G, B)$
- $auxb_L \leftarrow (\hat{I}, \check{I})$
- $auxt_L \leftarrow (1, t-1, t, t)$

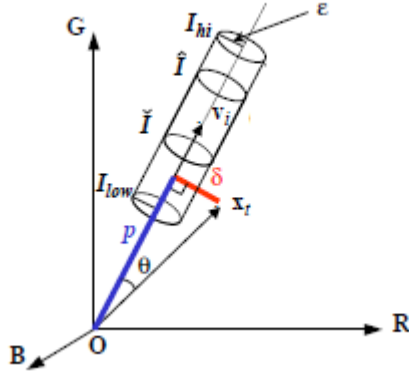
(iv). Diğer durumlarda $v_m = (\bar{R}_m, \bar{G}_m, \bar{B}_m)$, $auxb_m = (\hat{I}_m, \check{I}_m)$,

$auxt_m = (f_m, \lambda_m, p_m, q_m)$ değerlerine sahip eşleşen kod sözcüğü c_m 'yi aşağıdaki gibi güncelle

- $v_m \leftarrow \left(\frac{f_m \bar{R}_m + R}{f_m + 1}, \frac{f_m \bar{G}_m + G}{f_m + 1}, \frac{f_m \bar{B}_m + B}{f_m + 1} \right)$
- $auxb_m \leftarrow (\max(\hat{I}_m, I), \min(\check{I}_m, I))$
- $auxt_m \leftarrow (f_m + 1, \max(\lambda_m, t - q_m), p_m, t)$

for sonu.

III. Her kod sözcüğü için c_i $i=1,2,\dots,L$ λ_i 'yi $\lambda_i \leftarrow \max(\lambda_i, (N - q_i - q_i + p_i - 1))$ olacak şekilde güncelle.



Şekil 5.1 Kod çizelgesinde renk değişimi ve parlaklık hesaplamasında kullanılan şekil (Kim vd. 2005)

Kod çizelgesi algoritmasının 2. adımındaki (a) renk değişimi ve (b) parlaklık bilgisini hesaplayabilmek için şekil 5.1'den (Kim vd. 2005) faydalanır. Şekil 5.1'e göre oluşturulan renk değişimi ve mantıksal parlaklık fonksiyonları (5.1) ve (5.2) numaralı eşitliklerde verilmiştir.

$$\left\{ \begin{array}{l} \|x_t\|^2 = R^2 + G^2 + B^2 \\ \|v_i\|^2 = \bar{R}_i^2 + \bar{G}_i^2 + \bar{B}_i^2 \\ (x_t, v_i)^2 = (\bar{R}_i R + \bar{G}_i G + \bar{B}_i B)^2 \\ \rho^2 = \|x_t\|^2 \cos^2 \theta = \frac{(x_t, v_i)^2}{\|v_i\|^2} \\ \delta(x_t, v_m) = \text{renkdeğişimi}(x_t, v_m) = \sqrt{\|x_t\|^2 - \rho^2} \end{array} \right. \quad (5.1)$$

$$\text{parlaklık}(I, (\hat{I}, \tilde{I})) = \left\{ \begin{array}{l} \text{doğru eğer } I_{alçak} \leq \|x_t\| \leq I_{yüksek} \\ \text{yanlış diğer} \end{array} \right. \quad (5.2)$$

(5.2) numaralı eşitlikte parlaklığı belirli bir aralıkta tutmak için, yöntemin ani ışık değişikliklerinden ve gölgelerden etkilenmemesini sağlamak için $I_{alçak}$ ve $I_{yüksek}$ değerleri kullanılır. Bu değerleri hesaplarırken (5.3) ve (5.4) numaralı eşitliklerden faydalanılır. α , β parlaklık bilgisini belirli aralıkta tutabilmek için kullanılacak değişkenlerdir. Bu değişkenlerin alabileceği değer aralığı $0.4 \leq \alpha \leq 0.7$ ve $1.1 \leq \beta \leq 1.5$ şeklindedir. Bu değişkenlerin alabileceği değerler deneysel çalışmalar sonucu elde edilmişlerdir (Kim vd. 2005).

$$I_{alçak} = \alpha \hat{I} \quad (5.3)$$

$$I_{yüksek} = \min \left\{ \beta \hat{I}, \frac{\tilde{I}}{\alpha} \right\} \quad (5.4)$$

5.1.2 Gereksiz kod sözcüklerinin kod çizelgesinden çıkarılması

Kod çizelgesi yöntemi arka plan modelini oluştururken görüntülerde ön plan nesnelerinin de bulunabilceğinden bahsetmiştik. Bunu sağlayan gereksiz kod

sözcüklerinin kod çizelgesinden çıkarılması aşamasıdır. Bu aşamada ön plan nesnelere için ya da gürültü için oluşturulan kod sözcükleri kod çizelgesinden çıkarılarak arka plan nesnelere ait kod sözcüklerini içeren kod çizelgesi elde edilir.

Gereksiz kod sözcükleri atılmadan ilk elde edilen kod çizelgesine şişman kod çizelgesi denir. Arka plana ait piksellere erişim sıklığı ön plan piksellerine erişim sıklığından daha fazladır (Kim vd. 2005). Şişman kod çizelgesindeki ön plan nesnelere ait ve gürültüye ait kod sözcüklerini çıkarabilmek için (5.5) numaralı eşitlikten faydalanılır. M gereksiz kod sözcükleri temizlendikten sonra elde edilen kod çizelgesini gösterir.

$$M = \{c_m | c_m \in C \wedge \lambda_m \leq T_M\} \quad (5.5)$$

(5.5) numaralı eşitlikte, T_M arka plan modeli oluşturulurken kullanılan çerçeve sayısının yarısı olarak kabul edilir. λ_m daha önce bahsedildiği gibi kod sözcüğüne erişilmeyen en uzun süreyi gösterir. Bu durumda bir kod sözcüğüne model oluşturma aşamasındaki çerçeve sayısının yarısı kadar bir süre erişilmezse o kod sözcüğü kod çizelgesinden atılır. Yani arka plan kod sözcüklerine erişim sayısı model oluşturma aşamasında kullanılan çerçeve sayısının en az yarısı kadardır.

5.1.3 Kod çizelgesi yönteminde ön plan arka plan ayrımı

Ön plan arka plan ayrımı aşaması arka plan modeli oluşturma aşamasına benzerdir. Bu aşamada daha önce elde edilen kod çizelgesi kullanılarak gelen piksellerin ön plan pikseli ya da arka plan pikseli olduğuna karar verilir. Bu aşamada kullanılan algoritma Kim vd. (2005)'de verildiği şekliyle şöyledir:

$$I. x = (R, G, B), I \leftarrow \sqrt{R^2 + G^2 + B^2}$$

II. (5.5) numaralı eşitlikle elde edilen M kod çizelgesindeki her bir kod sözcüğü için (•) şartlarını x 'e göre sağlayan c_m kod sözcüğünü bul

- renkdeğişimi(x, c_m) $\leq \varepsilon_2$
- parlaklık($I, (\tilde{I}, \tilde{I})$) = *doğru*

Kod çizelgesi oluşturma algoritmasının ikinci aşamasındaki (iv) adımındaki gibi, eşleşen kod sözcüğünü güncelle

$$\text{III. } BGS(x) = \begin{cases} \text{ön plan, eşleşme yoksa} \\ \text{arka plan, diğer} \end{cases}$$

ε_2 ön plan arka plan ayırımında kullanılan eşik değerini, x gelen pikseli gösterir. Gelen piksel için oluşturulan kod çizelgesinde eşleşen herhangi bir kod sözcüğü bulunamazsa gelen piksel ön plan pikseli olarak işaretlenir, eşleşen kod sözcüğü bulunursa gelen piksel arka plan pikseli olarak işaretlenir.

5.1.4 Kod çizelgesi yönteminde arka plan modelinin güncellenmesi

Görüntü alınan yerdeki arka plan değişebileceği için arka plan modelinin değişen şartlara göre güncellenmesi gerekir. Örneğin park alanındaki araçlar park alanından çıkabilirler. Bu durumda kod çizelgesindeki park alanından ayrılan araçlara ait kod sözcüklerinin kod çizelgesinden çıkarılması gerekir. Kod çizelgesi yöntemi arka plan modelini güncelleyebilmek için ara kod çizelgesi H' 'ı ve T_H , T_{ekleme} ve $T_{çıkarma}$ şeklinde 3 değişken kullanır. T_H değişkenine bakılarak geçici kod çizelgesindeki kod sözcüklerinin kod çizelgesinden çıkarılıp çıkarılmayacağına karar verilir. T_{ekleme} değişkenine bakılarak geçici kod çizelgesinden ana kod çizelgesine taşınacak kod sözcüklerine karar verilir. $T_{çıkarma}$ değişkenine bakılarak ana kod çizelgesindeki silinecek kod sözcüklerine karar verilir. Ara kod çizelgesinde yeterince uzun süre bekleyen kod sözcükleri ana kod çizelgesine taşınır ve ana kod çizelgesinde uzun süre erişilmeyen kod sözcükleri ana kod çizelgesinden çıkarılır (Kim vd. 2005). Kod çizelgesi yöntemi arka plan modelini güncellerken aşağıdaki gibi bir algoritma kullanır:

- I. Arka plan modeli oluşturma aşamasında M ana kod çizelgesi elde edildikten sonra yeni bir H ara kod çizelgesi oluştur.
- II. Gelen pikseli için eşleşebilecek kod sözcüğünü M'de ara. Bulunursa kod sözcüğünü güncelle.
- III. Eşleşen kod sözcüğü bulunamazsa eşleşebilecek kod sözcüğünü H'de ara. Bulunursa kod sözcüğünü güncelle. Bulunamazsa yeni bir h kod sözcüğü oluştur ve H'ye ekle.
- IV. T_H 'ı kullanarak H kod çizelgesindeki gereksiz kod sözcüklerini temizle.

$$H \leftarrow H - \{h_i \mid h_i \in H \wedge \lambda_i \geq T_H\}$$

- V. Ara kod çizelgesi H'de yeterince uzun bekleyen kod sözcüklerini ana kod çizelgesi M'ye taşı.

$$M \leftarrow M \cup \{h_i \mid h_i \in H, h_i \text{ bekleme süresi} > T_{ekleme}\}$$

- VI. M'de uzun süre erişilmeyen kod sözcüklerini M'den çıkar.

$$M \leftarrow M - \{c_i \mid c_i \in M, c_i \text{ 'ye erişilmeyen süre} \geq T_{çıkarma}\}$$

Bu algoritma işletilerek arka plan modelinin güncellenmesi sağlanır.

5.2 Kutu Tabanlı Kod Çizelgesi Yöntemi

Kutu tabanlı kod çizelgesi yöntemi Bölüm 5.1'de bahsedilen kod çizelgesi yöntemi üzerine kurulmuş bir yöntemdir. Bu yöntem orijinal kod çizelgesindeki işlem sayısını azaltarak daha hızlı bir yöntem elde etmeyi hedeflemiştir (Tu vd. 2008). Kutu tabanlı kod çizelgesi yöntemi kod çizelgesi yöntemiyle aynı işleyişe sahiptir. Algoritma işleyişi genel anlamda aynıdır. Orijinal kod çizelgesinden farklı olarak kutu tabanlı kod çizelgesi yöntemi renk değişikliği yerine renk farkını kullanarak işlem sayısını azaltır. Renk değişikliği (5.1) numaralı eşitlikte verilmiştir. Renk farkı ise ilerleyen bölümlerde verilecektir. Kutu tabanlı kod çizelgesi yöntemi anlatılırken orijinal kod çizelgesiyle aynı olan kısımlardan bahsedilmeyecektir, bunun yerine iki yöntem arasındaki farklılıklarından bahsedilecektir.

