

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**MARKOV AĞLARI KULLANILARAK YÜKSEK
ÇÖZÜNÜRLÜKLÜ GÖRÜNTÜ ÜRETİLMESİ**

**YÜKSEK LİSANS TEZİ
Müh. Bora TUNÇER**

Anabilim Dalı: ELEKTRONİK ve HABERLEŞME MÜHENDİSLİĞİ

Programı: TELEKOMÜNİKASYON MÜHENDİSLİĞİ

EKİM 2007

**MARKOV AĞLARI KULLANILARAK YÜKSEK
ÇÖZÜNÜRLÜKLÜ GÖRÜNTÜ ÜRETİLMESİ**

**YÜKSEK LİSANS TEZİ
Müh. Bora TUNÇER
504041305**

**Tezin Enstitüye Verildiği Tarih : 21 Kasım 2007
Tezin Savunulduğu Tarih : 22 Ekim 2007**

**Tez Danışmanı : Prof.Dr. Bilge GÜNSEL
Diğer Jüri Üyeleri Prof. Dr. Melih PAZARCI
Prof.Dr. Muhittin GÖKMEN**

EKİM 2007

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**MARKOV AĞLARI KULLANILARAK YÜKSEK
ÇÖZÜNÜRLÜKLÜ GÖRÜNTÜ ÜRETİLMESİ**

**Anabilim Dalı: ELEKTRONİK ve HABERLEŞME MÜHENDİSLİĞİ
Programı: TELEKOMÜNİKASYON MÜHENDİSLİĞİ**

**YÜKSEK LİSANS TEZİ
Müh. Bora TUNÇER
504041305**

Tez Danışmanı: Prof.Dr. Bilge GÜNSEL

EYLÜL 2007

İÇİNDEKİLER

KISALTMALAR	iv
TABLO LİSTESİ	v
ŞEKİL LİSTESİ	vi
SEMBOL LİSTESİ	viii
ÖZET	ix
SUMMARY	xii
1. GİRİŞ	1
2. ÖN İŞLEME ADIMLARI	4
2.1 Giriş Görüntüsünün Farklı Frekans Bantlarına Ayrılması	5
2.1.1 Örnekleme Sıklığının Değiştirilmesi Yöntemi	6
2.1.2 Filtreleme Yöntemi	9
2.2 Görüntülerin Parçalara Bölünmesi	10
3. MARKOV AĞ TOPOLOJİSİ İLE YÜKSEK-DÜŞÜK ÇÖZÜNÜRLÜK İLİŞKİSİNİN ÖĞRENİLMESİ	13
4. YÜKSEK ÇÖZÜNÜRLÜKLÜ GÖRÜNTÜ ÜRETİLMESİ	18
4.1 Ön Adımlar	18
4.1.1 Giriş Görüntüsünün Farklı Frekans Bantlarına Ayrılması	19
4.1.2 Orta Frekans Görüntüsünün Parçalara Bölünmesi	19
4.2 Yüksek Frekans Görüntüsünün Bulunması	19
4.2.1 Markov Ağları Kullanılarak Yüksek Frekans Görüntüsünün Bulunması	20
4.2.1.1 Gizli Düğümler ve Gözlem Düğümleri	21
4.2.1.2 Geçiş Fonksiyonları	22
4.2.2 Tek Geçiş Algoritması	24
4.3 Yüksek Çözünürlüklü Video Üretmek	25
5. TEST SONUÇLARI	30
5.1 Veritabanında Kullanılan Görüntüler	30
5.2 Sonuçlar	32
5.2.1 Markov Ağları Kullanılarak Bulunan Yüksek Çözünürlüklü Görüntüler	33
5.2.2 Tek Geçiş Algoritması Kullanılarak Bulunan Yüksek Çözünürlüklü Görüntüler	35
5.3 Markov Ağları Kullanılarak Kestirilen Yüksek Çözünürlüklü Videolar	37

5.4	Alfa Parametresinin Başarıma Etkisi	42
5.5	Yüksek-Düşük Frekans Eşleme Olasılıkları	44
5.6	Örnekleme Sıklığının Değiştirilmesinin Yüksek Çözünürlüklü Görüntülerin Kestirimine Etkisi	46
6.	SONUÇLAR VE TARTIŞMA	49
	KAYNAKLAR	50
	ÖZGEÇMİŞ	52

KISALTMALAR

NTSC	: National Television Standards Committee
PAL	: Phase Alternating Line
SD	: Standard Definition
HD	: High Definition
HDTV	: High Definition Television
MMSE	: Minimum Mean Squared Error
MAP	: Maximum A Posteriori
PNL	: Probabilistic Network Library
PSNR	: Picture Signal to Noise Ratio
PNL	: Probabilistic Network Library

TABLO LİSTESİ

	<u>Sayfa No</u>
Tablo 2.1 Yüksek geçiren filtre	9
Tablo 5.1 Değişken sigma değerleri için PSNR sonuçları	43
Tablo 5.2 Düşük çözünürlüklü giriş görüntüsü için veri tabanından seçilen aday parçaların ve sonuç olarak kestirilen parçaların olasılıkları.	45
Tablo 5.3 Veri tabanından seçilen aday sayısına göre parçaların olasılıkları	46

ŞEKİL LİSTESİ

	<u>Sayfa No</u>
Şekil 2.1 : Yüksek çözünürlüklü giriş görüntüsünün değişik frekans bandlarındaki karşılığı	5
Şekil 2.2 : Örnekleme sıklığı değiştirilerek giriş görüntüsünün farklı frekanslara ayrılması	6
Şekil 2.3 : Örnekleme sıklığının değiştirilmesinin etkisi	7
Şekil 2.4 : Normalizasyon işlemi için kullanılan görüntüler ve normalizasyon sonucunda oluşan görüntüler.....	8
Şekil 2.5 : Filtreleme yöntemi kullanılarak giriş görüntüsünün farklı frekanslara ayrılması	9
Şekil 2.6 : Yüksek ve orta frekans parçalarının seçimi	11
Şekil 2.7 : Komşuluk ilişkisinin etkisi	11
Şekil 3.1 : Döngü içeren Markov Ağ Modeli	14
Şekil 3.2 : Döngü içermeyen Markov Ağ Modeli	15
Şekil 4.1 : Komşuluk ilişkisinin etkisi	20
Şekil 4.2 : Markov Ağ Modeli	21
Şekil 4.3 : Veri tabanından seçilen aday parçaların Markov Ağları'na uyarlanması	23
Şekil 4.4 : Tek Geçiş Algoritması	24
Şekil 4.5 : Kayan pencere yöntemi ile yüksek çözünürlüklü videoların oluşturulması.....	26
Şekil 4.6 : Yüksek çözünürlüklü bir video klibinin kestirimi için kullanılan yöntemler	28
Şekil 5.1 : Genel veri tabanını oluşturan görüntüler	31
Şekil 5.2 : Bisiklet veri tabanında bulunan görüntüler	31
Şekil 5.3 : Yüz veri tabanını oluşturan görüntüler	32
Şekil 5.4 : Yüz veri tabanı kullanılarak kestirilen görüntüler	32
Şekil 5.5 : Yüksek çözünürlüklü test görüntüleri	33
Şekil 5.6 : Markov Ağları kullanılarak bulunan sonuçlar – 1	34
Şekil 5.7 : Markov Ağları kullanılarak bulunan sonuçlar – 2	34
Şekil 5.8 : Markov Ağları kullanılarak bulunan sonuçlar – 3	34
Şekil 5.9 : Markov Ağları kullanılarak bulunan sonuçlar – 4	35
Şekil 5.10 : Tek Geçiş Algoritması ile bulunan sonuçlar – 1	36
Şekil 5.11 : Tek Geçiş Algoritması ile bulunan sonuçlar – 2	36
Şekil 5.12 : Tek Geçiş Algoritması ile bulunan sonuçlar – 3	36
Şekil 5.13 : Yüksek çözünürlüklü bir videodan bazı çerçeveler ve bunların yüksek çözünürlüklü karşılıkları	37

Şekil 5.14 :Düşük çözünürlüklü videodan alınan bazı çerçeveler için kestirilen çerçveler	38
Şekil 5.15 : Yükek çözünürlüklü bir videodan bazı çerçeveler ve bunların yüksek çözünürlüklü karşılıkları	39
Şekil 5.16 :Düşük çözünürlüklü videodan alınan bazı çerçeveler için kestirilen çerçveler	40
Şekil 5.17 : Yükek çözünürlüklü bir videodan bazı çerçeveler ve bunların yüksek çözünürlüklü karşılıkları	41
Şekil 5.18 :Düşük çözünürlüklü videodan alınan bazı çerçeveler için kestirilen çerçveler	42
Şekil 5.19 : α_i ve α_s değerlerine bağlı PSNR değerleri	43
Şekil 5.20 : Veri tabanından seçilen tüm aday parçaların olasılıkları	44
Şekil 5.21 : Giriş görüntüsü için veri tabanından kestirilen parçaların olasılıkları.....	45
Şekil 5.22 : Veri tabanından seçilen tüm aday parçaların olasılıkları	46
Şekil 5.23 : Örnekleme sıklığı değiştirilen görüntüler	47
Şekil 5.24 : Enterpolasyon kullanılarak orjinal boyutuna getirilen görüntüler....	47
Şekil 5.25 : Farklı örnekleme sıklığındaki görüntüler kullanılarak kestirilen sonuç görüntüleri	48

SEMBOL LİSTESİ

B	: Giriş görüntüsü
B_Y	: Yüksek frekans bandındaki görüntü
B_O	: Orta frekans bandındaki görüntü
B_A	: Alçak frekans bandındaki görüntü
B_E	: Enerji görüntüsü
B_{NY}	: Normalize yüksek frekans görüntüsü
B_{NO}	: Normalize orta frekans görüntüsü
d^l_{ji}	: Dairesel ve radyal eğrilikler
x_i	: Boyutsuz dairesele ve radyal eğrilikler
Ψ	: Markov Ağları'nda gizli düğümler arasındaki geçiş fonksiyonu
Φ	: Markov Ağları'nda gözlem düğümleri ile gizli düğümler arasındaki geçiş fonksiyonu

MARKOV AĞLARI KULLANILARAK YÜKSEK ÇÖZÜNÜRLÜKLÜ GÖRÜNTÜ ÜRETİLMESİ

ÖZET

Görüntüler, kayıt edildikleri kameranın çözünürlüğünden daha yüksek çözünürlüklü bir ortamda gösterilmek istenirlerse, büyütülmeleri gerekir. Bu işlem için “cubic spline” [1,2,3,4] ve benzeri enterpolasyon teknikleri kullanılabilmekle birlikte, enterpolasyon görüntülerde yüksek frekans bileşenleri olarak niteleyebileceğimiz detay bilgisini ekleyememektedir.

Tez çalışması kapsamında düşük çözünürlüklü görüntülerde eksik olan detay bilgisinin kestiriminde kullanılabilecek öğrenmeye dayalı bir yöntem incelenmiştir. Amaç yüksek çözünürlüklü görüntülerden detay bilgisinin öğrenilmesi, öğrenilen bu detayların düşük çözünürlüklü görüntülerde eksik olan detay bilgisini bulmak için kullanılmasıdır. Bu yöntemin bulanıklık etkisinin giderilmesinden, NTSC, PAL yayınların “HD” (yüksek çözünürlüklü) panellerde gösterilmesine kadar birçok uygulama alanı bulunmaktadır.

Tez kapsamında, düşük çözünürlüklü görüntüler ile yüksek çözünürlüklü görüntüler arasındaki ilintiler, uzlamsal komşuluk ilişkileri de hesaba katılarak görüntü işleme uygulamalarında sıkça kullanılan Markov Ağları ile modellenmiştir [1,2,3,4]. Modellemede “yüksek çözünürlüklü bir görüntünün yüksek frekans bandındaki görüntüsü, orta frekans bandındaki görüntüsü biliniyorsa, alçak frekans bandındaki görüntüsünden bağımsız olarak bulunabilir” varsayımı kullanılmıştır.

Tez kapsamında, teknik yazında önerilmiş olan ve Markov Ağları ile aynı olasılıksal modeli kullanan, birinci dereceden komşuluk ilişkileri ile çözüme giden daha basit bir yöntem de gerçekleştirilmiştir [2,3]. Tek Geçiş Algoritması olarak isimlendirilen bu yöntemde düşük çözünürlüklü giriş görüntüsünden elde edilen yerel görüntü bilgisi ve bir önceki kestirimden elde edilen yüksek frekans bilgisi kullanılarak bir sonraki kestirim yapılmaktadır.

Testlerde, Markov ağırları kullanarak yüksek çözünürlüklü görüntü üretme işleminin başarımını arttırmak için, yüksek ve düşük çözünürlüklü ikililerin bulunduğu eğitim veri tabanı içine bol detay içeren yüksek çözünürlüklü örnekler yerleştirilmiştir

Tez kapsamında ayrıca sabit görüntüler için kullanılan yöntemler video çerçeveleri üzerinde uygulanarak yüksek çözünürlüklü video elde edilmeye çalışılmıştır. Sabit görüntüler için geliştirilen yöntemler video çerçeveleri üzerine uygulandığında, video klibi içerisindeki sabit alanlarda başarımlar artarken, hareketli alanlarda kırpışma (flickering) etkileri olduğu gözlemlenmiştir. Bunun nedeni videolar üzerinde görüntünün düzgün olarak yüksek çözünürlüklü hale gelebilmesi için gerekli hareket bilgisinin hesaba katılmamasıdır.

Yapılan testlerde değişik görüntülerden oluşturulmuş bir veri tabanı kullanarak düşük çözünürlüklü görüntülerin yüksek çözünürlüklü karşılıkları kestirilmeye çalışılmıştır. Düşük çözünürlüklü giriş görüntüsünün benzerleri veri tabanına eklenmesi durumunda elde edilen yüksek çözünürlüklü görüntüde başarımın arttığı gözlemlenmiştir. Aynı veri tabanları ve giriş görüntüleri Tek Geçiş Algoritması ile birlikte kullanarak yüksek çözünürlüklü görüntüler kestirilmiş ve sonuçlar Markov Ağırları ile bulunan sonuçlarla karşılaştırılmıştır.

Tez kapsamında ayrıca Markov Ağırları ile modellemede kullanılan düğüm geçiş fonksiyonlarındaki “alfa” parametresinin değişik değerleri için yüksek çözünürlüklü görüntüler kestirilerek, kestirilen ve orijinal görüntüler arasındaki PSNR değerleri hesaplanmıştır. Kestirim sırasında oluşturulan yüksek frekanslı görüntü parçaları ile giriş görüntüsünden çıkarılan düşük frekanslı parçaların eşleşme olasılıkları incelenmiştir. Yüksek çözünürlüklü görüntülerden veri tabanı oluşturulurken kullanılan örnekleme sıklığı değiştirme yönteminde farklı örnekleme sıklıkları kullanarak değişik veri tabanları oluşturulmuş, bu veri tabanı kullanarak yapılan kestirimlerde algoritmanın başarımı incelenmiştir.

Yapılan bu testler sonucunda Markov Ağırları kullanarak yapılan kestirimlerde elde edilen yüksek çözünürlüklü görüntülerin Tek Geçiş algoritması ile yapılan kestirimlerde elde edilen yüksek çözünürlüklü görüntülerden daha fazla detay içerdiği görülmüştür. İki algoritmada da başarımın artma koşulunun, veri tabanı içinde bulunan görüntülerin,

düşük çözünürlüklü giriş resmine benzerliğinin artması olduğu saptanmıştır. Sabit görüntülerde kullanılan bu yöntemlerin hareketli görüntülere uygulandığında, hareketli görüntüler içindeki sabit alanların yüksek çözünürlüklü hale gelebildiği, hareketli alanlarda ise titreşim etkilerinin ortaya çıktığı saptanmıştır. Bu sorunun aşılabilmesi için videoda bulunan hareket bilgisinin kullanımı gerekmektedir.

GENERATING HIGH RESOLUTION IMAGES USING MARKOV NETWORKS

SUMMARY

Images have limitations at the pixel resolution at which they are sampled. It can not be zoomed beyond these limitations without a degradation of the quality. If it is tried to display image on a device that has higher resolution from the record device, images must be zoomed. Zooming can be done by using interpolation techniques such as cubic spline unfortunately interpolation usually brings a blur effect on the image and suffers from addition of the missing high frequency details. Our goal is to estimate plausible high frequency details to low resolution images, which are learned from the high frequency images. There are lots of applications, such as blur removal, de-convolution, enlarging photographs, converting Standard Definition videos to HD format.

In this thesis work, learned relations between high resolution images and low resolution images with spatial neighboring relations are modelled with Markow Networks which is frequently used in image processing applications [1,2,3,4]. For a given high resolution image, the highest frequency information is conditionally independent from the information in the lowest frequency band if we know the medium frequency information.