Kutu tabanlı kod çizelgesinin de temel elemanı kod sözcükleridir. Kod sözcüklerinin yapısı orijinal kod çizelgesindeki kod sözcüklerinin yapısından farklıdır. Kod sözcüklerinin yapısında aşağıdaki değişkenler bulunur:

μ : μ_R, μ_G, μ_B için ortalama değerleri içeren vektörü gösteren değişken

f : Kod sözcüğüne erişim sıklığını gösteren değişken

λ : mnrl (maximum negative run length) Kod sözcüğüne erişilmeyen en uzun süreyi gösteren değişken. Bu değişken kullanılarak kod çizelgesindeki gereksiz kod sözcükleri atılır.

p : Kod sözcüğüne ilk erişim zamanı gösteren değişken

q : Kod sözcüğüne son erişim zamanını gösteren değişken

Kutu tabanlı kod çizelgesi yönteminin kullandığı kod sözcüklerinin yapısında \hat{I} , \check{I} maksimum ve minimum parlaklık değişkenlerini gösteren değişkenler bulunmaz. Bu iki değişkeni kod sözcüklerinden çıkararak daha az bellek kullanımını sağlar. Maksimum ve minimum parlaklık bilgisini kod sözcüklerinde tutmadığı için kod çizelgesi oluştururken ve ön plan arka plan ayrımı yaparken orijinal kod çizelgesi yönteminde kullanılan (5.2) numaralı eşitliği kullanmaz.

5.2.1 Kutu tabanlı kod çizelgesi yönteminde arka plan modelinin oluşturulması (Kutu tabanlı kod çizelgesinin oluşturulması)

Kutu tabanlı kod çizelgesi yönteminin kullandığı algoritma daha önce bahsedildiği gibi orijinal kod çizelgesi algoritmasına benzerdir; ancak kutu tabanlı kod çizelgesi yöntemi renk değişimi yerine renk farkını kullanır ve parlaklık fonksiyonunu kullanmaz. Kutu tabanlı kod çizelgesi yönteminin kullandığı algoritma Tu vd. (2008)'de belirtildiği şekliyle aşağıdaki gibidir:

I. $L \leftarrow 0, C \leftarrow \emptyset$,

II. for t=1 den N'ye tekrarla

(i). $x_t = (R, G, B)$

(ii). (a) şartını x_t 'ye göre sağlayan $C = \{c_i \mid 1 \leq i \leq L\}$ içinde c_m kod sözcüğünü bul

$$(a) \text{ renk farkı}(\mu_R, R) \leq \varepsilon_1 \wedge \text{ renk farkı}(\mu_G, G) \leq \varepsilon_1 \wedge \text{ renk farkı}(\mu_B, B) \leq \varepsilon_1$$

(iii). Eğer $C = \emptyset$ veya eşleşen kod sözcüğü bulunamazsa, $L \leftarrow L + 1$ yap ve değerleri aşağıdaki gibi olacak şekilde yeni bir c_L kod sözcüğü oluştur

$$\bullet \mu_L \leftarrow (R, G, B)$$

$$\bullet \text{ aux}_L \leftarrow (1, t - 1, t, t)$$

(iv). Diğer durumlarda $\mu_m = (\mu_{R_m}, \mu_{G_m}, \mu_{B_m})$, $\text{aux}_m = (f_m, \lambda_m, p_m, q_m)$

değerlerine sahip eşleşen kod sözcüğü c_m 'yi aşağıdaki gibi güncelle

$$\bullet \mu_m \leftarrow \left(\frac{f_m \mu_{R_m} + R}{f_m + 1}, \frac{f_m \mu_{G_m} + G}{f_m + 1}, \frac{f_m \mu_{B_m} + B}{f_m + 1} \right)$$

$$\bullet \text{ aux}_m \leftarrow (f_m + 1, \max(\lambda_m, t - q_m), p_m, t)$$

for sonu.

III. Her kod sözcüğü için c_i $i=1, 2, \dots, L$ λ_i 'yi $\lambda_i \leftarrow \max(\lambda_i, (N - q_i - q_i + p_i - 1))$ olacak şekilde güncelle.

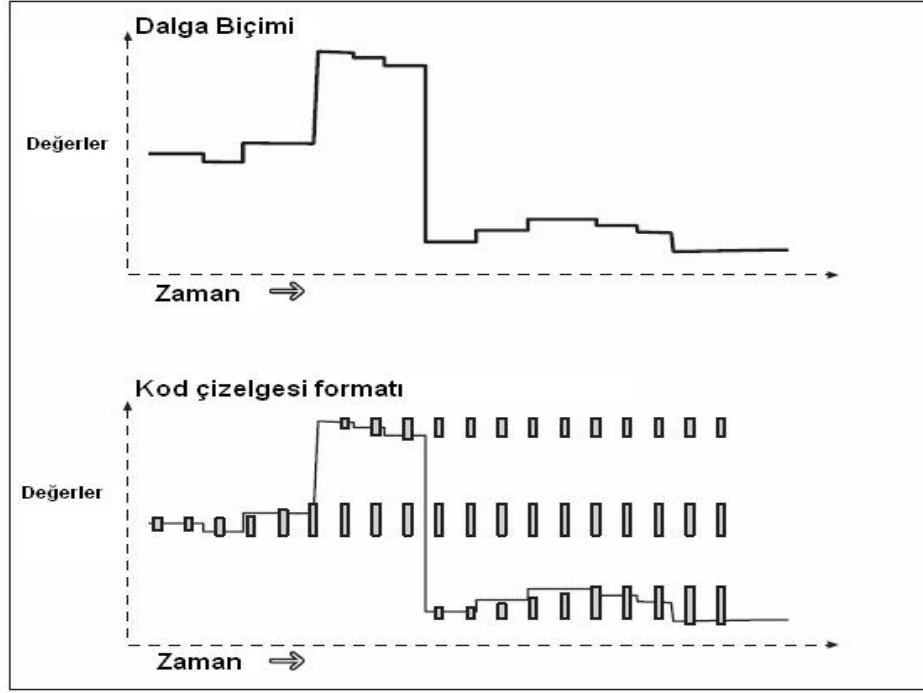
Kutu tabanlı kod çizelgesi yönteminin kullandığı algoritmanın temel farkı II. adımda (ii) maddesindeki (a) koşuludur. Burada renk farkı kullanılarak eşleşen kod sözcüğü olup olmadığına bakılır. Renk farkı eşitliği (5.6) numaralı eşitlikte verildiği gibidir.

$$\text{ renk farkı}(\mu_j, X_{ij}) = |\mu_j - X_{ij}| \quad (5.6)$$

(5.6) numaralı eşitlikte j R, G, B değerlerini gösterir. X_{ij} ise t zamanındaki pikseli göstermektedir. Kod çizelgesindeki ortalama R, G, B değerleriyle gelen pikselin R, G, B değerleri arasındaki fark renk farkı olarak ifade edilir. Renk farkına eşik değeri uygulayarak eşleşen kod sözcüğü olup olmadığına bakılır. Orijinal kod çizelgesi yönteminin kullandığı (5.2) numaralı eşitlik ile kutu tabanlı kod çizelgesi yönteminin

kullandığı (5.6) numaralı eşitlik karşılaştırıldığımızda (5.6) numaralı eşitlikteki işlem sayısının azlığını ve işlem basitliğini (sadece çıkarma ve mutlak değer) görebiliriz. Orijinal kod çizelgesi yönteminde kullanılan “renk değişimi” eşitliğinde 10 çarpma, 1 bölme, 1 karekök alma ve 5 toplama-çıkarma işlemi gerektirirken kutu tabanlı kod çizelgesi yönteminde kullanılan “renk farkı” eşitliği sadece 3 çıkarma işlemi gerektirir (Tu vd. 2008).

Pikseller için oluşturulan kod sözcükleri pikselin alabileceği değerlere göre genişleyebilen veya daralabilen küçük kutular olarak düşünülebilir. Yeni gelen piksellerin sınıflandırılmasında kutuların içinde olup olmadığına bakılır. Kutular R, G, B eksenlerini içerecek şekilde 3 boyutlu olarak düşünülür. Kutu tabanlı kod çizelgesini daha iyi açıklayabilmek için şekil 5.2’den faydalanabiliriz. Şekil 5.2’nin üst kısmında zaman içinde pikselin aldığı değerler gösterilmiştir. Şekil 5.2’nin alttaki kısmında ise ilk aşamada her bir piksel değeri için bir kutu oluşturulmuştur. Yeni gelen piksel değerleri bu kutuya yeterince yakınsa kutu yeni değeri de kapsayacak şekilde genişletilir. Yeni gelen piksel değeri kutuya yakın değilse yeni bir kutu oluşturulur ve yeni gelen değerlerin yakınlık uzaklığına göre genişler veya sabit kalır. Gürültüden kaynaklanabilecek ya da herhangi bir dış etkenden kaynaklanacak piksel değerinde meydana gelebilecek küçük değişiklikleri ortadan kaldırmak için kutu yakın değerleri de içine alacak şekilde genişletilir.



Şekil 5.2 Kod çizelgesi yönteminin çalışma prensibinin gösterimi

5.2.2 Kutu tabanlı kod çizelgesi yönteminde gereksiz kod sözcüklerinin kod çizelgesinden çıkarılması

Gereksiz kod sözcüklerinin kod çizelgesinden çıkarılması orijinal kod çizelgesi yöntemiyle aynıdır. Kod sözcüklerinin λ değişkenine bakılarak kod sözcüğünün kod çizelgesinden çıkarılıp çıkarılmayacağına karar verilir. Kod sözcüğüne erişilmeyen en uzun süre kod çizelgesi oluşturma aşamasında kullanılan çerçeve sayısının yarısından fazla ise o kod sözcüğü kod çizelgesinden çıkarılır. Kutu tabanlı kod çizelgesi yöntemi de (5.5) numaralı eşitliği kullanarak gereksiz kod sözcüklerini kod çizelgesinden çıkarır.

5.2.3 Kutu tabanlı kod çizelgesi yönteminde ön plan arka plan ayrımı

Kutu tabanlı kod çizelgesi yönteminde ön plan arka plan ayrımı yaparken arka plan modeli oluşturulması aşamasında elde edilen kod çizelgesi kullanılır. Orijinal kod çizelgesinden farklı olarak gelen pikselin kod çizelgesinde olup olmadığına bakılırken renk değişimi yerine renk farkını kullanır ve parlaklık fonksiyonunu kullanmaz. Kutu

tabanlı kod çizelgesi yöntemi ön plan arka plan ayrımı yaparken aşağıdaki algoritmayı kullanır (Tu vd. 2008):

I. $x = (R, G, B)$

II. (5.5) numaralı eşitlikle elde edilen M kod çizelgesindeki her bir kod sözcüğü için (•) şartını x 'e göre sağlayan c_m kod sözcüğünü bul

$$\bullet \text{ renk farkı}(\mu_R, R) \leq \varepsilon_2 \wedge \text{ renk farkı}(\mu_G, G) \leq \varepsilon_2 \wedge \text{ renk farkı}(\mu_B, B) \leq \varepsilon_2$$

Kod çizelgesi oluşturma algoritmasının ikinci aşamasındaki (iv) adımındaki gibi, eşleşen kod sözcüğünü güncelle

III. $BGS(x) = \begin{cases} \text{ön plan, eşleşme yoksa} \\ \text{arka plan, diğer} \end{cases}$

ε_2 ön plan arka plan ayrımında kullanılan eşik değerini, x gelen pikseli gösterir. Gelen pikselle eşleşen kod sözcüğü bulunursa piksel arka plan pikselidir. Eşleşen kod sözcüğü bulunamazsa piksel ön plan pikselidir.

5.2.4 Kutu tabanlı kod çizelgesi yönteminde arka plan modelinin güncellenmesi

Kutu tabanlı kod çizelgesi yöntemi arka plan modelini güncellerken orijinal kod çizelgesi yönteminin kullandığı algoritmanın aynısını kullanır. Yine bir ara kod çizelgesi oluşturulur. Ana kod çizelgesinde bulunamayan kod sözcükleri oluşturularak ara kod çizelgesine eklenir. Ara kod çizelgesinde yeterince uzun bekleyen kod sözcükleri ana kod çizelgesine alınır ve ana kod çizelgesinde yeterince uzun süre erişilmeyen kod sözcükleri ana kod çizelgesinden çıkarılarak arka plan modelinin güncellenmesi gerçekleştirilir. Ara çerçeveden çıkarma ve ana çerçeveye ekleme ve çıkarma sürelerini belirleyebilmek için sırasıyla T_H , T_{ekleme} ve $T_{çıkarma}$ değişkenlerini kullanır.

5.3 Geliştirilen Kutu Tabanlı Kod Çizelgesi Yöntemi

Arka plan modelini içeren piksel değerleri kod çizelgesindeki kod sözcüklerinde tutulmaktadır. Arka plan modeli oluşturulurken görüntüde bulunmayan arka plan pikselleri, ön plan arka plan ayrımı aşamasında görüldüğü zaman bu piksellerde ön plan nesnesi olarak bulunur. Hatalı bulunan bu değerler gerçekte arka planda meydana gelen küçük boyutlu değişikliklerden kaynaklanır. Rüzgâr hızının artması sonucu sallanan ağaç yapraklarının daha hızlı sallanmasıyla arka plana ait fakat arka plan modelinde olmayan arka plan pikselleri ön plan nesnesi olarak bulunur. Bunun dışında görüntüde bulunan gürültü miktarı fazla olursa gürültü de ön plan nesnesi olarak bulunabilir. Bunların dışında hareketli ön plan nesnelere gölgeleri de ön plan nesnesi olarak bulunabilmektedir. Ani ışık değişikliklerinde piksel parlaklık bilgileri değiştiği için arka plan pikselleri de ön plan olarak bulunabilmektedir. Bu sorunları ortadan kaldırmak için kutu tabanlı kod çizelgesi yönteminin kullandığı algoritmaya bazı eklemelerin yapılması gerekir. Bağlı bileşenler analizi yöntemi, gölge ve ışık maskeleri kutu tabanlı kod çizelgesi yöntemine eklenebilir. Bu yöntemlerin eklenmesiyle sırasıyla, ön planda arka plana ait hatalı bulunan pikseller ve ön plan olarak bulunan hareketli nesnelere gölgeleri çıkarılır, ani ışık değişikliklerinde hatalı sonuç elde edilmesi engellenir.

Geliştirilen kutu tabanlı kod çizelgesi yönteminde kullanılan kod sözcüklerinin yapısı kutu tabanlı kod çizelgesindeki kod sözcüklerinin yapısına benzerdir. Kod sözcüklerinin yapısında aşağıdaki değişkenler bulunur:

Kod sözcüklerinin yapısında aşağıdaki değişkenler bulunur:

μ : μ_H , μ_S , μ_V için ortalama değerleri içeren vektörü gösteren değişken

f : Kod sözcüğüne erişim sıklığını gösteren değişken

λ : mnrl (maximum negative run length) Kod sözcüğüne erişilmeyen en uzun süreyi gösteren değişken. Bu değişken kullanılarak kod çizelgesindeki gereksiz kod sözcükleri atılır.

p : Kod sözcüğüne ilk erişim zamanı gösteren değişken

q : Kod sözcüğüne son erişim zamanını gösteren değişken

5.3.1 Geliştirilen kutu tabanlı kod çizelgesi yönteminde arka plan modelinin oluşturulması

Geliştirilen kutu tabanlı kod çizelgesi yöntemi kutu tabanlı kod çizelgesi yönteminden farklı olarak RGB renk uzayı yerine HSV renk uzayını kullanır. HSV renk uzayının kullanılmasının nedenlerini aşağıdaki gibi verebiliriz:

2. HSV renk uzayında parlaklık bilgisi ve renk bilgisi ayrı ayrı elde edilebilir. V parlaklık bilgisini, H ve S de renk bilgisini içerir. Görüntünün parlaklığı değişirse V değeri etkilenir.
3. HSV renk uzayı insan görüşünü en iyi modelleyen renk uzayıdır. Böylece insan bakışıyla parlaklığın nasıl görüldüğü bilgisi elde edilir ve gölge bulmada daha iyi sonuçlar elde edilir.
4. RGB renk uzayı HSV renk uzayına göre gölge bulmada daha az hassastır (Cucchiara vd. 2001).
5. HSV renk uzayında gölge ve parlak nokta bulunurken daha az işlem gerekir.

Arka plan modeli oluşturulması aşamasında kutu tabanlı kod çizelgesinden farklı olarak pikselin RGB değerleri yerine HSV değerleri kullanılır. HSV değerlerine fark işlemi uygulanarak kod çizelgesinde eşleşen kod çizelgesi olup olmadığına bakılır. Renk uzayının değiştirilmesi haricinde kutu tabanlı kod çizelgesi yönteminde kullanılan algoritmanın aynısı kullanılır. Kod çizelgesinde eşleşen kod sözcüğü aranırken kullanılan renk farkı fonksiyonu da HSV renk uzayına göre değiştirilir. Buna göre renk farkı (5.7) numaralı eşitlikteki gibi olur. (5.7) numaralı eşitlikte j, H, S, V değerlerini ifade etmektedir. X_{ij} de t zamanında gelen pikselin H, S, V değerlerini ifade etmektedir.

$$\text{renkfarkı}(\mu_j, X_{ij}) = |\mu_j - X_{ij}| \quad (5.7)$$

Buna göre kod çizelgesi oluşturulması algoritması aşağıdaki gibi olur:

$$I. L \leftarrow 0, C \leftarrow \emptyset,$$

II. for t=1 den N'ye tekrarlarla

(i). $x_t = (H, S, V)$

(ii). (a) şartını x_t 'ye göre sağlayan $C = \{c_i \mid 1 \leq i \leq L\}$ içinde c_m kod sözcüğünü bul

(a) $renkfarki(\mu_H, H) \leq \varepsilon_1 \wedge renkfarki(\mu_S, S) \leq \varepsilon_1 \wedge renkfarki(\mu_V, V) \leq \varepsilon_1$

(iii). Eğer $C = \emptyset$ veya eşleşen kod sözcüğü bulunamazsa, $L \leftarrow L + 1$ yap ve değerleri aşağıdaki gibi olacak şekilde yeni bir c_L kod sözcüğü oluştur

• $\mu_L \leftarrow (H, S, V)$

• $auxt_L \leftarrow (1, t - 1, t, t)$

(iv). Diğer durumlarda $\mu_m = (\mu_{H_m}, \mu_{S_m}, \mu_{V_m})$, $auxt_m = (f_m, \lambda_m, p_m, q_m)$

değerlerine sahip eşleşen kod sözcüğü c_m 'yi aşağıdaki gibi güncelle

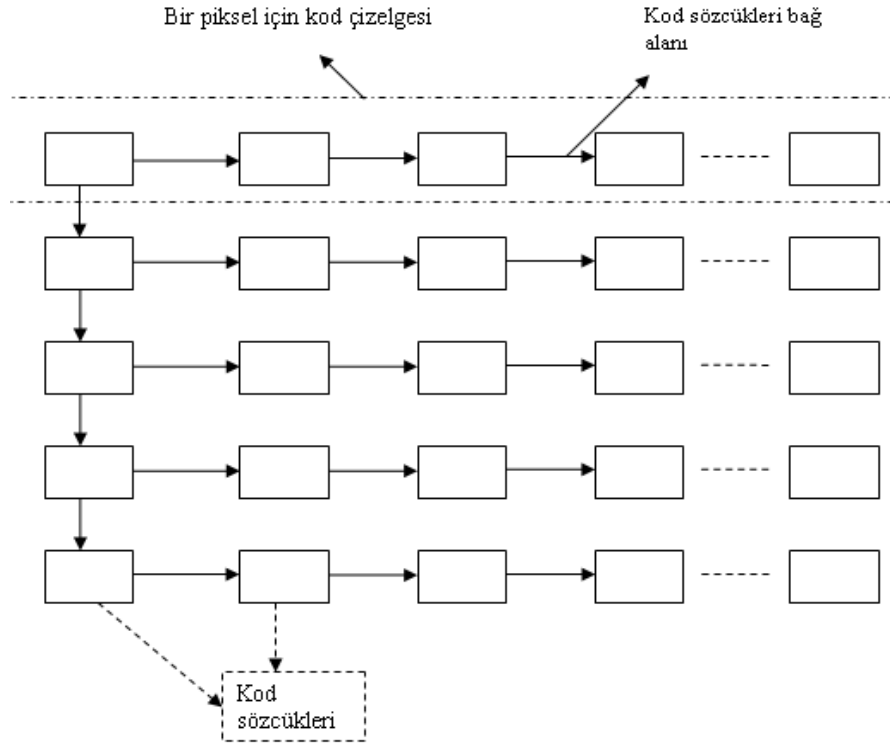
• $\mu_m \leftarrow \left(\frac{f_m \mu_{H_m} + H}{f_m + 1}, \frac{f_m \mu_{S_m} + S}{f_m + 1}, \frac{f_m \mu_{V_m} + V}{f_m + 1} \right)$

• $auxt_m \leftarrow (f_m + 1, \max(\lambda_m, t - q_m), p_m, t)$

for sonu.

III. Her kod sözcüğü için $c_i \ i=1, 2, \dots, L$ λ_i 'yi $\lambda_i \leftarrow \max(\lambda_i, (N - q_i - q_i + p_i - 1))$ olacak şekilde güncelle.

Kod çizelgesi oluşturma algoritması işletildikten sonra şekil 5.2'de mantıksal bellek gösterimi verilen kod çizelgesi elde edilir.