Estimation of the high frequency data is done by using example based approach. Main idea is to learn fine details from the different image regions then use the learned relationship to predict the fine details in other input images. The relationship between the high resolution and the low resolution images are modeled by using the high frequency image data and medium frequency image data. The local image information is not enough to estimate plausible high frequency data. Spatial neighborhood effects must be taken into account. For this purpose the relationships between the frequency images are modeled with the Markov Network Algorithm.

There is another way, which called One Pass Algorithm that is using same probabilistic relationships as Markov Networks, and estimating the missing high frequency

information by looking the first degree neighbors. The input image is divided into patches and traversed in raster scan order. At each step a high frequency patch is selected from the database by using the local image information and the previously determined high frequency information. This operation is done for the all patches that are extracted from the low resolution input image.

Concerning video sequences, methods that are used for still images are applied videos and tried to increase resolution of the input video. This operation caused high resolution regions on static parts of the video but also flickering effects on the moving objects. For generating true high resolution object motion must be taken into account.

For tests, a database is generated from the collection of high frequency and medium frequency images that are extracted from the high resolution images and use Markov Networks to model the relations in database. Using this database high resolution images are estimated for given low resolution input images. Second, database is updated with similar images of the low resolution input images and then estimation of high resolution images is done by using this database. It is seen that, if the similarity of the images in database and given input image increases, the performance of algorithm increases. Next, tests are repeated for One Pass Algorithm and high resolution images are estimated with using same databases and same low resolution input images.

Then, for testing effects of " α_i " and " α_s " parameters in Markov Networks, with different values of these two parameters PSNR differences are measured between estimated high resolution images and true high resolution images. The matching probabilities of low resolution patches and high resolution patches are observed. Then the effect of subsampling is tested with decreasing image resolution with using different subsampling ratios.

After comparison of estimated high resolution outputs, it is seen that Markov Networks algorithm works with better performance than One Pass Algorithm and finds more details for a given input image. Also it is seen that the similarity between images in database and low resolution input image has an effect on performance. If this similarity between database images and input images increase then performance of the algorithms increase. Concerning video sequences, it is seen that, methods that are used in still

images has an positive effect on the static regions but also causes flickering efect on the moving object. To estimate true high resolution video, object motion must be taken into account.

1. GİRİŞ

Görüntülerin çözünürlüğünü arttırma problemi göz önüne alınırsa, düşük çözünürlüklü bir görüntü, kayıt edildiği kameradan daha yüksek çözünürlüklü bir ortamda gösterilmek istendiğinde büyütülmesi gerekir. Bu işlem için “cubic spline” [1,2,3] ve benzeri enterpolasyon teknikleri kullanılabilir. Birlikte, enterpolasyon, düşük çözünürlüklü görüntülerde eksik olan detaylar bilgisini ekleyemez. Çözünürlük yükseltme işlemi, görüntülerde eksik olan detayları ekleyerek yüksek çözünürlüklü görüntü elde etme işlemi olarak tanımlanır. Amaç görsel olarak kabul edilebilir detayların kestirimini yapmaktır.

Geliştirilen yöntem, enterpolasyon sonucu görüntülerde eksilen detay bilgisinin kestiriminde kullanılabilecek öğrenmeye dayalı bir yöntemdir. Bu yöntemin ana fikri yüksek çözünürlüklü görüntülerden detayların öğrenilmesi ve bunların düşük çözünürlüklü görüntülerde eksik olan detay bilgisini bulmak için kullanılmasıdır. Çözünürlük yükseltme işlemi bulanıklık etkisinin giderilmesinden, NTSC, PAL yayınların HD panellerde gösterilmesine kadar birçok alanda kullanılmaktadır [3,4,5]. Yüksek çözünürlüklü görüntü üretme işlemi için giriş, düşük çözünürlüklü bir görüntü, çıkış ise aynı görüntünün yüksek çözünürlüklü versiyonudur.

Düşük çözünürlüklü görüntülerde eksik olan yüksek frekans bilgisinin eklenmesi kolay çözülebilir bir problem değildir. Farklı yüksek çözünürlüklü görüntüler uygulanacak aynı işlemler sonucunda benzer düşük çözünürlüklü görüntüleri üretebilmektedir. Başka bir deyişle aynı düşük çözünürlüklü görüntünün farklı yüksek çözünürlüklü benzerleri olabilir. Bu yüzden eklenmesi gereken yüksek frekans verisinin seçimi önem taşımaktadır [3,4,5]. Markov Ağları kullanılarak yüksek çözünürlüklü video üretilmesi için kullanılan yöntemler sabit görüntülerde kullanılan yöntemler ile aynıdır. Videodaki her çerçeve için bu işlemler tekrarlanır. Aradaki tek fark video üzerinde çalışacak veri tabanının oluşturulma şeklidir [4, 5].

Teknik yazında bu problemi çözmek için öğrenmeye dayalı teknikler önerilmiştir [1, 2]. Tez çalışması kapsamında benimsenen bu teknikler temel olarak; yüksek çözünürlüklü değişik görüntülerden, bir veri tabanı oluşturarak detayları öğrenmek, daha sonra öğrenilen bu bilgileri, veri tabanı dışındaki görüntülerde eksik olan detayları bulmak için kullanmak şeklindedir.

Çözünürlük yükseltme işlemi için kullanılan diğer yöntemlerden bir tanesi “cubic-spline” enterpolasyon tekniğinin başarımını arttırmak için Schreiber [18] tarafından önerilmiştir. Schreiber keskinleştirilmiş bir Gaussian enterpolasyon fonksiyonu kullanarak, pikseller arasında oluşan bilgi taşmasını azaltmaya çalışmıştır. Tez çalışması kapsamında da bir kısmı incelenen, Freeman [2, 3] tarafından gerçekleştirilen bir diğer yöntem ise sabit görüntülerin çözünürlüğünün artırılması için kullanılan metoddur. Amaç öğrenmeye dayalı teknikler kullanılarak orjinal görüntüde bulunmayan ve keskinleştirme işlemleri ile eklenemeyecek yüksek çözünürlüklü detaylar bulunmasıdır. Bishop tarafından geliştirilen yöntemde ise; düşük çözünürlüklü görüntüler ile yüksek çözünürlüklü görüntüler arasındaki geçişleri kullanıp, obje hareketini de hesaplayarak yüksek çözünürlüklü görüntü üretilmeye çalışılmıştır.

Tez 6 ana bölümden oluşmaktadır. Birinci bölüm girişten oluşur. Yüksek çözünürlüklü görüntü üretme işleminin tanımı ve hangi uygulamalarda kullanılacağı anlatılır. İkinci bölümde yüksek çözünürlüklü görüntüler ile düşük çözünürlüklü görüntüler arasında ilişkinin nasıl kurulacağı ve veri tabanında bu ilişkilerin nasıl saklanacağı anlatılmaktadır. Üçüncü bölümde görüntü işleme uygulamalarında sıkça kullanılan ve tez çalışması kapsamında da kullanılan Markov Ağ Topolojisi hakkında genel bilgiler verilmiştir.

Dördüncü bölümde düşük çözünürlüklü görüntülerde eksik olan detayların nasıl bulunabileceği gösterilmiş, ara basamaklarının nasıl yapılacağı anlatılmıştır. Markov Ağları'nın kullanılarak yüksek çözünürlüklü görüntü üretme işleminin nasıl modellendiği gösterilmiştir. Ayrıca problemin çözümünde kullanılabilecek birinci dereceden komşuluk ilişkisini kullanarak doğrudan eşleme yapan Tek Geçiş Algoritması anlatılmıştır. Ayrıca bu bölümde düşük çözünürlüklü videolardan yüksek çözünürlüklü videoların bulunma yöntemleri gösterilmiş, sabit görüntüler için çalışan yöntemlerin

videolar üzerine nasıl uygulanacağı anlatılmıştır. Beşinci bölümde Markov Ağları ve Tek Geçiş Algoritması kullanılarak sabit görüntülerde bulunan ara sonuçlar gösterilmiştir. Video çerçeveleri üzerinde bulunan sonuçlarda bu bölümde listelenmiştir. Altıncı bölümde sonuçlar tartışılmıştır.

2. ÖN İŞLEME ADIMLARI

Tez kapsamında incelenen çözünürlük yükseltme algoritması, denklem 2.1’de görüldüğü üzere, giriş görüntüsünün orta frekans bandındaki görüntüsü “ B_O ” biliniyor ise yüksek frekans bandındaki görüntüsü “ B_Y ” alçak frekans bandındaki görüntüsünden “ B_A ” dan bağımsızdır varsayımı dayanmaktadır [1,2,3,5]. Bu nedenle orta frekans bandındaki görüntü “ B_O ” biliniyor ise yüksek frekans bandındaki görüntü “ B_Y ” bulunabilir.

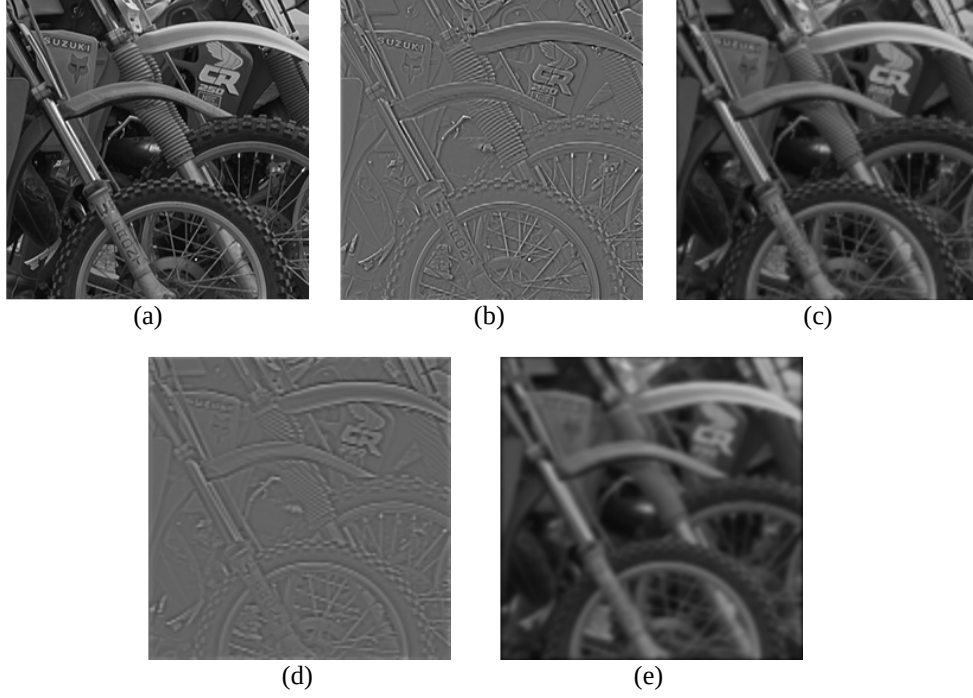
$$P(B_Y | B_O, B_A) = P(B_Y | B_O) \quad (2.1)$$

Burada temel yaklaşım, görüntülerin gri düzeylerinin belli bir olasılık dağılımına uyduğu, bu olasılık dağılımları öğrenilebilirse yüksek frekans bileşenlerinin oluşturulabileceği ve yüksek çözünürlüklü görüntü elde edilebileceği şeklindedir.

Modellenen bu ilişkiler bir veri tabanında saklanır. Veri tabanının oluşturulabilmesi için belirli işlemler uygulanır. Önce yüksek çözünürlüklü test görüntüleri temel frekans bantlarına ayrılır. Daha sonra bu farklı frekans bantlarındaki görüntüler 5×5 ve 7×7 boyutunda parçalara bölünerek, yüksek ve düşük çözünürlüklü ikililer olarak veri tabanına yerleştirilirler. Şekil 2.1(b) ve şekil 2.1(d) bu amaçla kullanılan görüntülere birer örnektir. Çözünürlük yükseltme işlemi, düşük çözünürlüklü görüntülerde eksik olan yüksek frekans verisini eklemek şeklinde özetlenebilir. İzlenmesi gereken genel adımlar aşağıdaki gibidir ve ilerleyen bölümlerde anlatılmaktadır.

- Eğitim görüntülerinin temel frekans bantlarına ayrılması ve bu frekans bantları arasındaki ilişkinin modellenmesi
- Farklı frekans bantlarındaki görüntülerin modellenen istatistiksel ilişkileri göz önünde bulundurularak parçalara ayrılması.
- Markov Ağları kullanarak değişik frekans bandındaki görüntü parçaları arasındaki istatistiksel bağların öğrenilmesi

- Belirlenen bu istatistiksel ilişkiler kullanılarak yüksek-düşük çözünürlüklü ikili parçaların veri tabanına yerleştirilmesi
- Yeni gelen düşük çözünürlüklü görüntü ve oluşturulan bu model kullanılarak yüksek frekans bilgilerinin bulunarak görüntüye eklenmesi.



Şekil 2.1 : Yüksek Çözünürlüklü Giriş Görüntüsünün Değişik Frekans Bantlarındaki Karşılığı (a) Yüksek Çözünürlüklü Giriş Görüntüsü, (b) “ B_Y ”- Yüksek Frekans Bandındaki Görüntü, (c) Ara Basamaklarda Elde Edilen Düşük Çözünürlüklü Görüntü, (d) “ B_O ”- Giriş Görüntüsünün Orta Frekans Bandındaki Karşılığı, (e) “ B_A ” – Giriş Görüntüsünün Alçak Frekans Bandındaki Karşılığı

2.1 Giriş Görüntüsünün Farklı Frekans Bantlarına Ayrılması

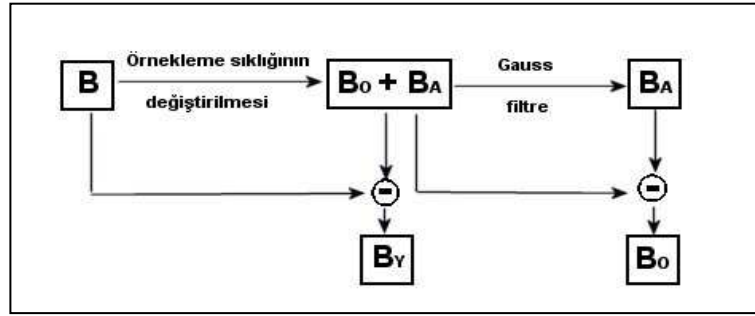
Yüksek çözünürlüklü görüntü üretilmesi işlemindeki temel varsayım denklem 2.1’de görüldüğü gibi, yüksek çözünürlüklü bir görüntünün orta frekans bandında bulunan görüntüsü “ B_O ” biliniyorsa, yüksek frekans bandında bulunan “ B_Y ” görüntüsü, alçak frekans bandında bulunan “ B_A ” görüntüsünden bağımsızdır [1]. Bu yüzden yüksek çözünürlüklü görüntü elde etme işlemine giriş görüntülerini 3 temel frekans bandına ayırarak başlanır.

Görüntüleri değişik frekans bantlarına ayırmak için literatürde iki genel yöntem önerilmiştir. Birinci yöntem Bishop [3] ve diğerlerinin yaptığı band geçirgen filtreler kullanarak görüntüyü üç ana frekans bandına (yüksek frekans, orta frekans, düşük frekans) ayırmak, ikinci yöntem Freeman tarafından kullanılan [1,2,3], örnekleme sıklığının azaltılıp tekrar artırılması ile yüksek çözünürlüklü görüntüleri üç temel frekans bandına ayırmak şeklinde tanımlanabilir. İki yöntem arasındaki tek fark yüksek frekans bandında bulunan görüntüyü elde etme şekilleridir. Bu basamaktan sonra izlenen adımlar birbirlerinin aynısıdır.

Yapılan tez çalışması kapsamında iki yöntem de denenmiş, örnekleme sıklığının değiştirilmesi ile elde edilen veri tabanında başarımın arttığı gözlenmiştir. Bu yüzden diğer basamaklarda örnekleme sıklığının değiştirilmesi esas alınmıştır.

2.1.1 Örnekleme Sıklığının Değiştirilmesi Yöntemi

Bu yöntem ile giriş görüntüsü, örnekleme düzeyi önce azaltılıp sonra tekrar aynı düzeye getirilerek, farklı frekans bandlarına ayrılmaya çalışılır. Şekil 2.2 yapılan işlemler özetlemektedir.



Şekil 2.2 : Örnekleme Sıklığı Değiştirilerek Giriş Görüntüsünün Farklı Frekanslara Ayrılması

Yüksek çözünürlüklü giriş görüntüsü önce bulanıklaştırılır, sonra yatay ve dikey ekseninde 1:2 oranında örnekleme sıklığı azaltılır. Toplam piksel sayısının 1:4 oranında düşürülür. Şekil 2.3'te yüksek çözünürlüklü görüntü ve aynı görüntünün 1:4 oranında küçültülmüş şekli görülmektedir.