Şekil 5.3 Kutu tabanlı kod çizelgesinin mantıksal bellek gösterimi

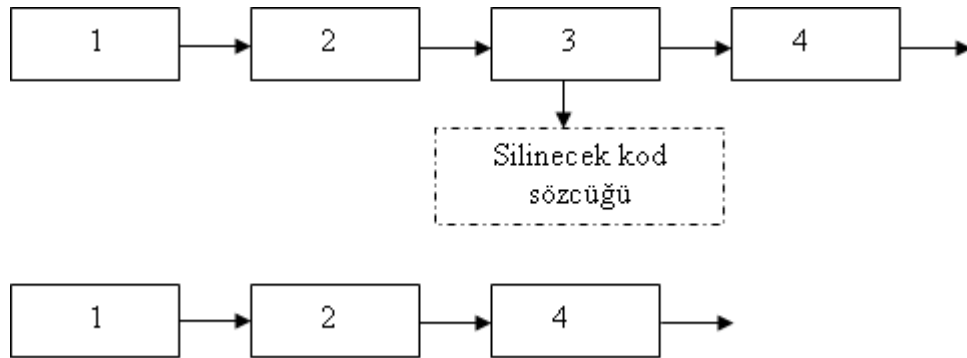
Şekil 5.3’de kutu tabanlı kod çizelgesinin mantıksal bellek gösterimine yer verilmiştir. Bu gösterime göre kutu tabanlı kod çizelgesi kod sözcüklerinden oluşur. Bir piksel ait kod sözcükleri birbirlerini gösterecek şekilde bir tek bağlaçlı liste olarak düşünülür. Tek bağlaçlı liste kullanımı, kod çizelgesinden kod sözcüğü çıkarılacağı zaman ya da kod çizelgesine yeni kod sözcüğü ekleneceği zaman büyük kolaylık sağlar. Kod sözcüklerinin sadece bağ alanları güncellenerek ekleme ve çıkarma işlemi gerçekleştirilir. Bağ alanlarının güncellenmesi örneği gereksiz kod sözcüklerinin kod çizelgesinden çıkarılmasında verilecektir.

5.3.2 Geliştirilen kutu tabanlı kod çizelgesi yönteminde gereksiz kod sözcüklerinin kod çizelgesinden atılması

Arka plan modeli oluşturulurken (kod çizelgesi oluşturulurken) kameranın görüntü aldığı ortamın ön plan nesnelere ve gürültüden arındırılmış olması gerekir. Bu dış ortam koşulları için mümkün olmadığından arka plan modeli oluşturulduktan sonra ön

plan nesnelere ve gürültünün arka plan modelinden çıkarılması gerekir. Kod çizelgesi yönteminde, kod çizelgesi oluşturulurken ön plan nesnelere ve gürültü için de kod sözcüğü oluşturulur. Ön plan arka plan ayrımı yapılmadan önce oluşturulan bu kod sözcüklerinin kod çizelgesinden çıkarılması gerekir. Aksi halde ön plan nesnelere de arka plan olarak bulunur. Kod sözcüklerinin arka plan kod sözcüğü olduğunun söylenebilmesi için işlem gören çerçeve sayısının en az yarısında erişilmiş olması gerekir (Kim vd. 2005). Gereksiz kod çizelgesinin kod çizelgesinden atılması kutu tabanlı kod çizelgesi yöntemiyle aynıdır ve (5.5) numaralı eşitliği kullanır. Kod sözcüklerindeki kod sözcüğüne erişilmeyen en uzun süreyi gösteren λ değişkeni kullanılarak kod sözcüğünün atılıp atılmayacağına karar verilir.

Kod çizelgesinde bir piksele ait kod sözcüklerinin tek bağlaçlı liste veri yapısında tutulduğunu arka plan modeli oluşturma aşamasında söylemiştik. Bu yapının tercih edilmesinin nedeni ekleme ve çıkarma işlemlerinde kolaylık sağlama ve işlem süresini kısaltmasıdır. Listedeki herhangi bir kod sözcüğü çıkarılacağı zaman diğer kod sözcüklerinde herhangi bir yer değiştirme işlemi yapılmaz. Sadece çıkarılacak kod sözcüğünü gösteren gösterge, çıkarılacak kod sözcüğünün gösterdiği kod sözcüğünü gösterecek şekilde güncellenir. Şekil 5.4'te kod sözcüğünün kod çizelgesinden çıkarılması gösterilmiştir.



Şekil 5.4 Kod çizelgesindeki gereksiz kod sözcüklerinin çıkarılması

Şekil 5.4'te gösterildiği üzere 3 numaralı kod sözcüğü kod çizelgesinden çıkarılacak kod sözcüğü olsun. 3 numaralı kod sözcüğünü gösteren 2 numaralı kod sözcüğünün bağ

alanı 3 numaralı kod sözcüğünün bağ alanının gösterdiği 4 numaralı kod sözcüğünü gösterecek şekilde güncellenir. Bu işlem sonucu şekil 5.4'te alttaki şekil ortaya çıkar. Silinen kod sözcüğü için alınan bellek alanı geri verilerek belleğin etkin kullanımı sağlanmış olur.

5.3.3 Geliştirilen kutu tabanlı kod çizelgesi yönteminde ön plan arka plan ayrımı

Kutu tabanlı kod çizelgesi yöntemine yapılan eklemeler daha çok bu adımda gerçekleştirilir. Gölge ve ışık maskesinin eklenmesiyle hatalı bulunan gölge ve parlak noktalar ön plan görüntüsünden çıkarılarak daha iyi sonuçlar elde edilir. Ön plan arka plan ayrımı yaparken referans arka plan modeli olarak oluşturulan kod çizelgesi kullanılır. Yeni gelen pikseller için piksellerin kod çizelgelerinde eşleşen kod sözcüğü olup olmadığına (5.7) numaralı eşitlikte gösterilen renk farkı eşitliği kullanılarak bakılır. Eşleşen kod sözcüğü bulunursa kod sözcüğü güncellenir. Eşleşen kod sözcüğü bulunamazsa piksel gölge maskesinden geçirilir. Gölge maskesinde piksel gölge olarak bulunursa bir sonraki piksel için yukarıdaki işlemler gerçekleştirilir. Gölge maskesinde gölge olarak bulunmazsa ışık maskesinden geçirilir. Işık maskesinde piksel parlak nokta olarak bulunursa bir sonraki piksele geçilir. Işık maskesinde ışık olarak bulunmazsa piksel ön plan nesnesine ait bir pikseldir ve ön plan maskesine eklenir. Bu anlatılanları aşağıdaki algorithmadaki gibi özetleyebiliriz:

I. $x = (H, S, V)$

II. (5.5) numaralı eşitlikle elde edilen M kod çizelgesindeki her bir kod sözcüğü için (•) şartını x 'e göre sağlayan c_m kod sözcüğünü bul

$$\bullet \text{renkfarkı}(\mu_H, H) \leq \varepsilon_2 \wedge \text{renkfarkı}(\mu_S, S) \leq \varepsilon_2 \wedge \text{renkfarkı}(\mu_V, V) \leq \varepsilon_2$$

Kod çizelgesi oluşturma algoritmasının ikinci aşamasındaki (iv) adımındaki gibi, eşleşen kod sözcüğünü güncelle

III. c_m bulunursa x 'i arka plan görüntüsüne ekle ve I. adıma git

IV. c_m bulunamazsa ve M kod çizelgesindeki her bir kod sözcüğü için (a) veya (b) şartlarını sağlayan c_n kod sözcüğünü bul

(a). $GM(x, c_n)$ =doğru

(b). $IM(x, c_n)$ =doğru

V. (a) veya (b) şartlarını sağlamazsa ön plan maskesine ekle ve I. adıma git

VI. (a) veya (b) şartlarını sağlamazsa herhangi bir işlem yapma ve I. adıma git

Yukarıdaki algoritma işletilerek gerçek ön plan nesnelere ortaya çıkarılır. Algoritmada kullanılan GM gölge maskesi ve IM ışık maskesi ilerleyen bölümlerde anlatılacaktır.

5.3.3.1 Gölge maskesi

Kod çizelgesi yöntemi ön plan arka plan ayrımı yaparken gelen pikselle kod çizelgesinde eşleşen kod sözcüğü olup olmadığına bakar. Kod çizelgesinde eşleşen kod sözcüğü bulamazsa pikseli ön plan olarak bulur. Hareketli nesnelere gölgeleri için kod çizelgesinde kod sözcüğü bulunmaz. Kod çizelgesi yönteminde erişim sayısı az olan piksellerin kod çizelgesinden çıkarıldığından gereksiz kod sözcüklerinin kod çizelgesinden çıkarılması bölümünde bahsetmiştik. Hareketli nesnelere için oluşturulan kod sözcüklerine erişim sayısı azdır. Çünkü hareketli nesnelere her zaman aynı piksel üzerinde olmazlar (Aynı piksel üzerinde olurlarsa durgun olurlar). Bu yüzden hareketli nesnelere gölgeleri için kod sözcüklerinin oluşturulması ve atılması hareketli nesnelere gibidirler. Hareketli nesnelere gölgeleri için kod çizelgesinde kod sözcüğü bulunmadığı için gölgeler de ön plan nesnesi olarak bulunurlar. Gölgelerin ön plan nesnesi olarak bulunmasını engelleyebilmek için kod çizelgesi yöntemine bazı özelliklerin eklenmesi gerekir. Geliştirilen kutu tabanlı kod çizelgesi yönteminde kutu tabanlı kod çizelgesi yöntemine gölge maskesi eklenerek hareketli nesnelere gölgelerinin ön plan nesnesi olarak bulunması engellenir.

Gölgeler görüntünün ışıklılık düzeyinde değişikliğe neden olurken, renkleri üzerinde çok az değişikliğe neden olurlar ya da hiç değişiklik yapmazlar (Horphrasat vd. 1999, Cucchiara vd. 2001, Han vd. 2003). Literatürde var olan yöntemler arasında kameranın yeri, ışık kaynağının yeri, ışığın geliş açısı gibi parametrelerle ilgili varsayımda

bulunmadan başarılı sonuç üreten, sağlam olan yöntem Cucchiara vd. (2001) künyesiyle verilen kaynakta bahsedilen yöntemdir (Doshi ve Trivedi 2006).

Bu yöntemde basitçe gölgenin görüntünün ışıklılık bilgisi üzerinde değişikliğe neden olduğu, renk bilgisini gerçek anlamda değiştirmedığı gerçeği üzerine kurulmuştur.

Video çerçevelerindeki gölgeleri bulabilmek için gölge maskesi oluşturulur. Gölge maskesi oluşturulurken arka plan modelindeki piksellere ait H, S, V değerlerini ve gelen çerçeveye ait piksellerin H, S, V değerlerini kullanır. Buna göre piksellerin gölge mi gerçek nesne mi olduğuna karar verilir ve bir gölge maskesi oluşturulur. Gölge maskesi oluşturmak için (5.8) numaralı eşitlikten faydalanılır.

$$GM_n(x, y) = \begin{cases} 1, & \text{eğer } \alpha \leq \frac{I_n^V(x, y)}{B_n^V(x, y)} \leq \beta \wedge (I_n^S(x, y) - B_n^S(x, y)) \leq T_S \wedge \\ & |I_n^H(x, y) - B_n^H(x, y)| \leq T_H \\ 0, & \text{diğer} \end{cases} \quad (5.8)$$

$I_n^H(x, y)$, $I_n^S(x, y)$, $I_n^V(x, y)$ gelen video çerçevesindeki n zamanında x, y koordinatındaki pikselin sırasıyla H, S ve V değerlerini göstermektedir. Aynı gösterim şekli B arka plana ait çerçeve için de kullanılmıştır. α , β , T_S , T_H değerleri dikkatlice seçilmesi gereken parametre değerleridir. (5.8) numaralı eşitliğe göre arka plan modelindeki ilgili piksele ait kod çizelgesindeki kod sözcüklerindeki parlaklık değeriyle yeni gelen pikselin parlaklık değerleri oranı α 'dan büyük β 'dan küçük ise ve yeni gelen piksel ile arka plan modelindeki ilgili piksele ait kod çizelgesindeki kod sözcüklerindeki doygunluk değeri arasındaki fark T_S doygunluk eşik değerinden küçük veya eşitse ve yeni gelen piksel ile arka plan modelindeki ilgili piksele ait kod çizelgesindeki kod sözcüklerindeki doygunluk değeri arasındaki fark T_H doygunluk eşik değerinden küçük veya eşitse yeni gelen piksel gölgeye ait bir pikselidir. Aksi durumlarda ise piksel gölgeye ait piksel değildir.