Şekil 2.3 : Örnekleme Sıklığının Değiştirilmesinin Etkisi (a) Yüksek Çözünürlüklü Görüntü, (b) Örnekleme Sıklığı 1:2 Oranında Azaltılmış Görüntü

Örnekleme sıklığı azaltılmış görüntü enterpolasyon kullanılarak büyütülür. Orijinal boyutlarına getirildiğinde şekil 2.1 (c)'de görülen ara sonuç görüntüsü ortaya çıkar. Yüksek frekans bilgisi içermeyen bu ara sonuç görüntüsü, alçak ve orta frekans bandlarındaki görüntülerin toplamıdır. Elde edilen ara sonuç görüntüsü yüksek çözünürlüklü görüntüden çıkarılarak şekil 2.1(b)'de görülen, yüksek frekans bandındaki görüntü elde edilir.

$$B_{Y_{ij}} = B_{ij} - (B_O + B_A)_{ij} \quad i, j = 1.....N \quad (2.2)$$

Bir sonraki adımda alçak frekans bandında bulunan görüntü “ B_A ” yı bulmak için ara sonuç görüntüsü “ $(B_O + B_A)$ ” bir Gauss filtresinden geçirilerek düzgünleştirilir, görüntüde bulunan detaylar kaybolur. Bu işlem sonucunda giriş görüntüsünün düşük frekans bandında bulunan karşılığı elde edilir.

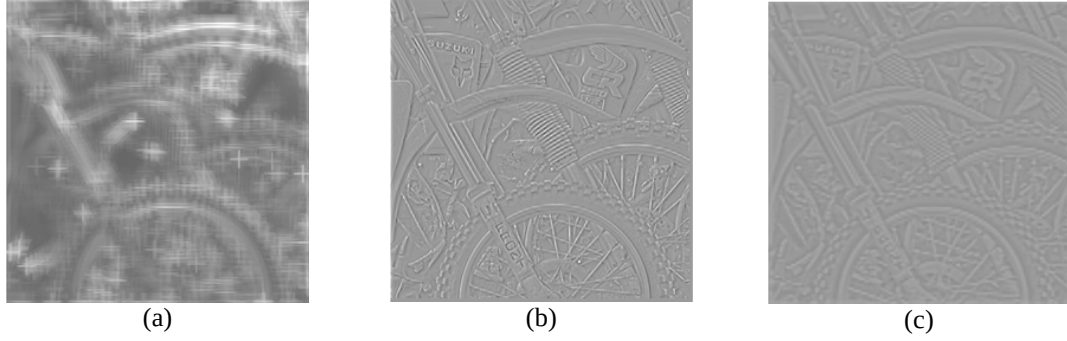
$$B_A = G * (B_O + B_A) \quad (2.3)$$

Son işlem olarak ara sonuç görüntüsünden alçak frekans görüntüsü çıkarılarak orta frekans bandındaki görüntü bulunur [3].

$$B_{O_{ij}} = (B_O + B_A)_{ij} - B_{A_{ij}} \quad i, j = 1.....N \quad (2.4)$$

Orta ve yüksek frekans görüntüleri arasındaki ilişkinin kontrastından bağımsız olduğunu kabul edilmektedir [1,2]. Bunun için farklı frekans bantlarında bulunan görüntüler için

kontrast normalizasyonu yapılır. Normalizasyon işlemi için önce bir enerji görüntüsü oluşturulur.



Şekil 2.4 : Normalizasyon İşlemi İçin Kullanılan Görüntüler ve Normalizasyon Sonucunda Oluşan Görüntüler (a) “ B_E ” Enerji Görüntüsü, (b) “ B_{NY} ” Normalize Yüksek Frekans Görüntüsü, (c) “ B_{NO} ” Normalize Orta Frekans Görüntüsü

Şekil 2.4 (a)’da oluşturulan enerji görüntüsü görülmektedir. Şekil 2.4 (b)’de ve (c)’de ise frekans bandlarındaki görüntülerin normalize edilmiş şekilleri görülmektedir. Enerji görüntüsünü oluşturmak amacı ile orta frekans bandında bulunan görüntüdeki her pikselin karesi alınır. Ortaya çıkan ara sonuç bir gauss filtreden geçirilerek düzgünleştirilir. Tekrar her pikselin karekökü alınarak şekil 2.3 (a)’da görülen enerji görüntüsü elde edilir.

$$B_{NO_{ij}} = \frac{B_{O_{ij}}}{\sqrt{G * (B^2_{O_{ij}}) + e}} \quad (2.5)$$

$$B_{NY_{ij}} = \frac{B_{Y_{ij}}}{\sqrt{G * (B^2_{O_{ij}}) + e}} \quad (2.6)$$

Denklem 2.5 ve denklem 2.6’da görüldüğü gibi yüksek frekans bandında bulunan görüntü ve orta frekans bandında bulunan görüntü, enerji görüntüsüne bölünerek kontrast normalizasyonu yapılır. Böylece şekil 2.3(b)’de görülen normalize yüksek frekans ve şekil 2.3 (c)’de görülen normalize orta frekans görüntüleri ortaya çıkar. İşlem sırasında sıfıra bölme hatalarından korunmak için ufak bir değer eklenir [3]. ($e = 0.00001$)

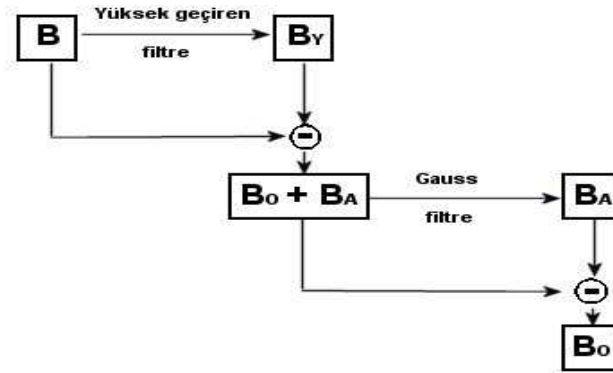
2.1.2 Filtreleme Yöntemi

Şekil 2.2’de yüksek çözünürlüklü görüntülerin filtreler kullanarak nasıl farklı frekans bantlarına ayrıldığı özetlenmiştir. Bu yöntem ile örnekleme sıklığı değiştirilerek frekans bandlarına ayırma yöntemi arasındaki tek fark, yüksek frekans görüntüsünün çıkarılma şeklidir. Diğer basamaklar birbirinin aynıdır.

Tablo 2.1 : Yüksek Geçiren Filtre

0	0.25	0
0.25	-1	0.25
0	0.25	0

İlk adım giriş görüntüsünün içerdiği yüksek frekans bandı içinde bulunabilecek verileri elde etmeye çalışmaktır. Yüksek çözünürlüklü giriş görüntüleri tablo 2.1’deki yüksek geçiren filtre ile evrişim yapılırlar. Böylece giriş görüntüsünün yüksek frekans bandında bulunan karşılığı elde edilir [3].



Şekil 2.5 : Filtreleme Yöntemi Kullanılarak Giriş Görüntüsünün Farklı Frekanslara Ayrılması

Elde edilen yüksek frekans görüntüsü, yüksek çözünürlüklü görüntüden çıkarılarak ara sonuç görüntüsü bulunur. Orta frekans ve alçak frekans görüntülerinin toplamı olan ara sonuç görüntüsü, şekil 2.1 (c)’de görülmektedir [3].

$$(B_O + B_A)_{ij} = B_{ij} - B_{Yij} \quad i, j = 1.....N \quad (2.7)$$

Veri tabanında yüksek frekans ve orta frekans görüntüleri arasındaki istatistiksel ilişkiler tutulacağı için ara sonuç görüntüsü kullanarak bir dizi işlem daha yapılır [1,2,3,4]. Önce orijinal görüntüden bulunan ara sonuç görüntüsü çıkarılarak, şekil 2.10’da gösterilen “ B_Y ” yüksek frekans bandında bulunan görüntü elde edilir.

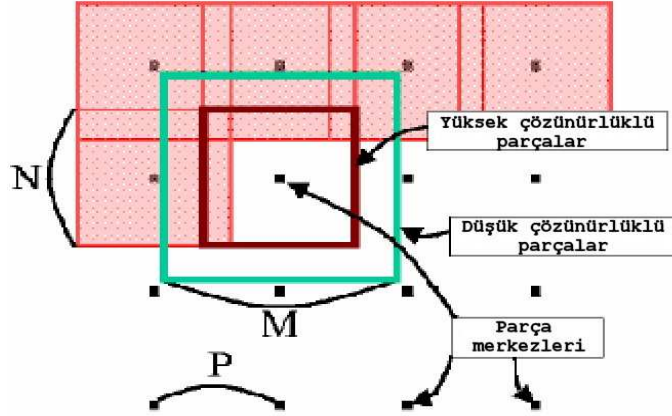
Bir sonraki adımda alçak frekans bandında bulunan görüntü “ B_A ” bulmak için ara sonuç görüntüsü “ $B_O + B_A$ ” bir gauss filtreden geçirilir. Görüntülerdeki keskin hatlar, detaylar belirsizleşir. Bulunan alçak frekans bandındaki görüntü “ B_A ” ara sonuç görüntüsünden çıkarılarak orta frekans görüntüsü elde edilir. Orta ve yüksek frekans görüntüleri arasındaki ilişkinin kontrastından bağımsız olduğu kabul edilir [1]. Bu yüzden bölüm 2.1.1’de anlatıldığı gibi görüntülerdeki lokal enerji kullanarak kontrast normalizasyonu yapılır.

2.2 Görüntülerin Parçalara Bölünmesi

Veri tabanı içinde arama işlemini kolaylaştırmak ve farklı frekanslar arası geçiş ilişkilerini modelleyebilmek amacı ile frekans bantlarındaki görüntüler $n \times n$ boyutunda parçalara bölünürler [5, 11]. Böylece birçok düşük çözünürlüklü giriş görüntüsü için kullanılabilecek genel bir veritabanı oluşturulur. Yüksek frekans görüntüleri 5×5 boyutunda parçalara, orta frekans görüntüleri 7×7 boyutunda parçalara ayrılır. Parça boyutları daha büyük seçilirse işlem sayısı azalır fakat görüntüdeki detaylar daha az yakalanır. Daha küçük boyutlarda parçalar seçilirse bulunan detaylar artmakla birlikte, işlem karmaşıklığı da artar.

Şekil 2.6’da yüksek ve orta frekans bantlarında bulunan görüntülerin parçalara bölünmesi gösterilmiştir. Bir görüntüyü parçalara ayırırken, yüksek frekans görüntüleri ile orta frekans görüntüleri arasındaki ilişkinin modellenebilmesi için, dikkat edilmesi gereken iki önemli nokta bulunmaktadır. Birinci nokta yüksek frekans parçaları ile orta frekans parçalarının aynı yerel bilgiyi taşıması gerekliliğidir. Bu Markov Ağlarında gözlem düğümleri ile gizli düğümler arasındaki geçiş ilişkilerini hesaplamada kullanılır. Yüksek frekans bandındaki görüntüler ile orta frekans bandındaki görüntüler parçalara ayrılırken parçaların merkezleri aynı seçilir. Böylece iki frekans bandında bulunan aynı

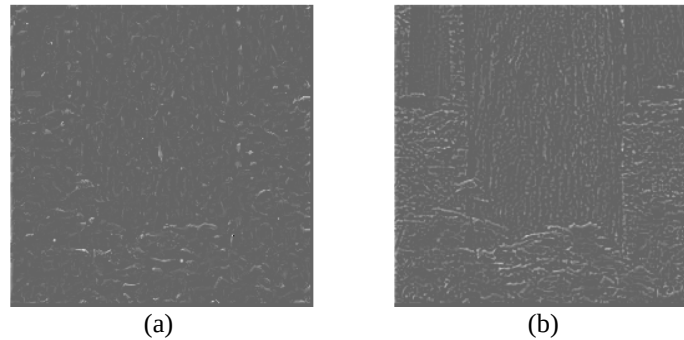
merkezli parçalar aynı bölgeden gelmiş olur ve şekil 2.6’da görüldüğü gibi iki parçada aynı yerel bilgiyi taşır.



Şekil 2.6 : Yüksek Frekans ve Orta Frekans Parçalarının Seçimi (Willian T. Freeman)

İkinci dikkat edilmesi gereken nokta parçalar seçilirken aralarında komşuluk ilişkilerinin hesaplanmasına olanak verecek bağıntıların bulunmasıdır. Bu bağıntılar Markov Ağları’nda gizli düğümler arasında komşuluk ilişkilerinin ve düğümler arasındaki geçişlerin olasılıklarını hesaplamak için kullanılacaktır. Geçiş fonksiyonları ve kullanımları hakkında detaylı bilgi bölüm 3’te anlatılmaktadır.

Eğer görüntülerinde bulunan yerel bilgi tek başına eksik olan yüksek frekans bilgisini bulmak için yeterli olsaydı veri tabanı içinde seçilen parçaların kendileri bu işlem için kullanılabilirdi. Şekil 2.7 görüntülerden çıkarılan yerel bilgi kullanılarak kestirilen parçaların tek başlarına detay bilgisinin kestirilmesinde yeterli olmadığı görülmektedir.



Şekil 2.7 : Komşuluk İlişkisinin Etkisi (a) Komşuluk İlişkileri Hesaba Katılmadan Kestirilen Yüksek Frekans Görüntüsü, (b) Komşuluk İlişkileri Hesaba Katılarak Kestirilen Yüksek Frekans Görüntüsü

Şekil 2.7 (b)'de aynı giriş görüntüsü için komşuluk ilişkileri hesaba katılarak kestirilen yüksek frekans görüntüsü, şekil 2.7 (a)'da ise uzamsal komşuluk ilişkileri göz önünde bulundurulmadan kestirilen yüksek frekans görüntüsü bulunmaktadır. Görüldüğü gibi komşuluk ilişkileri hesaba katılmadan kestirilen görüntü gerçek yüksek frekans görüntüsüne benzememektedir. Bu örnek bize görüntülerde bulunan yerel bilginin tek başına, eksik olan yüksek frekans bilgisinin kestirilmesinde yeterli olmayacağını göstermektedir [5,6,7]. Bu amaç ile görüntüler parçalara ayrılırken şekil 2.6'da gösterildiği gibi diğer parçalarla komşu oldukları bölgelerde piksellerin örtüşmeleri sağlanır. Yapılan çalışmada yüksek frekans bandından çıkarılan parçalar birer sütun ve satır, orta frekans bandından çıkarılan parçalar üçer sütun ve satır birbirleri ile örtüşmektedirler. Örtüşen bölümlerdeki pikseller daha sonra kestirim aşamasında komşuluk ilişkilerini hesaplamak için kullanılacaktır.

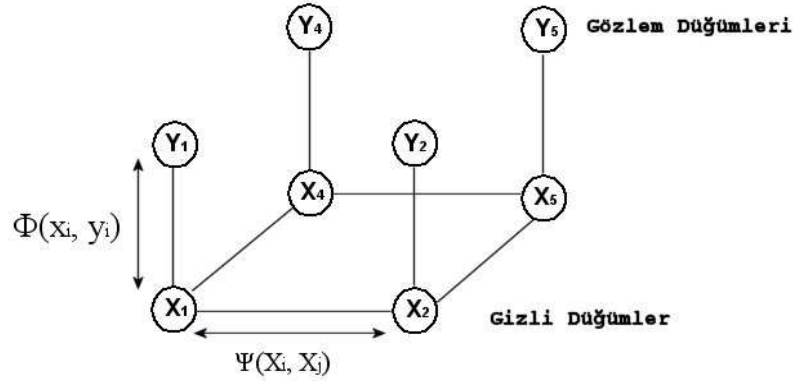
Tez çalışmasında yüksek çözünürlüklü görüntü üretmek için incelenen iki değişik yöntem bulunmaktadır [3]. İlk yöntem yüksek frekans parçaları ile orta frekans parçaları arasındaki ilişkiyi modellemek için Markov Ağ Modeli'ni kullanmaktır. İkinci yöntem ise Markov Ağ Modeli ile aynı komşuluk ilişkisini kullanan ve analitik bir algoritma olan Tek Geçiş Algoritması'nı kullanmaktır. Bu algoritma en yakın komşuluk ilişkisini kullanarak parçalar arasında doğrudan eşleme yapar [1,2].

3. MARKOV AĞ TOPOLOJİSİ İLE YÜKSEK-DÜŞÜK ÇÖZÜNÜRLÜK İLİŞKİSİNİN ÖĞRENİLMESİ

Literatürde bulunan çözünürlük yükseltme çalışmalarında olasılıksal modelleme yaklaşımının temel prensibi yüksek çözünürlüklü görüntüler ile düşük çözünürlüklü görüntüler arasındaki ilişkinin öğrenilmesidir [1,2,4]. Öğrenilen ilişkiler istatistiksel modeller kullanılarak modellenir. Bu model aracılığı ile düşük çözünürlüklü giriş görüntüsünde eksik olan detay bilgisi bulunur. Tez çalışması kapsamında incelenen “Markov ağları kullanılarak yüksek çözünürlüklü görüntü üretilmesi” işlemi de aynı prensip ile çalışmaktadır. Markov ağları kullanılarak modellenen yüksek ve düşük çözünürlüklü görüntüler arasındaki ilişki bulunarak diğer düşük çözünürlüklü görüntülerin çözünürlüklerinin yükseltilmesinde kullanılır.