α parametresi sahnedeki nesnelere yansıtma ve parlaklık özelliklerine ve ışık kaynağına bağlıdır. Yüksek yoğunluktaki ışık kaynakları ve yüksek yansıtıcı ve parlak nesnelere düşük α değerlerine sahiptirler. α genellikle 0.65 ve 0.80 değerleri arasında değişir (Bu değerler Cucchiara vd. (2002)'de yapılan deneysel çalışmalar sonucu elde edilmiştir). Dış ortamda çekilmiş video görüntülerinde gölge maskesi uygulanırken doğru sonuçların elde edilmesi için en düşük α değerinin seçilmesi gerekir (Cucchiara vd. 2002).

Diğer taraftan β parametresi ise gölgelerin hareketli nesnelere gibi bulunmasını önlemek için ve gürültüden kaynaklanan hatalı sınıflandırmaları önlemek için kullanılır. Cucchiara vd. 2002 de yapılan deneysel çalışmalar sonucuna göre β parametresi 0.90 ile 0.95 arasında değişir.

T_S (doygunluk) ve T_H (renk bileşenleri) değerleri doygunluk ve renk bileşenleri için izin verilen maksimum çeşitliliği göstermektedir. Gölge görüntüsünün doygunluğunda belirli bir değişikliğe yol açtığı daha önce belirtilmişti. Buna göre doygunluktaki değişiklik %15'i geçemez. Gölge renk değişiminde ise çok az değişikliğe neden olur. Rengin değişmesi için aradaki farkın 60'tan büyük olması gerekir (HSV renk uzayı modeline göre). 60 değeri 360 derecenin altı renge bölünmesiyle elde edilmiştir (Cucchiara vd. 2002).

5.3.3.2 Işık maskesi

Kod çizelgesi yöntemi arka plan modelini oluşturduktan sonra ön plan arka plan ayrımını gerçekleştirir. Ön plan arka plan ayrımı aşamasında ani ışık değişikliklerinde arka plana ait piksellerin değerleri de değişir. Dolayısıyla arka plan modelindeki kod sözcükleriyle arka plana ait pikseller eşleşmez. Gelen pikseller gerçek ön plan nesnesi olmamasına rağmen ön plan nesnesi olarak bulunurlar. Ani ışık değişiklikleri pikselin renkleri üzerinde çok az değişikliğe neden olurken daha çok parlaklıkları üzerinde değişikliğe sebep olurlar. Görüntü alınan yerin parlaklığı artarsa piksellerin de parlaklığı artar. Görüntü alınan yerin parlaklığı azalır ise piksellerin de parlaklığı azalır. Işık

maskesinin yapısı gölge maskesine benzemektedir. Cucchiara vd.'de (2001) verildiği gibi ışık maskesi olarak (5.9) numaralı eşitlik kullanılır:

$$IM_n(x, y) = \begin{cases} 1, & \text{eğer } \frac{1}{\beta} \leq \frac{I_n^V(x, y)}{B_n^V(x, y)} \leq \frac{1}{\alpha} \wedge (I_n^S(x, y) - B_n^S(x, y)) \leq T_S \wedge \\ & |I_n^H(x, y) - B_n^H(x, y)| \leq T_H \\ 0, & \text{diğer} \end{cases} \quad (5.9)$$

$I_n^H(x, y)$, $I_n^S(x, y)$, $I_n^V(x, y)$ gelen video çerçevesindeki n zamanında x , y koordinatındaki pikselin sırasıyla H, S ve V değerlerini göstermektedir. Aynı gösterim şekli B arka plana ait çerçeve için de kullanılmıştır. α , β , T_S , T_H değerleri dikkatlice seçilmesi gereken parametre değerleridir. (5.9) numaralı eşitliğe göre arka plan modelindeki ilgili piksele ait kod çizelgesindeki kod sözcüklerindeki parlaklık değeriyle yeni gelen pikselin parlaklık değerleri oranı $\frac{1}{\beta}$ 'dan büyük $\frac{1}{\alpha}$ 'dan küçük ise ve yeni gelen piksel ile arka plan modelindeki ilgili piksele ait kod çizelgesindeki kod sözcüklerindeki doygunluk değeri arasındaki fark T_S doygunluk eşik değerinden küçük veya eşitse ve yeni gelen piksel ile arka plan modelindeki ilgili piksele ait kod çizelgesindeki kod sözcüklerindeki doygunluk değeri arasındaki fark T_H doygunluk eşik değerinden küçük veya eşitse yeni gelen piksel gerçek ön plan pikseli değildir. Ani ışık değişikliği bu pikselin ön plan pikseli olarak bulunmasına neden olmuştur. Bu yüzden bu piksel ön plan maskesine dahil edilmez ve arka plan pikselidir. (5.9) numaralı eşitlik x , y koordinatındaki piksele ait bütün kod sözcükleri için işletilir. x , y koordinatındaki bir pikselin sahip olabileceği maksimum kod sözcüğü sayısı 6 ya da 7'dir (Kim vd. 2005).

5.3.4 Geliştirilen kutu tabanlı kod çizelgesi yönteminde bağlı bileşenler analizi

Ön plan arka plan ayırımı aşamasında, rüzgâr hızı arka plan modeli oluşturma aşamasındakinden fazla olursa sallanan ağaç yapraklarının daha hızlı sallanır ve arka plan modelinde olmayan arka plan pikselleri ön plan nesnesi olarak bulunur. Bunun dışında ön plan arka plan ayırımı aşamasında görüntüde bulunan gürültü miktarı arka plan modeli oluşturulma aşamasından fazla olursa gürültü de ön plan nesnesi olarak bulunabilir. Bu sorunları ortadan kaldırabilmek için algoritmaya bağlı bileşenler analizi yöntemi eklenmiştir. Bağlı bileşenler analizi yönteminde öncelikle ön plan arka plan ayırımı sonucu elde edilen ön plan maskesine morfolojik süzgeçlerden aşınma ve genişleme uygulanır (açma ve kapama da uygulanabilir). Morfolojik süzgeç işlemleri gerçekleştirildikten sonra görüntüdeki konturlar (aynı özneliğe sahip bir bölgeyi kuşatan ve sürekli bir eğri oluşturan noktalar kümesi) bulunur. Konturlar bulunurken siyah arka plan pikselleriyle beyaz ön plan piksellerinden faydalanılır. Aynı değere sahip komşu pikseller bir birine bağlıdır ve bu birbirine bağlı pikseller birleşerek konturu oluştururlar. Bulunan konturlardan büyüklüğü belirli bir eşik değerinin altında olanlar arka planda meydana gelen küçük değişikliklerden (rüzgar hızının artması, gürültünün artması) kaynaklanan nesnelere ve bu nesnelere ön plan maskesinden atılır. Konturlara uygulanacak eşik değeri görüntünün boyutlarına bakılarak elde edilir. Böylece eşik değeri görüntüye uygun seçilmiş bir değer olur. Bu yöntem sayesinde sallanan ağaç yaprakları ve gürültü ön plan görüntüsünden ayrılır.

5.3.4.1 Morfolojik işlemler

Morfolojik süzgeçler görüntü işlemede kullanılan şekillere dayalı bir süzgeç tekniğidir. Morfolojik süzgeç sonucu elde edilecek sonuç görüntüdeki her pikselin değeri, morfolojik süzgeç uygulanan girdi görüntüdeki piksellerin komşularıyla kıyaslanmasına dayanmaktadır. Komşuluğun büyüklüğü ve şekli seçilerek girdi görüntüdeki belirli şekillere duyarlı morfolojik süzgeç elde edilir. Morfolojik süzgeçler uygulanırken yapısal eleman girdi görüntü üzerinde piksel piksel dolaştırılır. Yapısal eleman görüntüden küçük matrislerdir. Genelde 3x3 lük 1 değerlerinden oluşan matrisler kullanılır. Morfolojik süzgeçler ikili görüntülerde ve gri ölçekli görüntülerde kullanılır.

Morfolojik süzgeçler görüntülerdeki nesnelerin genel şekillerini bozmaz. Morfolojik süzgeçlerin genişleme, aşınma, açma ve kapama gibi türleri vardır. Şekil 5.4'te bir görüntü ve morfolojik süzgeç uygulandıktan sonra görüntüde meydana gelen değişiklikler gösterilmiştir.

Genişleme: Sayısal bir görüntüye genişleme süzgecinin uygulanması demek görüntüyü yapısal elemanla kesiştiği bölümler kadar büyötmek demektir. Bunu yapabilmek için yapısal eleman görüntü üzerinde piksel piksel dolaştırılır. Eğer yapısal elemanın merkezi resim üzerinde "0" değerli bir piksel ile karşılaşırsa herhangi bir değişiklik meydana gelmez. Eğer değeri "1" olan bir piksel ile karşılaşırsa yapısal elemanla yapısal elemanın altında kalan pikseller mantıksal "or" işlemine tabi tutulurlar. Yani herhangi "1" değeriyle sonuç "1" e çevrilir.

Genişleme süzgecinin görüntüye uygulanmasıyla görüntü üzerindeki nesnelere büyöyür. Nesne içinde boşluklar varsa bunlar kapanır ve ayırık nesnelere birbirine yaklaşır ya da bağlanır.

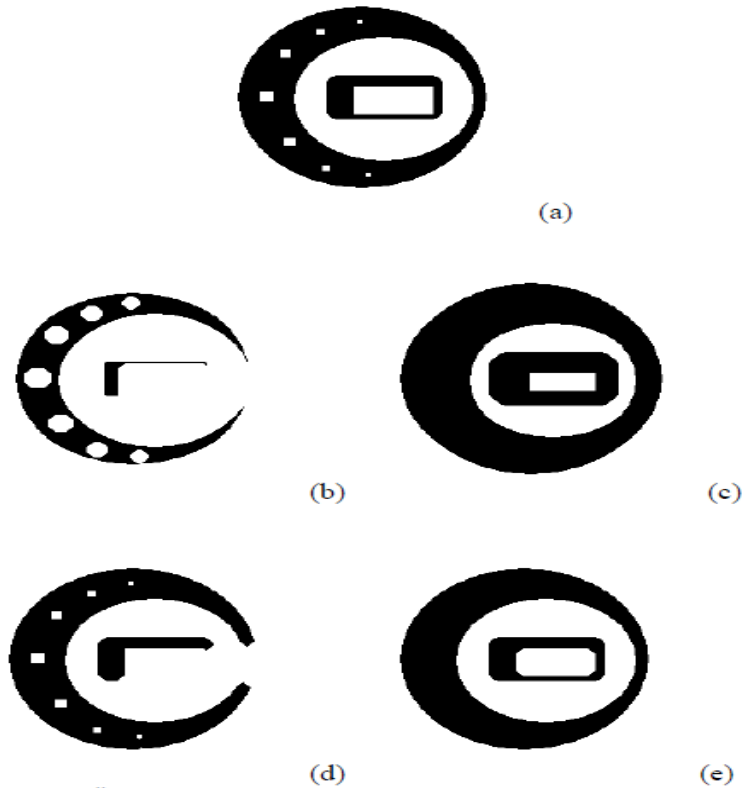
Aşınma: Aşınma işlemi bir bakıma genişlemenin tersi gibi görölebilir. Burada yine aynı şekilde yapısal eleman görüntü üzerinde piksel piksel dolaştırılır fakat bu defa yapısal elemanın merkez pikseli "1" değeri ile karşılaşırsa yapısal eleman içerisindeki piksellerin durumuna bakılır. Eğer yapısal eleman içerisindeki "1" olan piksellerden herhangi biri altında görüntüye ait "0" değeri varsa yapısal elemanın diğer "1"lerinin altındakilerle beraber bu piksel "0" a dönöştürölür.