Gerçeklenen sistemde önce yüksek çözünürlüklü görüntüler ile düşük çözünürlüklü görüntüler arasındaki ilişki modellenerek bir veri tabanında saklanır. Veri tabanının optimum olarak oluşturulması ve yüksek-düşük çözünürlüklü parçalardan oluşan ikililerin belirleyici örneklerinin saklanması algoritmanın başarımındaki en önemli etkidir. Bu yüzden veri tabanı oluşturulurken bol detay içeren yüksek çözünürlüklü görüntülerden yararlanılır.

Yüksek çözünürlüklü görüntü üretebilmek için, şekil 3.1’de görüldüğü gibi, yüksek çözünürlüklü parçalar ile düşük çözünürlüklü parçalar arasındaki ilişki Markov Ağları kullanarak modellenir. Şekil 3.1’de görülen daireler Markov Ağ Topolojisi’nde düğümleri, çizgiler ise düğümler arasındaki istatistiksel ilişkileri ifade etmektedir. Gözlem düğümleri ve gizli düğümler olmak üzere iki değişik düğüm bulunmaktadır. Gözlem düğümleri, giriş verisinden elde edilen bilgiyi kapsar, gizli düğümler ise bulunmaya çalışılan bilgiyi içermektedir.



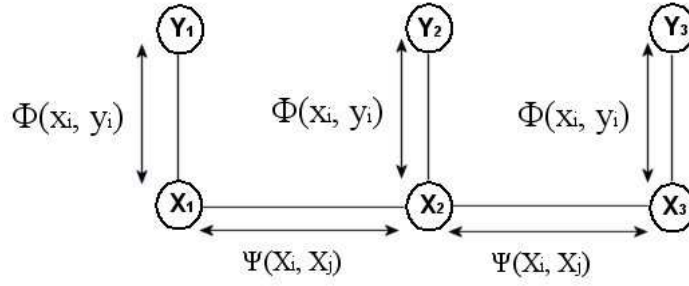
Şekil 3.1 : Döngü İçeren Markov Ağ Modeli

Şekil 3.1’te gösterildiği gibi, her gizli düğüm bir gözlem düğümüne ve kendisine komşu gizli düğümlere istatistiksel olarak bağlıdır. x_j gizli düğümünün bilinmesi, arasında tek geçişi olan y_j gözlem düğümü hakkında tüm bilgiyi sağlar. Çünkü x_j gizli düğümü ile y_j gözlem düğümü arasında tek geçiş bulunmaktadır. Aynı zamanda x_j gizli düğümü kendisine istatistiksel bağlantısı bulunan komşu gizli düğümler hakkında bilgi edinilmesini sağlar[6]. Markov Ağ Topolojisi iki aşamadan meydana gelir. İlk bölüm ağın parametrelerinin öğrenildiği aşamadır. İkinci bölümde gözlem düğümleri ve öğrenilen ağ parametreleri kullanılarak gizli düğümler arasında seçim yapılır. Amaç gözlem düğümlerini girişten seçip, gizli düğümlerin kestirimini sağlamaktır [6].

Bilinen gözlem düğümü ‘y’ kullanarak gizli düğümlerin ‘x’ kestirimi için önce denklem 3.1’de görülen koşullu olasılık $P(x/y) = c.P(x, y)$ hesaplanır. Bu işlem yapılırken ‘x’ değişkeni üzerinde bir sabit olan normalizasyon katsayısı olan $c = 1/P(y)$ hesap dışı bırakılır [6,7,8].

$$P(x/y) = \frac{P(x, y)}{P(y)} = c.P(x, y) \quad (3.1)$$

Şekil 3.1’de döngüler içeren Markov Ağ Modeli, şekil 3.2’de ise döngü içermeyen Markov Ağ Modeli görülmektedir. [6,7,9,10]



Şekil 3.2 : Döngü İçermeyen Markov Ağ Modeli

Markov rasgele alanları için ortak olasılık dağılım fonksiyonu, gizli düğümler “ x_i ” ve gözlem düğümleri “ y_i ” için denklem 3.2’de görüldüğü gibi, geçiş fonksiyonları $\Psi(x_j, x_j)$ ve $\Phi(x_k, y_k)$ cinsinden yazılabilir [6,11,12,13].

$$P(x_1, x_2, \dots, x_N, y_1, y_2, \dots, y_N) = \prod_{(i,j)} \Psi(x_i, x_j) \prod_i \Phi(x_i, y_i) \quad (3.2)$$

Bu denklemde (i,j) değerleri komşu düğümleri, N gizli ve gözlem düğümlerinin sayısını temsil etmektedir. $\Psi(x_j, x_j)$ fonksiyonu gizli düğümler arasında bulunan geçiş fonksiyonudur ve komşuluk ilişkilerinin hesaplanmasında kullanılır. $\Phi(x_i, y_i)$ fonksiyonu gözlem düğümleri ile gizli düğümler arasında bulunan geçiş fonksiyonudur [6,11,12,13].

$$\Psi(x_i^l, x_j^m) = e^{-|d_{ji}^l - d_{ij}^m| / 2\alpha_s^2} \quad (3.3)$$

Denklem 3.3’de gizli düğümler arasındaki geçiş potansiyelleri hesaplamak için kullanılan fonksiyon görülmektedir. x_i düğümü için seçilen l’inci aday parça ile x_j düğümü için seçilen m’inci aday parça arasındaki geçiş değeri hesaplanır. Burada i ve j birbirine komşu gizli düğümleri göstermekle birlikte, d_{ji}^l , x_i gizli düğümü l’ninci aday parçanın x_j sahne parçasıyla örtüştüğü noktalardan oluşan vektördür. Aynı şekilde d_{ij}^m , x_j gizli düğümü için seçilen m’inci aday parçanın x_i sahne parçasıyla örtüştüğü bölgedeki piksellerden oluşan vektördür [11].

$$\lambda(x) = \sum_k \sum_l (x_{i_{kl}} - x_{j_{kl}})^2 \quad i, j = 1, \dots, N \quad (3.4)$$

Denklem 3.3'te görülen d_{ji}^l ve d_{ij}^m uzaklıkları x_i gizli düğümü l'ninci aday parçası ile x_j gizli düğümünün m'inci aday parçası arasında örtüşen bölgelerde denklem 3.4 kullanılarak hesaplanır. Böylece bu parçaların komşuları ile ne kadar benzer bilgi taşıdığı belirlenmiş olur. $\Psi(x_i^l, x_j^m)$ geçiş fonksiyonu, veri tabanından seçilen N adet aday parça için hesaplandığında x_i ve x_j düğümleri arasındaki geçiş için "l" ve "m" ile indekslenen NxN boyutunda bir matris oluşur [1,2,3,5].

$$\Phi(x_i^l, y_i) = e^{-|y_i^l - y_0| / 2\alpha_i^2} \quad (3.5)$$

Benzer şekilde denklem 3.5'te gizli düğümler ve gözlem düğümleri arasındaki geçiş potansiyellerinin hesaplanmasında kullanılan fonksiyon görülmektedir. x_i^l sahne parçası ile ilişkili y_i^l görüntü parçası ve y_0 arasındaki uzaklık denklem 3.4 kullanılarak hesaplanır. Hesaplar x_i sahne parçaları için seçilen "l" adet aday parçalar için yapılır. Burada " x_i " düğümü için seçilen l'ninci aday parça giriş görüntüsünden çıkarılan " y_i " parça ile karşılaştırılır.

" x_i " gizli düğümü için kestirilen en iyi sahne, en düşük ortalama karesel hata (MMSE) veya sonsal olasılığın modu (MAP) kullanılarak hesaplanabilir. Genel olarak " x_i " gizli düğümü varsayımlar yapılmadan bulunması çok zor bir değerdir [6,7].

$\hat{x}_i$ yüksek çözünürlüklü parçayı bulmak için MMSE kestirimi, denklem 3.6'te görüldüğü gibi ortak olasılık dağılım fonksiyonunun ağırlıklı ortalaması alınarak bulunur. MAP kestirimini ise denklem 3.7'te görüldüğü gibi ortak olasılık dağılım fonksiyonunun değişkenler üzerinden maksimum değeri bularak hesaplanabilir.[5, 6]

$$\hat{x}_{jMMSE} = \sum_{x_j} x_j \sum_{all.x_i, i \neq j} P(x_1, x_2, \dots, x_N, y_1, y_2, \dots, y_N) \quad (3.6)$$

$$\hat{x}_{jMAP} = \arg \max_{x_j} (all.x_i, i \neq j). P(x_1, x_2, \dots, x_N, y_1, y_2, \dots, y_N) \quad (3.7)$$

Büyük ölçekli Markov Ağlarında boyutların fazla olması, maksimize edilmesi veya toplanması gereken parametrelerin fazla olmasına yol açacağından, ortak olasılık dağılım fonksiyonunun $P(x_1, x_2, \dots, x_N, y_1, y_2, \dots, y_N)$ hesaplanması çok zordur

[4,11,16]. Döngüsüz ağlarda MAP ve MMSE kestirimlerini bulmak için “Mesaj Aktarma” algoritması kullanılabilir. “Mesaj Aktarma” algoritması düğüm j’den i’ye M_i^j mesajlarını taşır. Denklem 3.19’da gösterildiği gibi $\hat{x}_i$ gizli düğümünün kestirimi kendisine gelen mesajlar ve “y_j” gözlem düğümü arasında bulunan istatistiksel geçişler hesaplanarak bulunabilir.

$$\hat{x}_i = \arg \max_{X_i} \Phi(x_i, y_i) \prod_j M_i^j \quad (3.9)$$

x_i düğümüne x_j düğümünden gelen mesajlar denklem 3.9’te gösterildiği gibi formüle edilmektedir. Mesaj M_i^j , x_j için hesaplanan geçiş fonksiyonlardan ve önceki iterasyonda x_j düğümüne gelen mesajlardan oluşan bir vektördür. Burada “j” değişkeni düğüm “i” için tüm komşu düğümleri, M_i^j ise x_j düğümünden x_i düğümüne gelen mesajları ifade etmektedir.

$$M_i^j = \Psi(x_i, x_j) \cdot \Phi(x_i, y_i) \prod_{k \neq i} \tilde{M}_j^k \quad (3.10)$$

$\tilde{M}_j^k$ bir önceki iterasyondan hesaplanan değerlerdir. $\tilde{M}_j^k$ mesajlarının ilk değerleri 1’lerden oluşan bir vektördür [4,11]. Girilen gözlem düğümü y_k için, gözlem düğümleri ve gizli düğümler arasındaki geçiş fonksiyonu $\Phi(x_k, y_k)$, x_k ’nin değişik durumları ile indekslenen bir sütun vektör olur. Gizli düğümler arasındaki geçiş fonksiyonu $\Psi(x_j, x_j)$ ise, düğüm i ve j için seçilen aday parçaları x_i ve x_j ’nin değişik durumları için hesaplanan bir matris olur.

4. YÜKSEK ÇÖZÜNÜRLÜKLÜ GÖRÜNTÜ ÜRETİLMESİ

Çözünürlük yükseltme işlemi, düşük çözünürlüklü görüntülerde eksik olan yüksek frekans bilgisini ekleyerek yüksek çözünürlüklü görüntü edilmesi olarak tanımlanır. Bu amaç için, veri tabanı oluştururken yapıldığı gibi, giriş görüntüsü değişik frekans bantlarına ayrılır. Düşük çözünürlüklü görüntülerde yüksek frekans bilgisini bulunmadığı için sadece orta frekans ve alçak frekans bandında bulunan görüntüler oluşturulur [1,2,3,4]. Orta frekans bandında bulunan görüntü 7×7 boyutunda parçalara bölünür. Görüntü parçalara bölünürken bölüm 2.3'te anlatıldığı gibi komşuluk ilişkilerinin göz önüne alınabilmesi için 3'er sütun ve satır komşu parçalar ile örtüşmesi sağlanır.

Düşük çözünürlüklü görüntüden seçilen parçalar kullanılarak veri tabanında arama yapılır. Tez çalışması kapsamında yüksek çözünürlüklü görüntü üretmek için iki değişik yöntem incelenmiştir. Birinci yöntem Markov Ağ Modelini kullanarak olasılıksal yöntemler ile detay bilgisini kestirmektir. Giriş görüntüsündeki parçaların her biri için benzer yerel bilgiyi taşıyan 16~20 adet aday parça veri tabanından seçilir. Aday parçalar aralarındaki istatistiksel geçişler modellenir. Amaç; veri tabanından seçilen aday parçaların birbirlerine olan komşuluk değerlerini belirleyip en uygun parçayı seçmektir. Görüntünün bütünlüğünü sağlayan en yüksek olasılıklı parça belirlenir.

İkinci yöntem olarak ta en yakın komşuluk ilişkisini kullanarak doğrudan eşleme yapan Tek Geçiş Algoritması ile düşük çözünürlüklü görüntülerde detay bilgisi kestirilir. Tek Geçiş Algoritması, Markov Ağları kullanılarak bulunan sonuçlara benzer sonuçlar üretmek ile birlikte daha hızlı çalışmaktadır. Algoritma, giriş görüntüsünden alınan lokal bilgiyi ve bir önceki aşamada kestirilen yüksek frekans verisini kullanarak veri tabanı içinde arama işlemi yapar.

4.1 Ön Adımlar

Süper çözünürlük işleminin çalışma adımları özetlenirse

- Giriş görüntüsünden orta frekans bandındaki bilginin çıkarılması
- Orta frekans bandında bulunan görüntünün parçalara bölünmesi
- Veritabanında Markov Ağları veya Tek Geçiş Algoritması kullanılarak olası parçaların bulunması
- Kestirilen yüksek frekans görüntüsü ve çıkış görüntüsünün oluşturulması

4.1.1 Giriş Görüntüsünün Farklı Frekans Bantlarına Ayrılması

Giriş görüntüsü düşük çözünürlüklü olduğundan içinde yüksek frekans bilgisinin olmadığını kabul edilir. Denklem 2.3 ve denklem 2.4'te gösterilen adımlar düşük çözünürlüklü görüntü için tekrarlanarak orta frekans bandında bulunan görüntü çıkarılır. Tüm işlemler sırasında kontrast değişikliklerinin etkisini azaltmak amacı ile, görüntü normalize edilir. Normalizasyon işlemi için orta frekans görüntüsünden bir enerji görüntüsü oluşturulup, orta frekans bandında bulunan görüntü bu enerji görüntüsüne bölünür. Böylece kontrastın etkisini minimuma iner.

4.1.2 Orta Frekans Görüntüsünün Parçalara Bölünmesi

Orta frekans bandında bulunan görüntü normalize edildikten sonra bölüm 2.3 'te yapıldığı gibi 7×7 boyutunda parçalara ayrılır. Böylece veritabanında arama işlemi yaparken aynı boyutlarda, eğitim görüntüsünün benzer bölgelerinden alınmış parçalar üzerinde karşılaştırma yapılabilir. Markov Ağ Modeli'nde komşu düğümler arasında geçiş değerlerinin hesaplanabilmesi için, veritabanı oluşturulurken yapıldığı gibi, orta frekans görüntüsünden çıkarılan parçaların 3'er sütun ve satır örtüşmesi sağlanır. Bu örtüşen bölgede bulunan pikseller komşuluk ilişkilerinin hesaplanması için kullanılacaktır [1,2,3,4,5] .

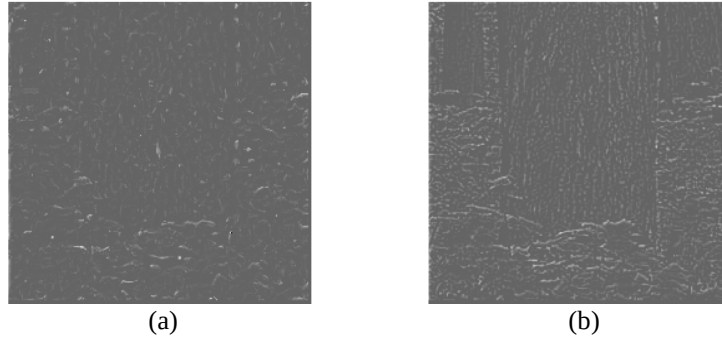
4.2 Yüksek Frekans Görüntüsünün Bulunması

Tez çalışması kapsamında, yüksek frekans görüntüsünün bulunması için iki yöntem incelenmiştir. İlk yöntem Markov Ağları kullanarak düşük çözünürlüklü görüntüden yüksek çözünürlüklü görüntüye geçişi sağlamaktır. Orta frekans görüntüsünden çıkarılan parçalar şekil 3.1'de görüldüğü gibi modellenir. Görüntüdeki her parça için veri

tabanından karşılık düşecek aday parçalar seçilir. Algoritma bu adaylar topluluğunu kullanarak, her parçanın olasılığını maksimum yapan adayı giriş görüntüsündeki parçalara karşılık olarak seçer. İkinci yöntem orta frekans görüntüsünden çıkarılan yerel bilgiyi ve bir önceki adımda kestirimi yapılan yüksek frekans parçaları ile aralarındaki komşuluk ilişkisini kullanarak sonucu bulan Tek Geçiş Algoritması'nı kullanmaktır [11]. Kestirilen parçalar bir araya getirilip yüksek frekans görüntüsü oluşturulur, giriş görüntüsüne göre ters normalizasyon yapıp, yüksek çözünürlüklü görüntü elde edilir.