Aşınma işlemi ile görüntü aşındırılmış olur. Yani görüntü içerisindeki nesnelere küçölür, boşluk varsa genişler, bağlı nesnelere ayrılma eğilimi gösterir.

Açma: Bir görüntüye önce aşınma daha sonra genişleme uygulanırsa görüntüye açma işlemi uygulanmış olur. Açma işlemi ince çıkıntılarını, dışarı doğru olan sivri sınır düzensizliklerini, ince birleşimlerini ve izole olmuş küçük nesnelere yok eder. Açma

işlemi ile birbirine yakın iki nesne görüntüde fazla değişime sebebiyet vermeden ayrılmış olurlar.

Kapama: Bir görüntüye önce genişleme daha sonra aşınma uygulanırsa kapama işlemi uygulanmış olur. Kapama işlemi ince girintileri, içe doğru olan sivri sınır düzensizliklerini ve küçük boşlukları yok eder. Kapatmada birbirine yakın iki nesne görüntüde fazla değişiklik yapılmadan birbirine bağlanmış olurlar.



Şekil 5.5 Morfolojik süzgeç örnekleri

a.Örnek görüntü, b. aşınma işlemi sonucu oluşan görüntü, c. genişleme işlemi sonucu oluşan görüntü, d. açılış işlemi sonucu oluşan görüntü, e. kapanış işlemi sonucu oluşan görüntü

Şekil 5.5'te orijinal bir görüntü ve bu görüntüye biçimsel süzgeçler uygulandıktan sonra elde edilen görüntüler verilmiştir.

6. DENEYSEL ÇALIŞMALAR

Yazılımlar geliştirilirken C, C++ programlama dilleri ve Intel şirketinin görsel bilgisayar uygulamaları için geliştirdiği “opencv” kütüphanesi kullanılmıştır. Gerçek zamanlı uygulamalarda hız önemli olduğu için, makine diline en yakın üst düzey programlama dilleri olması ve bunun sonucu olarak hızlı çalışmaları bu ortamların tercih nedenidir.

Yöntemler uygulanırken Intel Core2 CPU T7400 2.26GHz işlemci ve 1 GB belleğe sahip kişisel bir bilgisayar kullanılmıştır. Deneysel çalışmalar kısmında 3 farklı video için çerçeve farkı yöntemi, gauss fonksiyonları karışımı yöntemi, kutu tabanlı kod çizelgesi (KÇ) yöntemi ve geliştirilen kutu tabanlı kod çizelgesi yöntemleri kullanılarak ön plan arka plan ayrımı gerçekleştirilmiştir. Ön plan arka plan ayrımı süresince arka plan modelinin oluşturulması ve ön plan arka plan ayrımı esnasında geçen süreler hesaplanarak yöntemlerin hız bakımından kıyaslanması ve her yöntem ve her video için bulunan arka plan, ön plan maskesi ve ön plan görüntüsü verilmiştir.

Belirtilen yöntemler uygulanırken aşağıda verilen videolar kullanılmıştır. Bu videolar 2 farklı veri kümesinden alınmıştır. Videoların alındığı veri kümeleri aşağıda videoların özelliklerinin yanında belirtilmiştir.

Video1 : 384x288 10fps 2882 çerçeve (“Visor repository” veri kümesi)

Video2 : 320x240 10fps 887 çerçeve (“Visor repository” veri kümesi)

Video3 : 160x120 10fps 287 çerçeve (“Wallflower” veri kümesi)

Çizelge 6.1 Arka plan modeli oluşturma aşamasında yöntemlerin videolara göre hızları

Arka plan modeli oluşturma aşaması	Video1	Video2	Video3
Kutu Tabanlı KÇ (RGB)	30 milisaniye/çerçeve	25 milisaniye/çerçeve	15 milisaniye/çerçeve
Kutu Tabanlı KÇ (HSV)	40 milisaniye/çerçeve	30 milisaniye/çerçeve	25 milisaniye/çerçeve
Geliştirilen Kutu Tabanlı KÇ	40 milisaniye/çerçeve	30 milisaniye/çerçeve	25 milisaniye/çerçeve
Gauss Karışımları Yöntemi	120 milisaniye/çerçeve	100 milisaniye/çerçeve	80 milisaniye/çerçeve
Çerçeve Farkı	-	-	-

Çizelge 6.2 Ön plan arka plan ayırma aşamasında yöntemlerin videolara göre hızları

Ön plan arka plan ayırma aşamasında	Video1	Video2	Video3
Kutu Tabanlı KÇ (RGB)	34 milisaniye/çerçeve	29 milisaniye/çerçeve	22 milisaniye/çerçeve
Kutu Tabanlı KÇ (HSV)	62 milisaniye/çerçeve	43 milisaniye/çerçeve	34 milisaniye/çerçeve
Geliştirilen Kutu Tabanlı KÇ	76 milisaniye/çerçeve	52 milisaniye/çerçeve	34 milisaniye/çerçeve
Gauss Karışımları Yöntemi	105 milisaniye/çerçeve	80 milisaniye/çerçeve	65 milisaniye/çerçeve
Çerçeve Farkı Yöntemi	17 milisaniye/çerçeve	15 milisaniye/çerçeve	10 milisaniye/çerçeve

Çizelge 6.1 ve 6.2’de yöntemlerin hız bakımından kıyaslanması verilmiştir. Her yöntemin bir çerçeveyi kaç milisaniyede işlediği gösterilmiştir. Çizelge 6.1’de arka plan modeli oluşturma aşamasında elde edilen sonuçlar bulunmaktadır. Kullanılan yöntemlerden çerçeve farkı yöntemi arka plan modeli oluşturmadığı için arka plan modeli oluşturma aşamasında hızı bulunmamaktadır. RGB renk uzayındaki kutu tabanlı kod çizelgesi yöntemiyle HSV renk uzayındaki kutu tabanlı kod çizelgesi yöntemi arasında bulunan süre farkı renk uzayı dönüşümünden kaynaklanmaktadır. Geliştirilen kutu tabanlı kod çizelgesiyle HSV renk uzayındaki kutu tabanlı kod çizelgesi yöntemi

arasında süre farkı bulunmamaktadır. Çünkü kutu tabanlı kod çizelgesi yöntemine yapılan eklemeler ön plan arka plan ayrımı aşamasında bulunmaktadır. Çizelge 6.2’de ise ön plan arka plan ayrımı aşamasında elde edilen süreler bulunmaktadır. Bu aşamada kutu tabanlı kod çizelgesi yapılan eklemelerden dolayı geliştirilen yöntemin işlem süresi kutu tabanlı kod çizelgesi yönteminden fazladır. Çizelge 6.1 ve 6.2’de görüldüğü üzere çerçeve farkı yöntemi en kısa sürelerle sahiptir. Çünkü en az işlem sayısına sahiptir (basit bir çıkarma işlemi ve eşik değeri uygulanmasından ibarettir).

Yöntemlerin hız karşılaştırmasını verdikten sonra elde edilen ön plan ve arka plan görüntüleri verilebilir. Video1 1010. çerçeve için çerçeve farkı yöntemi, Gauss fonksiyonları karışımı yöntemi, kutu tabanlı kod çizelgesi yöntemi ve geliştirilen kutu tabanlı kod çizelgesi yöntemiyle elde edilen sonuçlar sırasıyla şekil 6.1-6.4’te verilmiştir.



Şekil 6.1 Video1 için çerçeve farkı yöntemiyle elde edilen sonuçlar
Sol üst orijinal görüntü, Sağ üst arka plan görüntüsü, Sol alt ön plan maskesi, Sağ alt ön plan görüntüsü



Şekil 6.2 Video1 için Gauss karışımları yöntemiyle elde edilen sonuçlar
Sol üst orijinal görüntü, Sağ üst arka plan görüntüsü, Sol alt ön plan maskesi, Sağ alt ön plan görüntüsü



Şekil 6.3 Video1 için kutu tabanlı kod çizelgesi yöntemiyle elde edilen sonuçlar
Sol üst orijinal görüntü, Sağ üst arka plan görüntüsü, Sol alt ön plan maskesi, Sağ alt ön plan görüntüsü



Şekil 6.4 Video1 için geliştirilen kutu tabanlı kod çizelgesi yöntemiyle elde edilen sonuçlar
Sol üst orijinal görüntü, Sağ üst arka plan görüntüsü, Sol alt ön plan maskesi, Sağ alt ön plan görüntüsü

Video1 güvenlik kamerasından alınan bir videodur. Bu videoda hareket eden insanlar ve sabit bir arka plan bulunmaktadır. Şekil 6.1-6.4 video1 için sırasıyla çerçeve farkı yöntemiyle, Gauss fonksiyonları karışımı yöntemiyle, kutu tabanlı kod çizelgesi yöntemiyle ve geliştirilen kutu tabanlı kod çizelgesi yöntemiyle 1010. çerçeve için elde edilen sonuçları göstermektedir. Şekil 6.1’de sonuçları verilen çerçeve farkı yönteminde arka plan görüntüsünde ön plan nesnelere de yer almaktadır (bir önceki çerçeveyi arka plan olarak kabul ettiği için). Ön plan maskesinde ve ön planda da görüldüğü üzere hareketli nesnelerin kenarlarını içermektedir. Şekil 6.2’de sonuçları verilen Gauss karışımları yöntemi hareketli nesnelerin gölgelerini de ön plan nesnesi olarak bulmuştur. Arka plan görüntüsünde ise hareketli nesnelerin izlerini yok edememektedir. Şekil 6.3’te sonuçları verilen kutu tabanlı kod çizelgesi yönteminde hareketli nesnelerin gölgeleri ve gürültü hatalı ön plan nesnelere olarak bulunmuştur. Kutu tabanlı kod çizelgesi yöntemi arka planı bulmada başarılıdır. Şekil 6.4’te sonuçları

verilen kutu tabanlı kod çizelgesi yöntemi hareketli nesnelerin gölgelerini ve gürültüyü ön plan nesnesinden ayırmıştır ve arka planı bulmada da başarılıdır.

Video2 380. çerçeve için çerçeve farkı yöntemi, Gauss fonksiyonları karışımı yöntemi, kutu tabanlı kod çizelgesi yöntemi ve geliştirilen kutu tabanlı kod çizelgesi yöntemiyle elde edilen sonuçlar sırasıyla şekil 6.5-6.8’de verilmiştir.



Şekil 6.5 Video2 için çerçeve farkı yöntemiyle elde edilen sonuçlar
Sol üst orijinal görüntü, Sağ üst arka plan görüntüsü, Sol alt ön plan maskesi, Sağ alt ön plan görüntüsü



Şekil 6.6 Video2 için Gauss karışımları yöntemiyle elde edilen sonuçlar
Sol üst orijinal görüntü, Sağ üst arka plan görüntüsü, Sol alt ön plan maskesi, Sağ alt ön plan görüntüsü



Şekil 6.7 Video2 için kutu tabanlı kod çizelgesi yöntemiyle elde edilen sonuçlar
Sol üst orijinal görüntü, Sağ üst arka plan görüntüsü, Sol alt ön plan maskesi, Sağ alt ön plan görüntüsü



Şekil 6.8 Video2 için geliştirilen kutu tabanlı kod çizelgesi yöntemiyle elde edilen sonuçlar

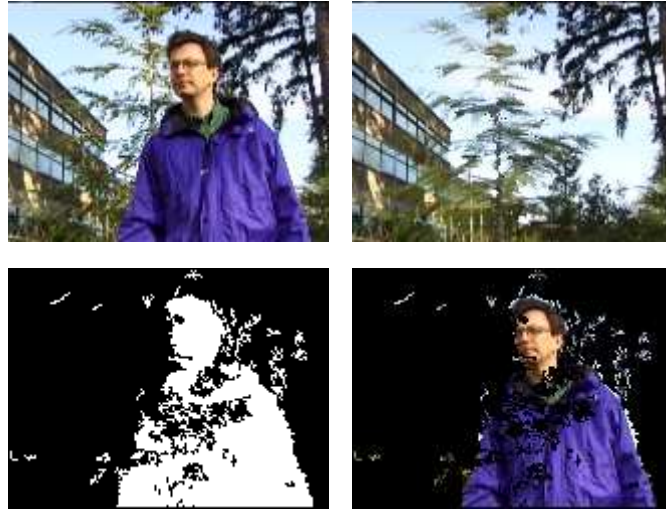
Sol üst orijinal görüntü, Sağ üst arka plan görüntüsü, Sol alt ön plan maskesi, Sağ alt ön plan görüntüsü

Video2 de bir yerleşke giriş kapısının videosudur. Yerleşke giriş kapısında bulunan bariyer ilk başlarda açıkken ilerleyen çerçevelerde kapanmaktadır. Şekil 6.5-6.8 video2 için sırasıyla çerçeve farkı yöntemiyle, Gauss fonksiyonları karışımı yöntemiyle, kutu tabanlı kod çizelgesi yöntemiyle ve geliştirilen kutu tabanlı kod çizelgesi yöntemiyle 380. çerçeve için elde edilen sonuçları göstermektedir. Şekil 6.5'te sonuçları verilen çerçeve farkı yönteminde yine ön plan nesnelere arka plan olarak bulunmuştur. Şekil 6.6'da sonuçları verilen gauss fonksiyonları karışımı yönteminde hareketli nesnelere gölgeleri de ön plan nesnesi olarak bulunmuştur. Bariyer ise ne ön planda ne de arka planda bulunmuştur. Şekil 6.7'de sonuçları verilen kutu tabanlı kod çizelgesi yönteminde hareketli nesnelere gölgeleri ön plan nesnesi olarak bulunmuştur. Şekil 6.4'te sonuçları verilen kutu tabanlı kod çizelgesi yöntemi hareketli nesnelere gölgelerini ve gürültüyü ön plan görüntüsünden ayırmıştır.

Video3 247. çerçeve farkı yöntemi, Gauss fonksiyonları karışımı yöntemi, kutu tabanlı kod çizelgesi yöntemi ve geliştirilen kutu tabanlı kod çizelgesi yöntemiyle elde edilen sonuçlar sırasıyla şekil 6.9-6.12’de verilmiştir.



Şekil 6.9 Video3 için çerçeve farkı yöntemiyle elde edilen sonuçlar
Sol üst orijinal görüntü, Sağ üst arka plan görüntüsü, Sol alt ön plan maskesi, Sağ alt ön plan görüntüsü



Şekil 6.10 Video3 için Gauss karışımları yöntemiyle elde edilen sonuçlar
Sol üst orijinal görüntü, Sağ üst arka plan görüntüsü, Sol alt ön plan maskesi, Sağ alt ön plan görüntüsü



Şekil 6.11 Video3 için kutu tabanlı kod çizelgesi yöntemiyle elde edilen sonuçlar
Sol üst orijinal görüntü, Sağ üst arka plan görüntüsü, Sol alt ön plan maskesi, Sağ alt ön plan görüntüsü

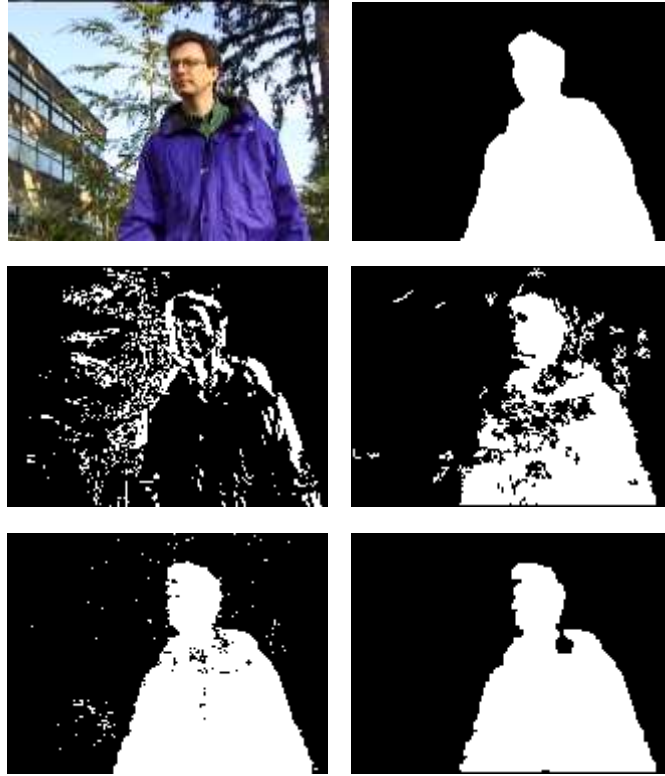


Şekil 6.12 Video3 için geliştirilen kutu tabanlı kod çizelgesi yöntemiyle elde edilen sonuçlar
Sol üst orijinal görüntü, Sağ üst arka plan görüntüsü, Sol alt ön plan maskesi, Sağ alt ön plan görüntüsü

Video3 hareketli arka plan nesnelere (sallanan ağaç yaprakları) içeren bir videodur. Videoda ilk önce sadece sallanan ağaç yaprakları bulunmaktadır. Daha sonra bir kişi videoda görünmektedir. Şekil 6.9-6.12 video3 için sırasıyla çerçeve farkı yöntemiyle, Gauss fonksiyonları karışımı yöntemiyle, kutu tabanlı kod çizelgesi yöntemiyle ve

geliştirilen kutu tabanlı kod çizelgesi yöntemiyle 247. çerçeve için elde edilen sonuçları göstermektedir.

“Waterfall” veri kümesinde video 3’ün 247’nci çerçevesi için elle bölütlenmiş ön plan maskesi bulunmaktadır. Bu maskeyle diğer yöntemlerden elde edilen ön plan maskesi Şekil 6.13’te verilmiştir. Şekil 6.13’teki sonuçlara göre geliştirilen kutu tabanlı kod çizelgesi yöntemiyle elle bölütlenmiş ön plan maskesine yakın sonuç elde edilmiştir.



Şekil 6.13 Elle bölütlenmiş ön plan maskesiyle diğer ön plan maskeleri

Sol üst orijinal görüntü, sağ üst elle bölütlenmiş görüntü, orta sol çerçeve farkı yöntemi, orta sağ Gauss fonksiyonları karışımı, sol alt kutu tabanlı kod çizelgesi, sağ alt geliştirilen kutu tabanlı kod çizelgesi

7. SONUÇLAR

Videolarda ön plan ve arka plan ayrılmasının önemi görsel bilgisayar uygulamalarında gün geçtikçe artmaktadır. Yapılan uygulamaya göre değişen çeşitli algoritmalar vardır. Bu algoritmalarından bazılarının bellek kullanımı iyi olmasına karşın algoritma karmaşıklığı yüksektir. Bazılarının da algoritma karmaşıklığı düşük fakat bellek kullanımı yüksektir. Algoritmalarından bazıları karmaşık arka planlar üzerinde iyi sonuç üretirken bazıları üretemez. Değişen arka plana adaptasyonu uzun süren algoritmalar olduğu gibi kısa süren algoritmalar da vardır. Gerçek zamanlı uygulamalarda kullanılacak olan algoritmanın hızlı ve doğru sonuç üretmesine dikkat edilir. Yapılan uygulamanın gereksinimlerine göre algoritma seçilmelidir.

Bu yüksek lisans tez çalışmasında videolarda ön plan arka plan ayrımı konusu incelenmiştir. Videolarda ön plan arka plan ayrımında izlenmesi gereken yollardan bahsedilmiş ve ön plan arka plan ayrımında kullanılan yöntemlerden bazıları anlatılmıştır. Yöntemler anlatılırken gerçek zamanlı uygulamalar için uygun olup olmamaları, bellek kullanımları ve doğru sonuç üretmeleri üzerinde durulmuştur. Gerçek zamana uyumluluk, etkin bellek kullanımı ve doğru sonuç üretilmesi kapsamında kod çizelgesi yöntemi anlatılmıştır. Kod çizelgesi yönteminin daha hızlı çalışmasını sağlayan kutu tabanlı kod çizelgesi yöntemi anlatılmıştır. Kutu tabanlı kod çizelgesi yönteminin avantajlarını kullanarak daha iyi sonuçlar elde etmemizi sağlayan geliştirilen kutu tabanlı kod çizelgesi yöntemi anlatılmıştır. Kutu tabanlı kod çizelgesi yöntemine bağlı bileşenler analizi yöntemi, ışık maskesi ve gölge maskesi eklenerek bu yöntem geliştirilmiştir. Geliştirilen kutu tabanlı kod çizelgesi yöntemi ve bazı yöntemlerden elde edilen sonuçların karşılaştırılması deneysel çalışmalar kısmında verilmiştir.

Deneysel çalışmalar kısmında ön plan arka plan ayrımında kullanılan yöntemlerden çerçeve farkı yöntemi, Gauss karışımları yöntemi, kutu tabanlı kod çizelgesi yöntemi ve geliştirilen kutu tabanlı kod çizelgesi yöntemi üç farklı video için uygulanmıştır. Her bir yöntemin uygulanması sonucu elde edilen arka plan görüntüsü, ön plan maskesi ve ön plan görüntüsü her bir video için verilmiştir. Elde edilen sonuçlar kendi aralarında

karşılaştırılmıştır. Son olarak elle bölütlenmiş ön plan maskesiyle uygulanan yöntemler sonucu elde edilen ön plan maskeleri karşılaştırılmıştır. Karşılaştırmalar sonucunda geliştirilen kutu tabanlı kod çizelgesi yönteminin başarılı olduğu görülmüştür.