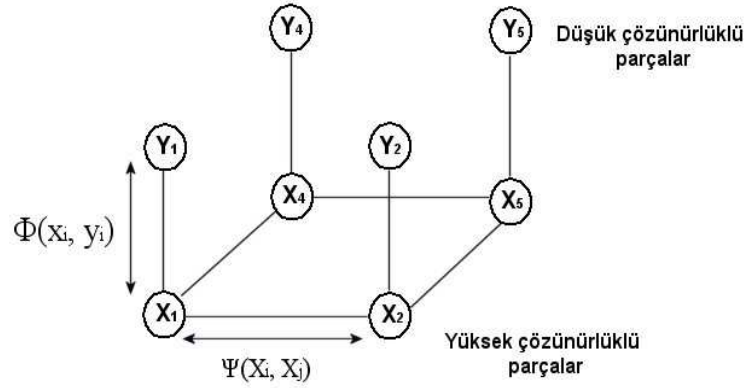
4.2.1 Markov Ağları Kullanılarak Yüksek Frekans Görüntüsünün Bulunması

Düşük çözünürlüklü görüntülerde bulunan yerel bilgi kullanarak kestirilen parçaların tek başlarına eksik olan detay bilgisinin oluşturulması için yeterli olmadığı şekil 4.1'den görülmektedir.



Şekil 4.1 : Komşuluk İlişkisinin Etkisi (a) Komşuluk İlişkileri Hesaba Katılmadan Kestirilen Yüksek Frekans Görüntüsü, (b) Komşuluk İlişkileri Hesaba Katılarak Kestirilen Yüksek Frekans Görüntüsü

Doğru bir kestirim yapabilmek için uzamsal komşuluk ilişkileri göz önünde bulundurulmalıdır. Bu tez çalışması kapsamında yüksek çözünürlüklü görüntü üretmek için gerekli yapı, uzamsal komşuluk ilişkilerini kullanarak çözümü bulan ve görüntü işleme uygulamalarında sıkça kullanılan Markov Ağ Topolojisi kullanılarak modellendi [8,9,13]. Şekil 3.1'de görülen Markov Ağ Topolojisi'nde daireler düğümleri, çizgiler ise düğümler arasındaki istatistiksel ilişkileri ifade etmektedir.



Şekil 4.2 : Markov Ağ Modeli (William T. Freeman)

Uygulanan bu model bize i'ninci pozisyonda bulunan gizli düğümün bilinmesi ile aynı pozisyonda bulunan gözlem hakkında tüm bilginin bulunabileceğini ifade etmektedir. Çünkü her gizli düğüm sadece bir gözlem düğümüne bağlantısı bulunmaktadır. Aynı zamanda i'ninci pozisyondaki gizli düğümün bilinmesi komşu gizli düğümler hakkında bilgi verebilmektedir [6,7,8]. Markov Ağ Topolojisi iki aşamadan meydana gelir. İlk bölüm öğrenim aşaması, ağın parametreleri öğrenilir. İkinci bölüm kestirim aşamasıdır, bu aşamada gözlem düğümleri ve öğrenilen ağ parametreleri kullanılarak gizli düğümler arasında seçim yapılır. Amaç gözlem düğümlerini girişten seçip, gizli düğümlerin kestirimini sağlamaktır. Bu modelin çalışabilmesi için düğümler ve aralarındaki geçişlere birer form bulunması gerekir [6].

4.2.1.1 Gizli Düğümler ve Gözlem Düğümleri

Gözlem düğümleri 'y_i', düşük çözünürlüklü görüntüden çıkarılan parçalar ile, gizli düğümler 'x_i' ise kestirilmek istenen yüksek çözünürlüklü parçalar ile oluşturulur. Markov Ağlarını yüksek çözünürlüklü görüntü üretme problemine uygulanırsa, amaç gözlem düğümlerini girişten seçip, gizli düğümlerin kestirimini sağlamaktır. Gözlem düğümleri giriş görüntüsünden çıkarılan orta frekans görüntüsünün 7×7 boyutunda parçalara bölünmesi ile elde edilir [1,2,3,4].

Benzer şekilde gizli düğümler, veri tabanından seçilen, giriş görüntüsündeki parçalara en yüksek olasılıkla benzeyen yüksek frekans parçalarından meydana gelmektedir. Aralarındaki ilişki bölüm 2.3'te anlatıldığı gibi kurulur. Veri tabanı oluşturulurken aynı

eğitim görüntüsünün yüksek ve orta frekans bandlarındaki karşılıklarından çıkarılan ikili olarak saklanırlar. Bu ikililer aynı yerel bilgiyi ve benzer uzlamsal komşuluk ilişkisini taşırlar.

$$\lambda(x) = \sum_i \sum_j (x_{k_{ij}} - x_{m_{ij}})^2 \quad i, j = 1.....N \quad (4.1)$$

Kestirim sırasında giriş görüntüsünden çıkarılan parçaların her biri için veri tabanında denlem 4.1 kullanılarak arama işlemi yapılır. $\lambda(x)$ ile hesaplanan uzaklığı en düşük olan 16~20 aday parça seçilir. Daha sonra bu parçalar kullanılarak, gözlem düğümleri ile gizli düğümler arasındaki geçiş potansiyeli, $\Phi(x_k, y_k)$ fonksiyonu ile hesaplanır. Komşu düğümler arasındaki geçiş potansiyelleri de örtüşen bölgelerde $\Psi(x_j, x_k)$ olasılık fonksiyonunu ile hesaplanır. İki geçiş fonksiyonu birlikte kullanılarak denklemler 3.5’te belirtilen olasılık fonksiyonunu maksimize eden yüksek frekans, orta frekans ikilisi seçilir [6].

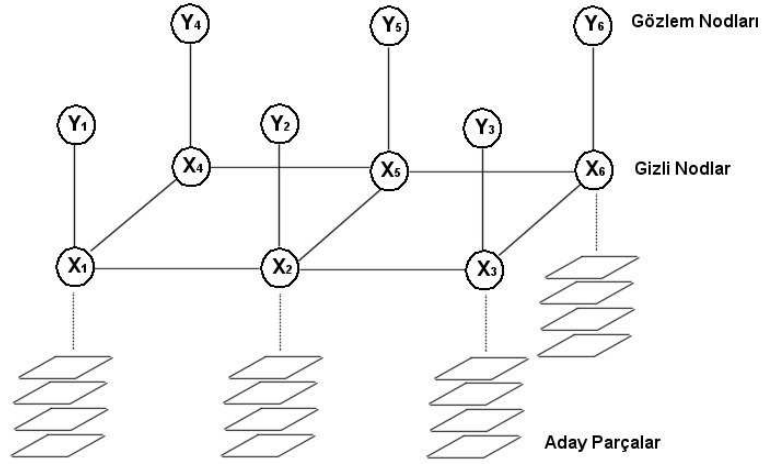
4.2.1.2 Geçiş Fonksiyonları

Markov Ağları’nda, gözlem düğümleri ile gizli düğümler arasındaki ilişkinin çıkarılması ve birbirine komşu gizli düğümler arasındaki istatistiksel ilişkilerin belirlenmesi için geçiş fonksiyonları kullanılır. Denklem 4.2’de görülen fonksiyon birbirine komşu gizli düğümler arasındaki geçiş değerlerini hesaplamak için kullanılır [1,2,3,4,5].

$$\Psi(x_i^l, x_j^m) = e^{-|d_{ji}^l - d_{ij}^m| / 2\alpha_s^2} \quad (4.2)$$

Bu denklemde x_j ve x_i düğümleri, Markov Ağ Modeli’nde birbirine komşu iki gizli düğümü, uygulamamızda kestirilmeye çalışılan sahne parçalarını belirtmektedir. Gizli düğümler x_j ve x_i için veritabanından N adet aday parça seçilmiş, seçilen aday parçalar “m” ve “l” ile indekslenmiştir. Burada i ve j birbirine komşu sahne parçaları (gizli düğümleri) göstermekle birlikte, d_{ji}^l , x_i sahne parçaları için seçilen l’ninci aday parçanın x_j sahne parçasıyla örtüştüğü noktalardan oluşan vektördür. Aynı şekilde d_{ij}^m , x_j sahne parçaları için seçilen m’inci aday parçanın x_i sahne parçasıyla örtüştüğü bölgedeki piksellerden oluşan vektördür [3,5].

Aday parçalar arasında örtüşen bölgelerde uzaklıkları denklem 4.1 kullanılarak hesaplanır. Böylece bu parçaların komşuları ile ne kadar benzer bilgi taşıdığı belirlenmiş olur. $\Psi(x_i^l, x_j^m)$ geçiş fonksiyonu, veri tabanından seçilen N adet aday parça için hesaplandığında x_i ve x_j düğümleri arasındaki geçiş için “l” ve “m” ile indekslenen NxN boyutunda bir matris oluşur [1, 2, 3, 4].



Şekil 4.3 : Veritabanından Seçilen Aday Parçaların Markov Ağları'na Uyarlanması

Şekil 4.3'te giriş görüntüsündeki her parça için veri tabanından seçilen aday parçaları görülmektedir. Bu aday parçalar her gizli düğüm için ayrı ayrı seçilir. Her aday parça için geçiş fonksiyonları tekrar hesaplanır [1,2,3].

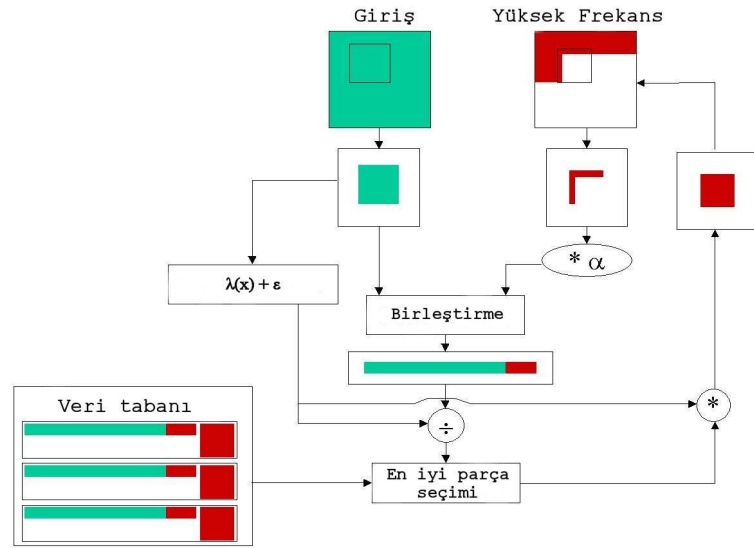
$$\Phi(x_i^l, y_i) = e^{-|y_i^l - y_0| / 2\alpha_i^2} \quad (4.3)$$

Aynı şekilde gözlem düğümleri ile gizli düğümler arasında ilişki hesaplanırken denklem 4.3'te görülen geçiş fonksiyonu kullanılır. x_i^l sahne parçası ile ilişkili y_i^l görüntü parçası ve y_0 arasındaki uzaklık denklem 4.1 kullanılarak hesaplanır. Hesaplar x_i sahne parçaları için seçilen “l” adet aday parçalar için yapılır. Dikkat edilmesi gereken nokta giriş görüntüsünden çıkarılan orta frekan parçaları ile veri tabanından seçilen aday parçalar arasında karşılaştırma yapılması gerektiğidir. Böylece veri tabanından seçilen parçaların düşük çözünürlüklü görüntü ile ne kadar benzer yerel bilgiyi taşıdığı belirlenmiş olur[2,3,5]. $\Phi(x_i^l, y_i)$ fonksiyonu, veritabanından seçilen “l” adet aday parça için

hesaplandığında x_i ve y_i düğümleri arasındaki geçiş değerleri için “1” ile indekslenen $N \times 1$ boyutunda bir vektör oluşur.

4.2.2 Tek Geçiş Algoritması

Tek Geçiş Algoritması birinci dereceden komşuluk ilişkisini kullanarak doğrudan eşleme yapar. Markov Ağları kullanılarak bulunan sonuçlara benzer sonuçlar üretmek ile birlikte daha hızlı çalışmaktadır. Kestirilen her yüksek frekans parçası giriş görüntüsündeki orta frekans parçasına ve bir önceki adımda kestirilen yüksek frekans parçalarına bağlıdır.



Şekil 4.4 : Tek Geçiş Algoritması (William T. Freeman)

Algoritma iki basamakta çalışmaktadır. Birincisi giriş görüntüsünden çıkarılan orta frekans parçaları ile veri tabanında bulunan orta frekans parçaları denklem 4.1 kullanılarak karşılaştırılır, en yakın olan 16~20 adet aday parça seçilir. Böylece benzer yerel bilgisi taşıyan parçalar eşlenmiş olur. İkinci basamakta örtüşen bölgelerde bir önceki işlemde kestirilen yüksek frekans parçası ile veri tabanından seçilen aday parçalar karşılaştırılır. En yakın olan aday parça seçilir. Böylece kestirilen yüksek frekans parçaları daha önce kestirilen yüksek frekans parçaları ile aynı doğrultuda olup benzer komşuluk bilgisini içerir. Şekil 4.4'te görüldüğü gibi Tek Geçiş Algoritması giriş görüntüsündeki parçalar için satır tarama yaparak çalışır. Her aşamada giriş

görüntüsünden çıkarılan yerel bilgi ve bir önceki aşamada kestirilen yüksek frekans verileri, bir sonraki parçanın kestirimini yapmak için kullanılır [2].

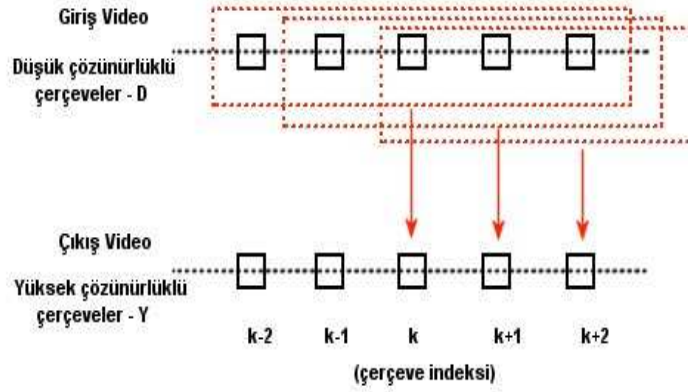
4.3 Yüksek Çözünürlüklü Video Üretmek

Önceki bölümlerde, düşük çözünürlüklü bir görüntüye eksik olan yüksek frekans bilgisinin eklenmesi ile algısal olarak kabul edilebilir yüksek çözünürlüklü görüntülerin elde edilebileceğini gördük. Aynı işlem düşük çözünürlüklü videolar üzerinde de uygulanabilir.

Bu bölümde önce teknik yazında yapılan çalışmalar özetlenmektedir [3, 4]. Yapılan tez çalışması kapsamında bu düzeye gelinememiş, tek bir görüntü için kullanılan yöntemler video kliplerine uygulanmıştır. Bir video klipinin çözünürlüğünü yükseltmek için D_{k-1} , D_k , D_{k+1} D_N şeklinde adlandırılan her video çerçevesi ayrı ayrı süper çözünürlük işlemine tabi tutulmuştur.

Çözünürlük yükseltme işlemlerinde kullanılan bazı tekniklerde sentetik videolar kullanılmış veya çerçeveler arasında genel kestirilebilir hareket bilgisi varsayılmıştır. Pratikte bir video klipi içindeki gerçek hareketin bilinmesi zordur. Schultz ve diğerleri [17] tarafından Bayesian süper çözünürlük modeline değişik hareket kestirim teknikleri uygulanmıştır. Schults tarafından [17] yapılan çalışmalar sonucunda parametrik hareket kestirim modelinin video çerçeveleri arasında oluşan genel hareketler için uygun olduğu, “block matching” ve “optical flow” yöntemlerinin de video çerçeveleri arasındaki düzensiz hareketlerin kestirimine uygun olduğu görülmüştür.

Hong [4] tarafından önerilen yöntemde tek bir video çerçevesinin çözünürlüğünün yükseltmesi problemine çözüm bulunmaya çalışılmıştır. Bu çalışmada, önceden kestirilmiş yüksek çözünürlüklü çerçeveler kayan pencere yöntemi kullanılarak veri tabanına eklenmiştir. Şekil 4.5’te görüldüğü gibi kayan pencere yöntemi, bir video çerçevesinin çözünürlüğünü yükseltmek için önceki ve sonraki çerçevelerden elde edilecek yüksek frekans bilgisinin kullanılmasıdır. Her yüksek çözünürlüklü Y_k çerçevesi için anlık çerçeveler M_t ($t \in T_k$), $T_k = \{k-2, k-1, k, k+1, k+2\}$ kullanılarak kestirilmeye çalışılmıştır.



Şekil 4.5 : Kayan Pencere Yöntemi İle Yüksek Çözünürlüklü Videoların Oluşturulması

Anlık çerçeveler sabit görüntülerde kullanılan çözünürlük yükseltme teknikleri uygulanarak bulunurlar. Yüksek çözünürlüklü Y_k çerçevesi bu anlık çerçeveler kullanılarak yapılan ikinci bir kestirim ile elde edilmeye çalışılır. Sabit görüntüler için kullanılan çözünürlük yükseltme işlemi çok fazla sayısal işlemler içermektedir. Yüksek çözünürlüklü çerçeveler için kayan pencere yöntemi ile ikinci bir kestirim yapmak işlem karmaşıklığını çok fazla arttırmaktadır.

Bishop [3] tarafından uygulanan yöntemde yüksek ve orta frekans bandlarında ayrılmış görüntülerden geçişlerin öğrenildiği bir veri tabanı kullanılarak video kliplerin çözünürlüğünün yükseltilmesine çalışılmıştır. Kullanılan yöntem ile video içindeki hareketsiz bölgelerde başarı sağlanmış, ikinci adımda video içindeki hareket hesaba katılmıştır. Bishop tarafından yapılan çalışmada başarıyı arttırmak amacıyla önceden kestirilmiş yüksek çözünürlüklü çerçeveler bir sonraki çerçevenin kestirimi için veri tabanı içine eklenerek kullanılmıştır. Video çerçevelerinin çözünürlüklerini yükseltmek için Freeman [1, 2, 5] tarafından önerilen sabit görüntülerde kullanılan öğrenmeye dayalı yöntem kullanılmıştır. Temel problem video içinde hareket eden nesneler üzerinde titreşimler meydana gelmesidir. Bu titreşimlerin etkisini azaltmak için bir önceki basamakta işlem yapılan düşük çözünürlüklü çerçeve ve karşıtı olan kestirilmiş yüksek çözünürlüklü çerçeve işlemlere katılmıştır.

Hong [4] tarafından önerilen başka bir yöntemde ise çok seviyeli blok eşleme yöntemi ile video içindeki hareket bilgisi bulunarak sabit görüntülerde kullanılan çözünürlük yükseltme yöntemleri ile birleştirilmiştir. Algoritma iki aşamada özetlenebilir;

- Yüksek çözünürlüklü anahtar çerçevelerin (key frames) oluşturulması
- Yüksek çözünürlüklü normal çerçevelerin oluşturulması

İlk aşamada, sabit görüntüler için kullanılan çözünürlük yükseltme işlemi video içindeki belirlenmiş anahtar çerçeveler D_t (düşük çözünürlüklü t zaman indeksli çerçeve) üzerinde de kullanılır. Diğer çerçeveler için anahtar çerçevelerin kestirilmiş yüksek çözünürlüklü halleri veri tabanına eklenir.

Anahtar çerçevelerin oluşturulmasındaki amaç D_t düşük çözünürlüklü video çerçeveleri kullanılarak yüksek çözünürlüklü Y_t video çerçevelerini kestirmektir. Bu yöntem için düşük çözünürlüklü videonun çerçeveleri arasından anahtar çerçeveler seçilerek, üzerlerinde sabit görüntülerde kullanılan çözünürlük yükseltme teknikleri uygulanır. Anahtar çerçevelerin indeksleri, düşük çözünürlüklü çerçeveler $D_t(t = 1, 2, \dots, N)$ için $k=\{3, 8, 13, \dots, N\}$ ($k \in t$, $D_k \in D_t$ ve $Y_k \in Y_t$) olarak seçilebilir. Böylece D_k düşük çözünürlüklü çerçeveler için Y_k yüksek çözünürlüklü çerçeveler elde edilebilir. Eğer giriş videosu için önsel bilgiye sahip olunursa anahtar çerçeveler uyarlanabilir bir şekilde seçilebilir.

Düşük çözünürlüklü videoda D_m ($m \neq k$, $m \in t$) anahtar olmayan çerçevelerin çözünürlüklerini yükseltmek amacıyla önceki adımlarda oluşturulan anahtar çerçevelerin yüksek çözünürlüklü benzerleri veri tabanına yerleştirilir. Anahtar olmayan çerçeveler için bir önceki basamakta kestirilen Y_k yüksek çözünürlüklü anahtar çerçeveler ve D_t ile D_k arasındaki hareket bilgisi kullanılarak yapılır.

İçinde nesnelerin başka nesnelerle örtüşmediği (occlusion), fazla ışık farklılığına sahip bölgelerin bulunmadığı videolarda yüksek çözünürlüklü video Y_m , düşük çözünürlüklü çerçeveler D_t ve D_k arasında kestirilen hareket bilgisi $M^{(t-k)}$ kullanılarak bulunabilir. Bu hareket bilgisi daha önce kestirilen anahtar çerçevelere uygulanarak, yüksek çözünürlüklü anahtar olmayan çerçeveler elde edilebilir [4]

$$H_m = M^{(t-k)}(H_k); \forall k, m \in t \quad (4.4)$$

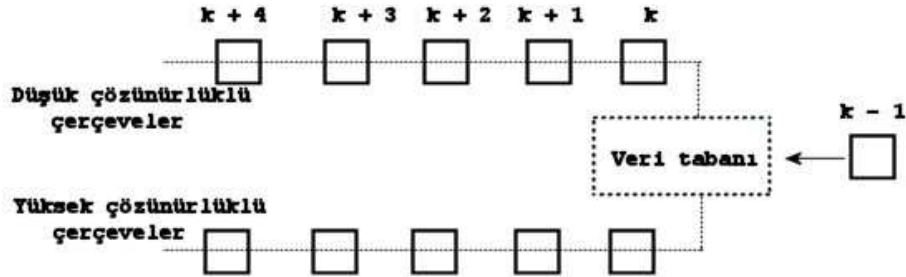
Genel olarak gerçek yüksek çözünürlüklü çerçeve Y_t içindeki her piksel “x” için, “ $\hat{y}$ ” Y_k çerçevesi içindeki kestirilen hareket bilgisini, “y” gerçek hareket bilgisini içersin;

$$y = M^{(t-k)}(x) \quad (4.5)$$

$$\hat{y} = M^{(t-k)}(x) \quad (4.6)$$

Benzer şekilde kestirilen yüksek çözünürlüklü çerçeveler için ışık şiddeti $\hat{H}_t(x)$ ve gerçek yüksek çözünürlük çerçeveler için ışık şiddeti $H_t(x)$ aşağıdaki şekilde gösterilebilir. Sonuç olarak Markov Ağları kullanılarak yüksek çözünürlüklü video elde etmek için düşük çözünürlüklü giriş videosunda bulunan her çerçeve teker teker ele alınır.

Tez çalışması kapsamında yüksek çözünürlüklü video klipleri üretilmeye çalışılmış bu amaç için Bishop[3] tarafında önerilen yöntem uygulanmaya çalışılmıştır. Video çerçevelerinin çözünürlükleri sabit görüntülerde kullanılan [1, 2, 3] teknikler ile artırılmış, hareket bilgisi hesaplanarak video çözünürlüğünü yükseltme işlemleri bu çalışma kapsamında incelenememiştir. Sabit görüntüler için geçerli olan tekniklerin video için uygulanabilmesinin en önemli nedeni bir videoda bulunan çerçevelerin birbirine benzemeleridir.



Şekil 4.6 : Yüksek Çözünürlüklü Bir Video Klibinin Kestirimi İçin Kullanılan Yöntemler

Şekil 4.6’de bu tez çalışması kapsamında yüksek çözünürlüklü video üretmek için kullanılan yöntem özetlenmektedir. Bir video klibindeki her çerçeve için teker teker sabit görüntüler için kullanılan çözünürlük yükseltme işlemleri uygulanır. Kestirilen yüksek çözünürlüklü çerçeve bir sonraki kestirim için veri tabanına eklenir. İşlem bir sonraki çerçeve için tekrarlanır.

5. TEST SONUÇLARI

Tez çalışmasında incelenen yöntemlerin uygulaması Visual Studio programı ile C++ dili kullanılarak yazıldı. Hem Markov ağları ile çözüm hemde tek geçiş algoritması gerçekleştirildi. Markov ağlarının oluşturulması, parametrelerinin belirlenmesi ve çözümü için Intel tarafında yazılan PNL kütüphanesi kullanıldı [19]. Bir sonraki adımda Linux ortamına geçilip toplu iş kütükleri kullanılarak, çalışma süreleri kısaltıldı.

5.1 Veritabanında Kullanılan Görüntüler

Veri tabanının iyi bir şekilde oluşturulması algoritmanın başarımını doğrudan etkilemektedir. Bu amaç ile yüksek çözünürlüklü görüntülerden bir grup görüntü veri tabanını oluşturmak için kullanılır. Testlerin gerçekleştirilmesi için üç değişik veri tabanı kullanıldı. Birinci veritabanı içinde 512×512 boyutunda şekil 5.1’de görülen 8 değişik görüntü kullanıldı. İkinci veri tabanı şekil 5.2’de görülen bisiklet parçalarından alınmış 3 görüntüden oluşmaktadır, üçüncü veri tabanı, şekil 5.3’te görülen, değişik boylardaki 3 adet yüz görüntülerinden oluşmuştur.



(a)



(b)



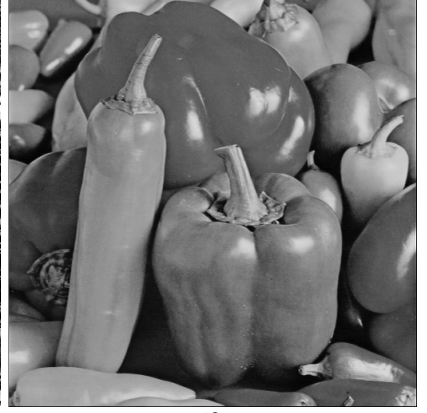
(c)



(d)



(e)



(f)



(g)



(h)

Şekil 5.1 : Genel Veri Tabanını Oluşturan Görüntüler



(a)



(b)



(c)

Şekil 5.2 : Bisiklet Veri Tabanında Bulunan Görüntüler

Şekil 5.2’de bisiklet veri tabanını oluşturmak için kullanılan, değişik boyutlardaki 3 görüntü görülmektedir. Bu görüntüler kullanılarak oluşturulan veri tabanı “Bike.sr” içinde toplam 53.534 yüksek çözünürlüklü, orta çözünürlüklü ikililer bulunmaktadır.

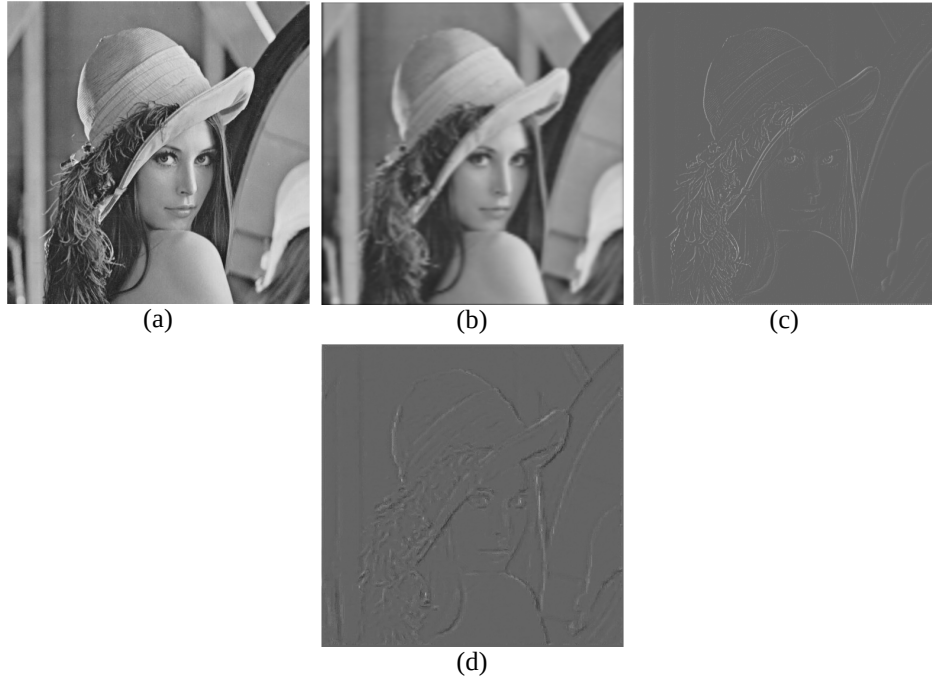


Şekil 5.3 : Yüz Veri Tabanını Oluşturmak İçin Kullanılan Görüntüler

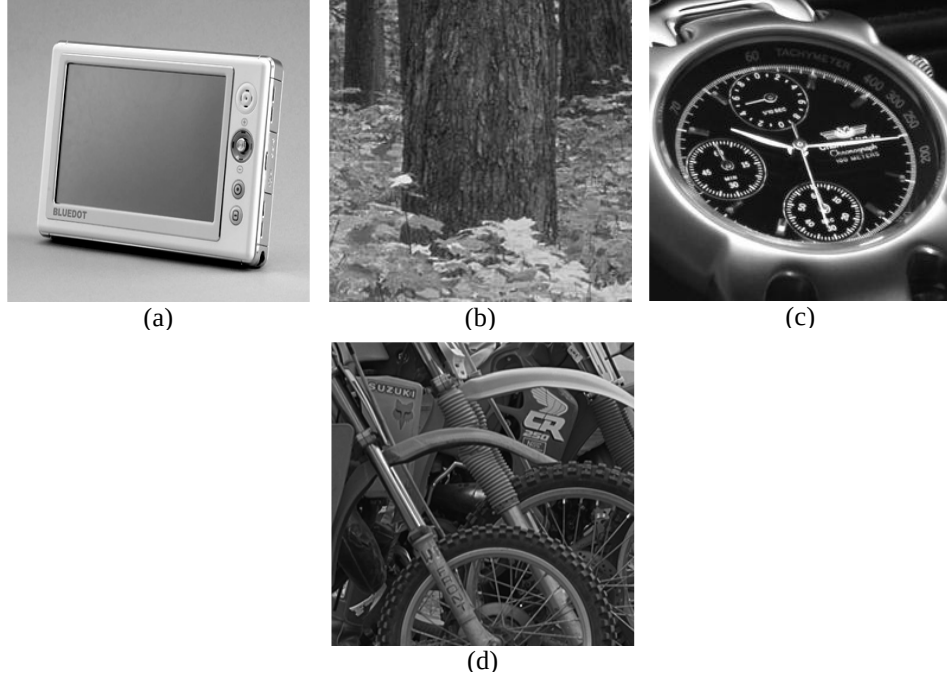
Yukarıda, şekil 5.3’te görünen üç görüntü kullanılarak yüz görüntüleri için kullanılan veri tabanı “Face.sr” oluşturuldu. Bu veri tabanı 102.320 adet parça içermektedir.

5.2 Sonuçlar

Şekil 5.4(a)’da yüksek çözünürlüklü görüntü, şekil 5.4(b)’de düşük çözünürlüklü hali görülmektedir. Şekil 5.4(c)’de düşük çözünürlüklü görüntülerde eksik olan yüksek frekans görüntüsü, Şekil 5.4(d)’de ise kestirilen yüksek frekans görüntüsü görülmektedir.