KAYNAKLAR

- Anonim. 2010. Web Sitesi:<http://www.goruntuisleme.org/>, Erişim Tarihi:28/05/2010
- Anonymus. 1995. Web Sitesi:
http://files.acdsystems.com/english/support/canvas/canvas-software-downloads/pdf-tutorials/color_channels.pdf, Erişim Tarihi:10/10/2009
- Anonymus. 2008. Web Sitesi:http://bersoft.com/bimagem/help/color_channels.htm
Erişim Tarihi:10/10/2009
- Anonymus. Web Sitesi: <http://www.wikipedia.org>, Erişim Tarihi:10/10/2009
- Benton, S. 2008. Background Subtraction Web Sitesi: <http://www.dspdesignline.com>
- Bobick, A. and Davis, J. 2001. The Recognition of Human Movements Using Temporal Templates, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 23, No.3
- Bradski, G. and Kahler A. 2008. O'Reilly Learning OpenCV Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.
- Cheung, S. S. and Kamath, C. 2004. Robust techniques for background subtraction in urban traffic video, Visual Communications and Image Processing 2004, Vol. 5308, No. 1., pp. 881-892.
- Collins, R.T., Lipton, A. J., Fujiyoshi, H., and Kanade, T. 2001. Algorithms for Cooperative Multi sensor Surveillance, Proceeding of IEEE, Vol. 89. No.10
- Cucchiara, R., Grana, C., Neri, G., Piccardi, M. and Prati, A. 2002. "The Sakbot System for Moving Object Detection and Tracking," 10th ACM International Conference on Multimedia, ACM Press, France, pp. 223-226
- Cucchiara, R., Grana, C., Piccardi, M., Prati, A. and Sirotti, S. 2001. Improving shadow suppression in moving object detection with hsv color information, Proc. IEEE Int'l Conf. Intelligent Transportation Systems, pages 334-339
- Cucchiara, R., Piccardi, M. and Prati, A. 2003. Detecting moving objects, ghosts, and shadows in video streams," IEEE Transactions on Pattern Analysis and Machine Intelligence 25, pp. 1337-1342
- Cutler R., and Davis, L. 1998. View-based detection, in Proceedings Fourteenth International Conference on Pattern Recognition, 1, pp. 495-500, (Brisbane, Australia)
- Cücioğlu İ. 2003. Analog video temelleri, Web Sitesi: www.eproje.com
- Deane, S. 2007. A Comparison of Background Subtraction Techniques COMP6009 Individual Research Project
- Doshi, A. and Trivedi M. M. 2006. "Hybrid Cone-Cylinder" Codebook Model for Foreground Detection with Shadow and Highlight Suppression. AVSS 2006: 19
- Ekinci, M. ve Gedikli, E. 2005. Turk J Elec Engin, VOL.13, NO.2 TUBITAK Silhouette Based Human Motion Detection and Analysis for Real-Time Automated Video Surveillance
- Elgammal A, Harwood D, Davis LS. 2000 Non-parametric model for background subtraction. European Conference on Computer Vision 2:751-67.
- Elgammal, A., Harwood, D. and Davis L. 1999. Non-parametric model for background subtraction, in Proceedings of IEEE ICCV'99 Frame-rate workshop
- Ertürk, S. 2009. Digital Image Processing Ders Notları, Kocaeli Üniversitesi

- Fuentes, L. and Velastin, S. 2003. From tracking to advanced surveillance, in Proceedings of IEEE International Conference on Image Processing, (Barcelona, Spain)
- Gonzales, R.C. and Woods, R.E. 2002 "Digital image processing", Prentice Hall
- Goutsias, J. and Batman, S. 2000. Morphological methods for biomedical image analysis. Handbook of Medical Imaging, vol.2, Medical Image Processing and Analysis, M. Sonka, J.M. Fitzpatrick (ed.s), SPIE Press, Bellingham, Washington, 175-272.
- Han, H., Wang, Z., Liu, J., Li, Z., Li, B. and Z. Han. 2003. Adaptive background modeling with shadow suppression. In Intelligent Transportation Systems, 2003. Proceedings. 2003 IEEE, volume 1, pages 720–724
- Haritaoglu, I., Harwood, D. and Davis, L. 2000. Real-time surveillance of people and their activities," IEEE Transactions on Pattern Analysis and Machine Intelligence 22, pp. 809-830
- Heikkila J. and Silven O. 1999. A real-time system for monitoring of cyclists and pedestrians, in Second IEEE Workshop on Visual Surveillance, pp. 246-252, (Fort Collins, Colorado)
- Horprasert, T., Harwood, D. and Davis, L. S. 1999. A statistical approach for realtime robust background subtraction and shadow detection. In Proceedings of IEEE Frame Rate Workshop, pages 1–19
- Jack K. 1995. "Video Demystified", LLH Technology Publishing
- Javed, O. and Shah, M. 2002. Tracking and Object Classification for Automated Surveillance, ECCV'2002, European Conference on Computer Vision, Copenhagen, Denmark
- KaewTraKulPong, P. and Bowden, R. 2001. An improved adaptive background mixture model for real-time tracking with shadow detection," in Proceedings of the 2nd European Workshop on Advanced Video-Based Surveillance Systems
- Kang, J., Cohen, I. and Medioni, G. 2003. Continuous tracking within and across camera streams," in Proceedings IEEE conference on computer vision and pattern recognition, 1, pp. 267-272, (Madison, WI)
- Karman, K. P. and Brandt, A. 1990. Moving object recognition using and adaptive background memory," in Time-Varying Image Processing and Moving Object Recognition, V. Cappellini, ed., 2, pp. 289-307, Elsevier Science Publishers B.V.
- Kim, K., Chalidabhongse, T. H., Harwood D. and Davis L. 2005. Realtime foreground-background segmentation using codebook model Real-time Imaging, Volume 11, Issue 3, Pages 167-256
- Kohonen T. 1988. Learning vector quantization. Neural Networks ;1:3–16.
- Lee, D. S., Hull, J. and Erol B. 2003. A Bayesian framework for gaussian mixture background modeling, in Proceedings of IEEE International Conference on Image Processing, (Barcelona, Spain)
- Magee, D. R. 2002. Tracking multiple vehicles using foreground, background, and motion models," in Proceedings of the Statistical Methods in Video Processing Workshop, pp. 7-12, (Copenhagen, Denmark)
- Matheron, G. 1975. Random Sets and Integral Geometry. Wiley, 261, New York
- McIvor, A.M., 2000. Background subtraction techniques. Proc. Image Vision Comput., 1: 155-163.

- Meier T. and Ngan K. N. 1999. Video segmentation for content based coding, IEEE Trans CSVT Vol 9, No: 8, 1190-102
- Mirza, H., Thai, H. and Nakao, Z. 2008. Digital Video Watermarking Based on RGB Color Channels and Principal Component, Analysis Knowledge-Based Intelligent Information and Engineering Systems, Volume 5178/2010
- Mittal, A. and Huttenlocher, D. 2000. Scene modeling for wide area surveillance and image synthesis, in Proceedings IEEE conference on computer vision and pattern recognition, 2, pp. 160-167, Hilton Head Island, SC
- Oliver, N., Rosario, B. and Pentland, A. 2000. A Bayesian computer vision system for modeling human interactions, IEEE Transactions on Pattern Analysis and Machine Intelligence 22, pp. 831-843
- Perez, P., Hue, C., Vermaak, J. and Gangnet, M. 2002 Color-Based Probabilistic Tracking, Proc. Of European Conference on Computer Vision, Copenhagen
- Power, P.W. and Schoonees, J.A. 2002. Understanding background mixture models for foreground segmentation, in Proceedings Image and Vision Computing New Zealand, pp. 267-271, (Auckland, New Zealand)
- Prati, A., Mikic, I., Trivedi, M. and Cucchiara, R. 2001. Detecting Moving Shadows: Formulation, Algorithms and Evaluation – Survey
- Remagnino, P., Baumberg, A., Grove, T., Hogg, D., Tan, T., Worrall, A. and Baker, K. 1997. An integrated traffic and pedestrian model based vision system. In British Machine Vision Conference, volume 2, pages 380–389
- Ridder, C., Munkelt, O. and Kirchner, H. 1995. Adaptive Background Estimation and Foreground Detection using Kalman-Filtering
- Ripley BD. 1996. Pattern recognition and neural networks. Cambridge University Press
- Selen A. U. 2008. Web Sitesi: <http://www.ceviktavsan.com/piksel-cozunurluk-vederinlik>, Erişim Tarihi: 10/10/2009
- Serra, J. 1982. Image Analysis And Mathematical Morphology. Academic Press, 610, London
- Shireen Y. E., Khaled M. E. and Sumaya H. A. 2007. Moving Object Detection in Spatial Domain using Background Removal Techniques, Recent Patents on Computer Science
- Stauffer C, Grimson WEL. 1999. Adaptive background mixture models for real-time tracking. IEEE International Conference on Computer Vision and Pattern Recognition, 2:246–52
- Sun, J., Zhang, W., Tang, X. and Shum, H. Y. 2006. Background Cut, in Microsoft Research
- Toyama, K., Krumm, J., Brumitt, B. and Meyers, B. 1999. Wallflower: Principles and practice of background maintenance," in ICCV (1), pp. 255-261
- Toyama, K., Krumm, J., Brumitt, B. and Meyers, B. Wallflower: Principles and Practice of Background Maintenance, Seventh International Conference on Computer Vision, Kerkyra, Greece, pp. 255-261, IEEE Computer Society Press
- Tu, Q., Xu, Y. and Zhou, M. 2008. Box-based Codebook Model for Real-time Objects Detection, Proceedings of the 7th World Congress on Intelligent Control and Automation, Chongqing, China
- ViSOR repository. 2007. Web Sitesi: <http://www.openvisor.org> Erişim Tarihi: 03/02/2009

- Wren, C., Azabajejani, A., Darrel, T. and Pentland, A. 1997. Real-time tracking of the human body, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 780-785
- Yılmaz İ., 2002, “Renk Sistemleri, renk uzayları ve dönüşümler”, Selçuk Üniversitesi Jeodezi ve Fotogrametri Mühendisliği Öğretiminde 30. Yıl Sempozyumu, Konya
- Zaibi R. 1999. Change Detection in Digital Signals, M. Sc Thesis, Institute of Engineering and Sciences, Bilkent University

ÖZGEÇMİŞ

Adı Soyadı : Ahmet BUĞDAY
Doğum Yeri : Elazığ
Doğum Tarihi : 01.01.1984
Medeni Hali : Evli
Yabancı Dili : İngilizce

Eğitim Durumu (Kurum ve Yıl)

Lise : Elazığ Kaya Karakaya Fen Lisesi (1999-2002)
Lisans : Hacettepe Üniversitesi Mühendislik Fakültesi Bilgisayar
Mühendisliği Bölümü (2002-2006)
Yüksek Lisans: Ankara Üniversitesi Fen Bilimleri Enstitüsü Elektronik
Mühendisliği Anabilim Dalı (Şubat 2007- Haziran 2010)

Çalıştığı Kurum/Kurumlar ve Yıl

2007-2008: Meteksan Sistem A.Ş.
2008----- : TÜİK(Türkiye İstatistik Kurumu)