Şekil 5.4 : Yüz Veri Tabanı Kullanılarak Bulunan Yüksek Çözünürlüklü Görüntüler (a) Yüksek Çözünürlüklü Görüntü, (b) Düşük Çözünürlüklü Görüntü, (c) Gerçek Yüksek Frekans Görüntüsü, (d) Kestirilen Yüksek Frekans Görüntüsü

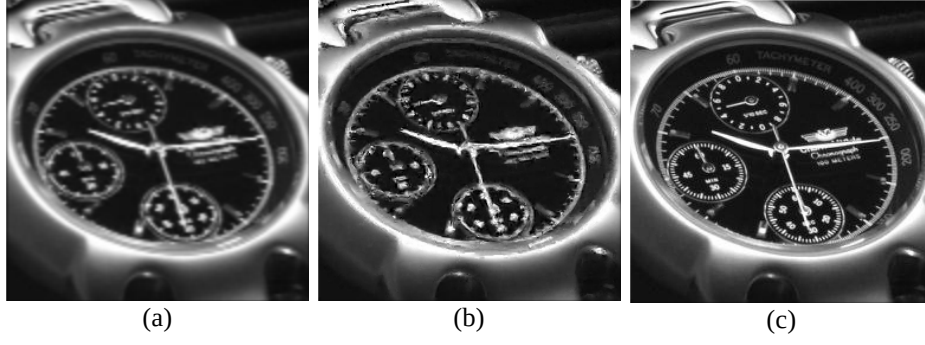


Şekil 5.5 : Yüksek Çözünürlüklü Test Görüntüleri, Bu Test Görüntülerin Düşük Çözünürlüklü Versiyonları Giriş Olarak Alınıp, Markov Ağları ve Tek Geçiş Algoritması Kullanılarak, Yüksek Çözünürlüklü Versiyonları Elde Edilmeye Çalışılacak.

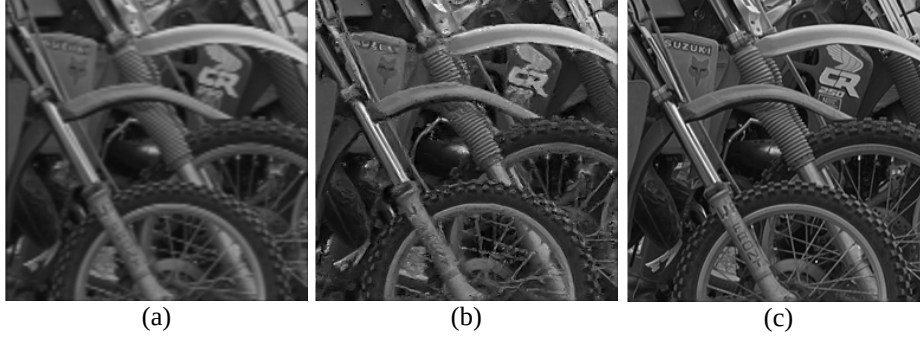
Yapılan testlerde Şekil 5.5'teki yüksek çözünürlüklü test görüntülerinin düşük çözünürlüklü versiyonları giriş verisi olarak alınmış, Markov Ağları ve Tek Geçiş Algoritması kullanarak yüksek çözünürlüklü versiyonları (şekil 5.5) kestirilmeye çalışılmıştır.

5.2.1 Markov Ağları Kullanılarak Bulunan Yüksek Çözünürlüklü Görüntüler

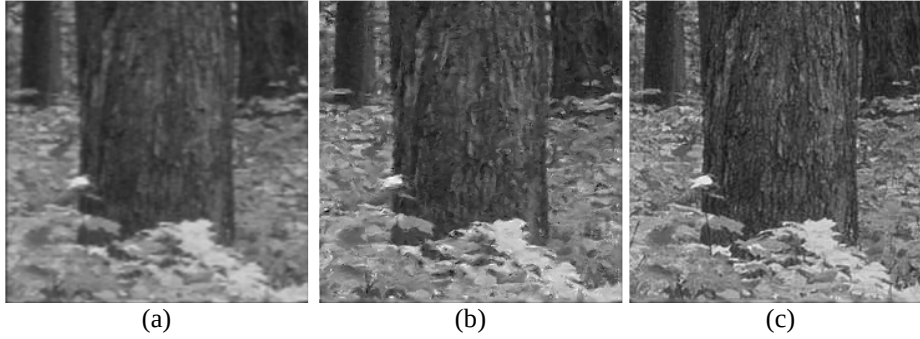
Bu alt bölümde Markov Ağları kullanarak bulunan sonuçlar görülmektedir. Veri tabanının etkisi iki basamakta incelenmiştir. İçinde benzer görüntülerin bulundurulduğu durumda ve benzer görüntülerin olmadığı durumda düşük çözünürlüklü giriş görüntülerinin çözünürlükleri yükseltilmeye çalışılmıştır.



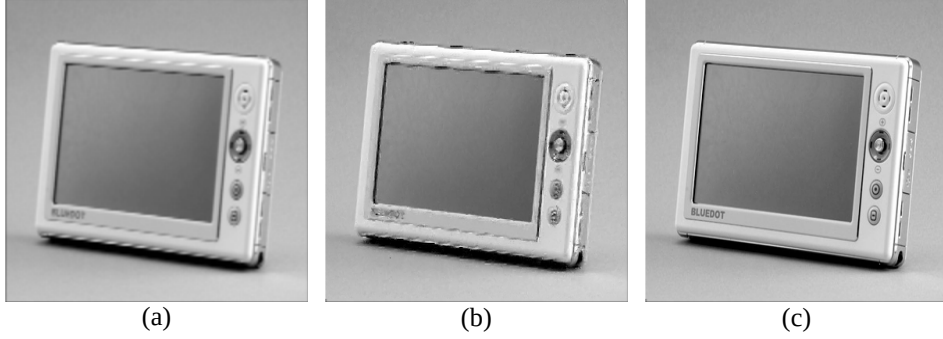
Şekil 5.6 : Markov Ağları kullanılarak Bulunan Sonuçlar (a) Düşük Çözünürlüklü Giriş Görüntüsü, (b) Veri Tabanı İçinde Benzerleri Bulunmadan Kestirilen Yüksek Çözünürlüklü Görüntü, (c) Veri Tabanı İçinde Giriş Görüntüsünün Benzerleri Varken Kestirilen Yüksek Çözünürlüklü Görüntü



Şekil 5.7 : Markov Ağları kullanılarak Bulunan Sonuçlar (a) Düşük Çözünürlüklü Giriş Görüntüsü, (b) Veri Tabanı İçinde Benzerleri Bulunmadan Kestirilen Yüksek Çözünürlüklü Görüntü, (c) Veri Tabanı İçinde Giriş Görüntüsünün Benzerleri Varken Kestirilen Yüksek Çözünürlüklü Görüntü



Şekil 5.8 : Markov Ağları kullanılarak Bulunan Sonuçlar (a) Düşük Çözünürlüklü Giriş Görüntüsü, (b) Veri Tabanı İçinde Benzerleri Bulunmadan Kestirilen Yüksek Çözünürlüklü Görüntü, (c) Veri Tabanı İçinde Giriş Görüntüsünün Benzerleri Varken Kestirilen Yüksek Çözünürlüklü Görüntü

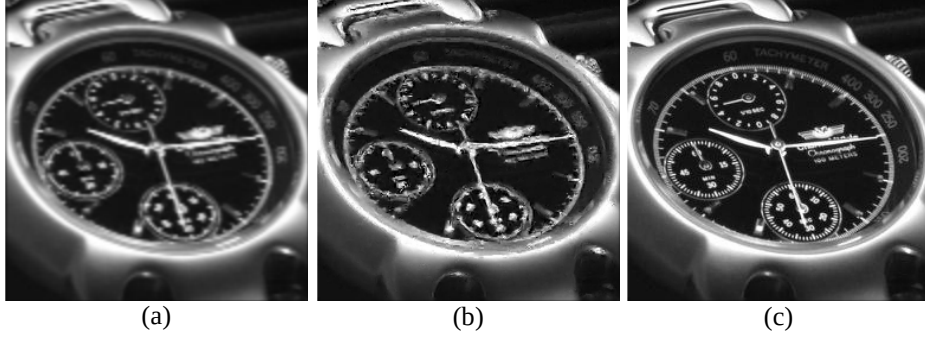


Şekil 5.9 : Markov Ağları kullanılarak Bulunan Sonuçlar (a) Düşük Çözünürlüklü Giriş Görüntüsü, (b) Veri Tabanı İçinde Benzerleri Bulunmadan Kestirilen Yüksek Çözünürlüklü Görüntü, (c) Veri Tabanı İçinde Giriş Görüntüsünün Benzerleri Varken Kestirilen Yüksek Çözünürlüklü Görüntü

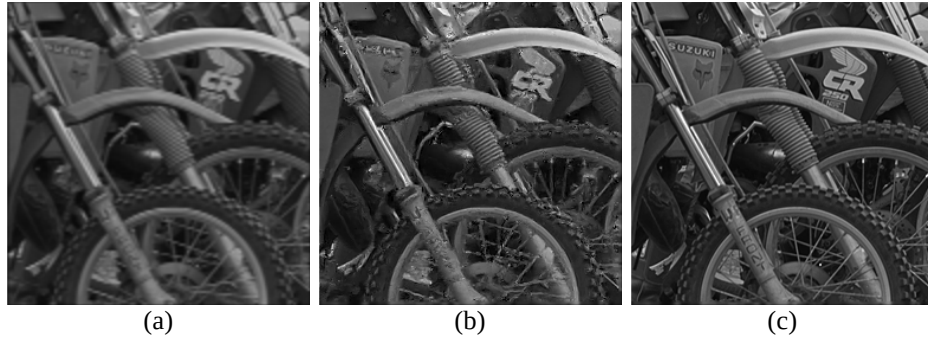
Yukarıda değişik düşük çözünürlüklü görüntüler için kestirilen yüksek çözünürlüklü karşılıkları bulunmaktadır. Şekil 5.6(a)'da düşük çözünürlüklü giriş görüntüsü, şekil 5.6(c)'de veri tabanında giriş görüntüsünün benzerleri varken kestirilen görüntü, şekil 5.6(b)'de ise veri tabanında girişin benzerleri yok iken kestirilen sonuçlar görülmektedir. Benzer biçimde şekil 5.7, şekil 5.8 ve şekil 5.9'da aynı işlemler kullanılarak bulunan sonuçlar görülmektedir. Bütün testler 10 aday parça sayısı, " $\alpha_s = 300$ " ve " $\alpha_i = 60$ " değerleri kullanılarak yapılmıştır.

5.2.2 Tek Geçiş Algoritması Kullanılarak Bulunan Yüksek Çözünürlüklü Görüntüler

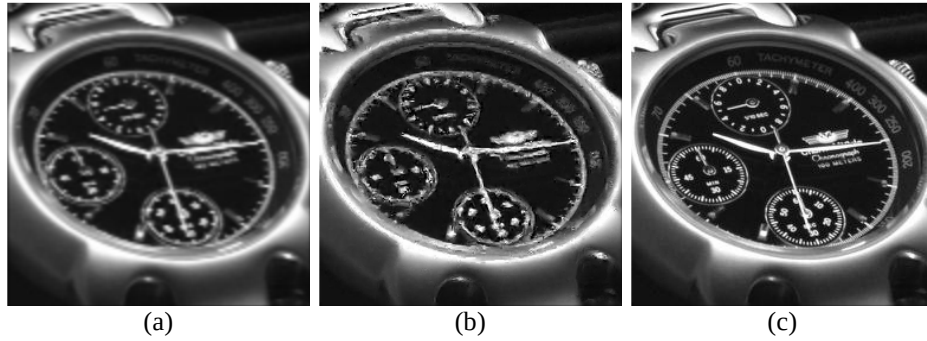
Bu bölümde Tek Geçiş Algoritması kullanılarak bulunan sonuçlar görülmektedir. Bir önceki bölümde olduğu gibi veri tabanının etkisi iki basamakta incelenmiştir. İçinde benzer görüntülerin bulundurulduğu durumda ve benzer görüntülerin olmadığı durumda düşük çözünürlüklü giriş görüntülerinin çözünürlükleri yükseltilmeye çalışılmıştır. Tek geçiş algoritması, Markov ağlarını kullanarak eşleme yapan yöntemle göre daha hızlı çalışmaktadır. Tek geçiş algoritması ile bulunan sonuçlar ise daha kötüdür.



Şekil 5.10 : Tek Geçiş Algoritması Kullanılarak Bulunan Sonuçlar (a) Düşük Çözünürlüklü Giriş Görüntüsü, (b) Veri Tabanı İçinde Benzerleri Bulunmadan Kestirilen Yüksek Çözünürlüklü Görüntü, (c) Veri Tabanı içinde Girişin Benzerleri Varken Kestirilen Yüksek Çözünürlüklü Görüntü



Şekil 5.11 : Tek Geçiş Algoritması Kullanılarak Bulunan Sonuçlar (a) Düşük Çözünürlüklü Giriş Görüntüsü, (b) Veri Tabanı İçinde Benzerleri Bulunmadan Kestirilen Yüksek Çözünürlüklü Görüntü, (c) Veri Tabanı içinde Girişin Benzerleri Varken Kestirilen Yüksek Çözünürlüklü Görüntü



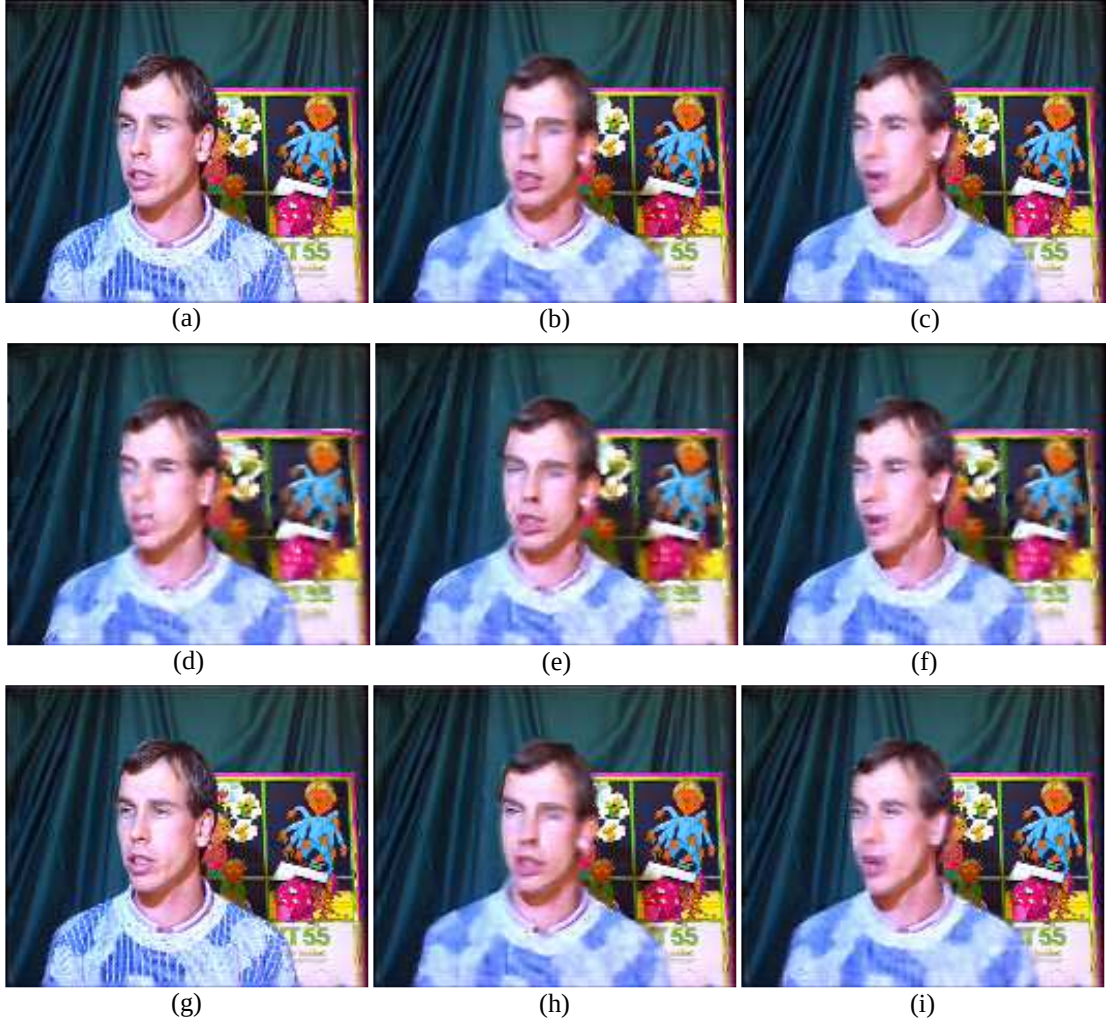
Şekil 5.12 : Tek Geçiş Algoritması Kullanılarak Bulunan Sonuçlar (a) Düşük Çözünürlüklü Giriş Görüntüsü, (b) Veri Tabanı İçinde Benzerleri Bulunmadan Kestirilen Yüksek Çözünürlüklü Görüntü, (c) Veri Tabanı içinde Girişin Benzerleri Varken Kestirilen Yüksek Çözünürlüklü Görüntü

5.3 Markov Ağları Kullanılarak Kestirilen Yüksek Çözünürlüklü Videolar

Şekil 5.13'te yüksek çözünürlüklü bir videodan alınmış bazı çerçeveler ve bunların düşük çözünürlüklü halleri görülmektedir. Şekil 5.13 (d), (e) ve (f)'de görülen görüntüler değişik veri tabanları kullanılarak çözünürlükleri yükseltilmeye çalışılacaktır. İlk testlerde kullanılan veri tabanı içerisinde şekil 5.1'de görülen görüntüler ve test videosunun yüksek çözünürlüklü halinin ilk çerçevesi bulunmaktadır. İkinci yapılan testlerde ise sadece Şekil 5.1'de bulunan görüntülerden oluşturulan veri tabanı kullanılmıştır. Üçüncü testlerde ise veri tabanı sadece yüksek çözünürlüklü videonun ilk çerçevesi kullanılarak oluşturulmuştur.



Şekil 5.13 : Yüksek Çözünürlüklü Videodan Çerçeveler ve Bunların Düşük Çözünürlüklü Karşılıkları (a) Yüksek Çözünürlüklü 1'inci Çerçeve, (b) Yüksek Çözünürlüklü 4'üncü Çerçeve, (c) Yüksek Çözünürlüklü 50'inci Çerçeve, (d) Düşük Çözünürlüklü 1'inci Çerçeve, (e) Düşük Çözünürlüklü 4'üncü Çerçeve, (f) Düşük Çözünürlüklü 50'inci Çerçeve,



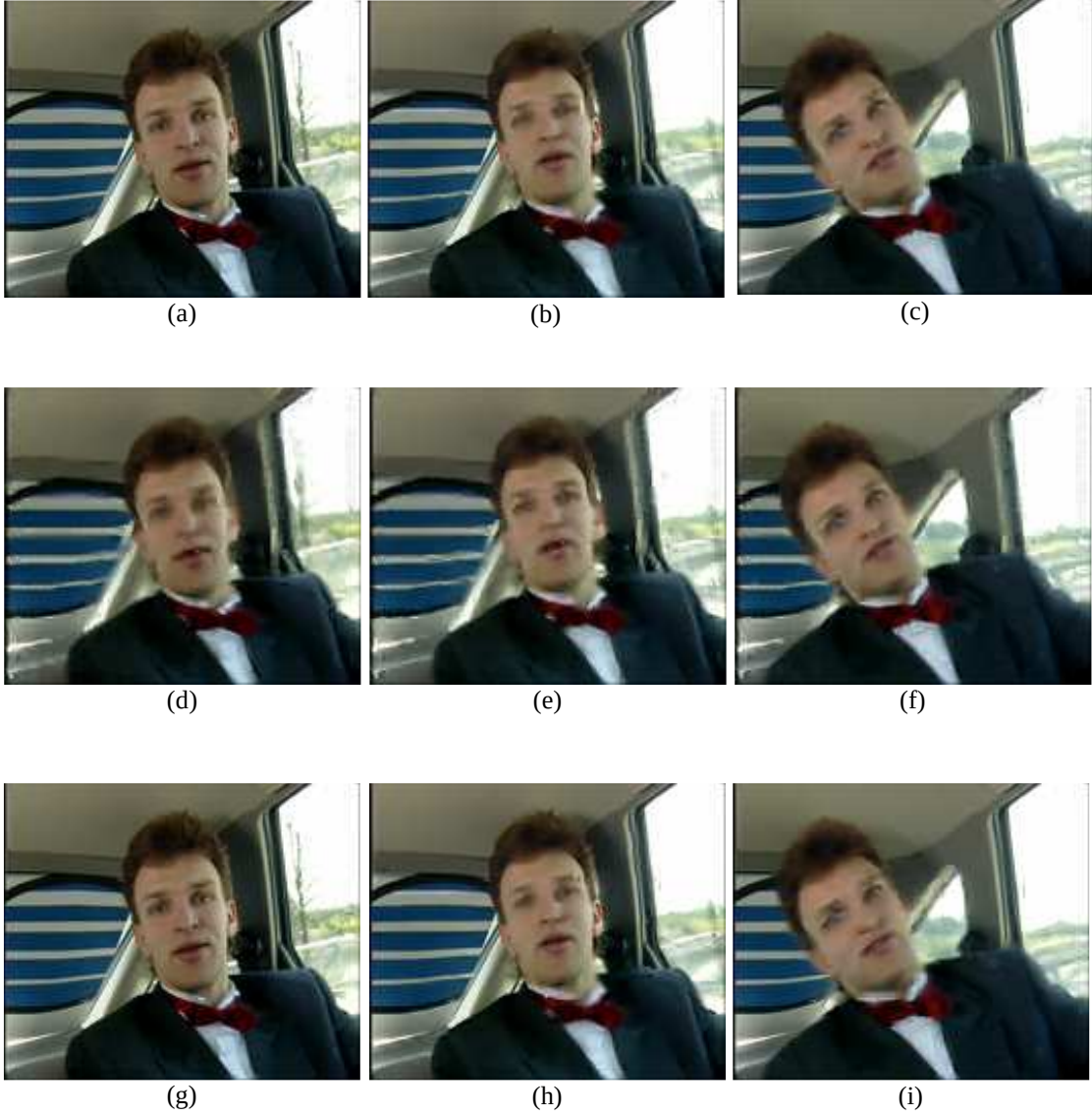
Şekil 5.14 : Düşük Çözünürlüklü Videodan Alınan Bazı Çerçeveler İçin Kestirilen Çerçeveler (a) İlk Veri Tabanı Kullanılarak Kestirilen 1'inci Çerçeve, (b) İlk Veri Tabanı İle Kestirilen 4'üncü Çerçeve, (c) İlk Veri Tabanı İle Kestirilen 50'inci Çerçeve, (d) İkinci Veri Tabanı İle Kestirilen 1'inci Çerçeve, (e) İkinci Veri Tabanı İle Kestirilen 4'üncü Çerçeve, (f) İkinci Veri Tabanı İle Kestirilen 50'inci Çerçeve, (g) Üçüncü Veri Tabanı İle Kestirilen 1'inci Çerçeve, (h) Üçüncü Veri Tabanı İle Kestirilen 4'üncü Çerçeve, (i) Üçüncü Veri Tabanı Kullanılarak Kestirilen 50'inci Çerçeve,

Şekil 5.14'te görüldüğü gibi veri tabanında bulunan görüntülerin düşük çözünürlüklü giriş görüntüsüne benzerlikleri arttıkça algoritmanın başarısıda artmaktadır. Şekil 5.14(b) ve (e) kıyaslandığında veri tabanı içine benzer tek bir çerçevenin eklenmesinin etkisi açıkça görülmektedir. Arka planda bulunan sabit objelerin çözünürlükleri belirgin bir şekilde artarken hareketli cisimler üzerinde titreşim etkileri meydana gelmektedir.

Bu amaç ile Bishop [3] ve diğerlerinin de yaptığı gibi yüksek çözünürlüklü video üretilmesi işleminde başarıyı arttırmak için önceden kestirilmiş yüksek çözünürlüklü videonun çerçeveleri veri tabanı içine eklenir. Şekil 5.15’de ve şekil 5.17’de yüksek çözünürlüklü bir videodan alınmış çerçeveler ve bunların düşük çözünürlüklü karşılıkları görülmektedir. Benzer biçimde şekil 5.16’da görülen test sonuçları, içerisinde şekil 5.1’de görülen görüntüler ile yüksek çözünürlüklü test videosunun ilk çerçevesi bulunan, içerisinde sadece Şekil 5.1’de görülen görüntüler bulunan ve içerisinde sadece yüksek çözünürlüklü videonun ilk çerçevesi bulunan üç değişik veri tabanları kullanarak bulunmuştur.



Şekil 5.15 : Yüksek Çözünürlüklü Video Çerçeveleri ve Bunların Düşük Çözünürlüklü Karşılıkları (a) Yüksek Çözünürlüklü 1’inci Çerçeve, (b) Yüksek Çözünürlüklü 20’inci Çerçeve, (c) Yüksek Çözünürlüklü 90’ıncı Çerçeve, (d) Düşük Çözünürlüklü 1’inci Çerçeve, (e) Düşük Çözünürlüklü 20’inci Çerçeve, (f) Düşük Çözünürlüklü 90’ıncı Çerçeve,



Şekil 5.16 : Düşük Çözünürlüklü Video İçin Kestirilen Çerçeveler (a) İlk Veri Tabanı Kullanılarak Kestirilen 1'inci Çerçeve, (b) İlk Veri Tabanı İle Kestirilen 20'inci Çerçeve (c) İlk Veri Tabanı İle Kestirilen 90'ıncı Çerçeve, (d) İkinci Veri Tabanı İle Kestirilen 1'inci Çerçeve, (e) İkinci Veri Tabanı İle Kestirilen 20'inci Çerçeve, (f) İkinci Veri Tabanı İle Kestirilen 90'ıncı Çerçeve, (g) Üçüncü Veri Tabanı İle Kestirilen 1'inci Çerçeve, (h) Üçüncü Veri Tabanı İle Kestirilen 20'inci Çerçeve, (i) Üçüncü Veri Tabanı İle Kestirilen 90'ıncı Çerçeve,



(a)

(b)

(c)



(d)

(e)

(f)

Şekil 5.17 : Yüksek Çözünürlüklü Bir Videodan Çerçeveler ve Bunların Düşük Çözünürlüklü Karşılıkları (a) Yüksek Çözünürlüklü 1'inci Çerçeve, (b) Yüksek Çözünürlüklü 30'uncu Çerçeve, (c) Yüksek Çözünürlüklü 80'inci Çerçeve, (d) Düşük Çözünürlüklü 1'inci Çerçeve, (e) Düşük Çözünürlüklü 30'uncu Çerçeve, (f) Düşük Çözünürlüklü 80'inci Çerçeve,



(a)

(b)

(c)



(d)

(e)

(f)



Şekil 5.18 : Düşük Çözünürlüklü Video İçin Kestirilen Çerçeveler (a) İlk Veri Tabanı Kullanılarak Kestirilen 1’inci Çerçeve, (b) İlk Veri Tabanı İle Kestirilen 20’inci Çerçeve, (c) İlk Veri Tabanı İle Kestirilen 90’ıncı Çerçeve, (d) İkinci Veri Tabanı İle Kestirilen 1’inci Çerçeve, (e) İkinci Veri Tabanı İle Kestirilen 20’inci Çerçeve, (f) İkinci Veri Tabanı İle Kestirilen 90’ıncı Çerçeve, (g) Üçüncü Veri Tabanı İle Kestirilen 1’inci Çerçeve, (h) Üçüncü Veri Tabanı İle Kestirilen 20’inci Çerçeve, (i) Üçüncü Veri Tabanı İle Kestirilen 90’ıncı Çerçeve,

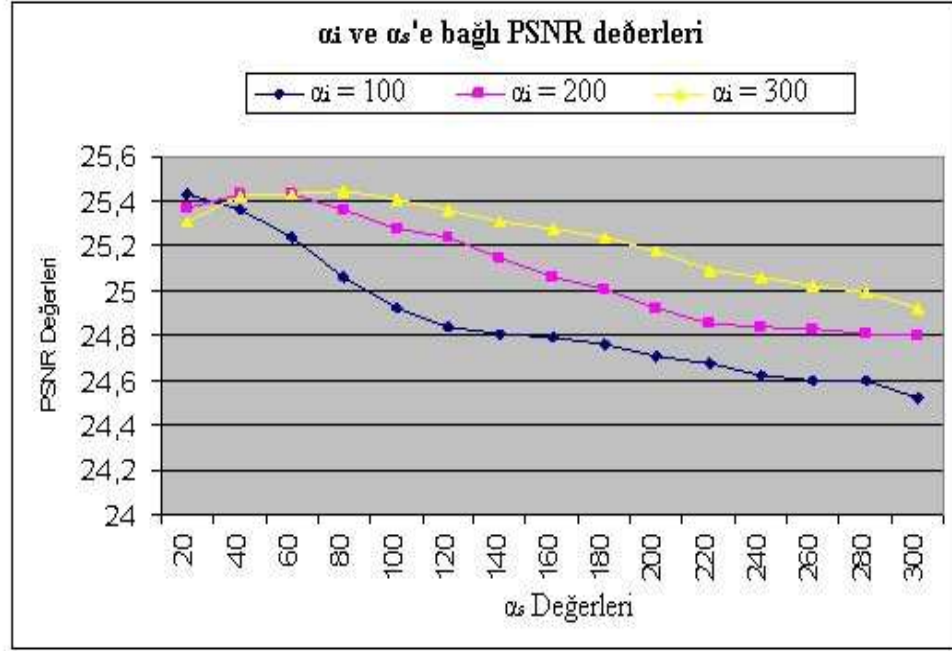
5.4 Alfa Parametresinin Başarıma Etkisi

Denklem 5.1’de gizli düğümler arasındaki geçiş potansiyellerinin, denklem 5.2’de gözlem düğümleri ile gizli düğümler arasındaki geçiş potansiyellerinin hesaplanması için kullanılan fonksiyonlar görülmektedir [2,3,5,12,13]. Bu denklemlerde bulunan “ α_s ” ve “ α_i ” katsayıları veri tabanına göre iteratif bir şekilde seçilir.

$$\Psi(x_i^l, x_j^m) = e^{-|d_{ji}^l - d_{ij}^m| / 2\alpha_s^2} \quad (5.1)$$

$$\Phi(x_i^l, y_i) = e^{-|y_i^l - y_0| / 2\alpha_i^2} \quad (5.2)$$

Her durum için kestirilen yüksek çözünürlüklü görüntü ile gerçek yüksek çözünürlüklü görüntü arasındaki PSNR değerleri hesaplandı. Değişik alfa değerlerinin başarıma etkisini incelemek için Şekil 2.1(c)’de görülen düşük çözünürlüklü görüntü giriş alınıp “Bike.sr” veri tabanında arama yapıldı.



Şekil 5.19 : PSNR Değerlerinin α_s ve α_i Değerlerine Bağlı Değişimi

Tablo 5.1’de “ α_s ” ve “ α_i ” parametrelerinin değişik değerlerinin etkisi görülmektedir. Algoritmanın başarımı aşağıdaki durumlarda artmaktadır;

- “ α_i ” sabit iken “ α_s ” azaldığı bölgelerde
- “ α_s ” sabit iken “ α_i ” arttığı bölgelerde

Tablo 5.1’de görüldüğü gibi gizli düğümler ile gözlem düğümleri arasındaki geçiş potansiyeli algoritmanın başarımı için temel etkindir.

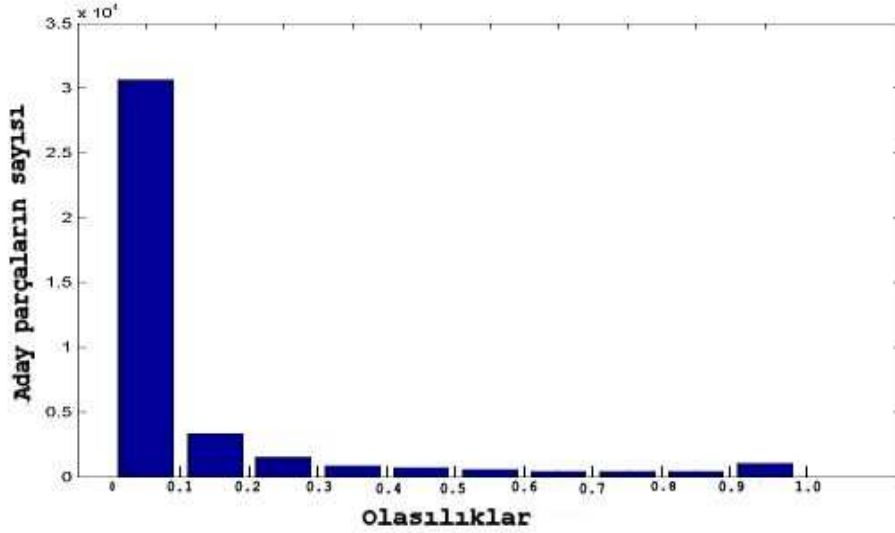
Tablo 5.1 : Değişken Alfa Değerleri İçin Hesaplanan PSNR Değerleri

S1(x)	S2 = 100	S2 = 200	S2 = 300
S1 = 20	25,43	25,37	25,43
S1 = 40	25,36	25,43	25,36
S1 = 60	25,24	25,43	25,24
S1 = 80	25,06	25,36	25,06
S1 = 100	24,92	25,28	24,92
S1 = 120	24,84	25,24	24,84
S1 = 140	24,81	25,15	24,81
S1 = 160	24,79	25,06	24,79
S1 = 180	24,76	25,01	24,76
S1 = 200	24,71	24,92	24,71
S1 = 220	24,68	24,85	24,68

S1 = 240	24,62	24,84	24,62
S1 = 260	24,60	24,83	24,60
S1 = 280	24,60	24,81	24,60
S1 = 300	24,52	24,80	24,52

5.5 Yüksek-Düşük Frekans Eşleme Olasılıkları

Yüksek-düşük frekans parçaların eşlenmesinde aday parçaların hangi olasılıklarının seçildiğinin belirlenebilmesi için şekil 2.1(c)'de görülen düşük çözünürlüklü görüntü giriş olarak alınmıştır.



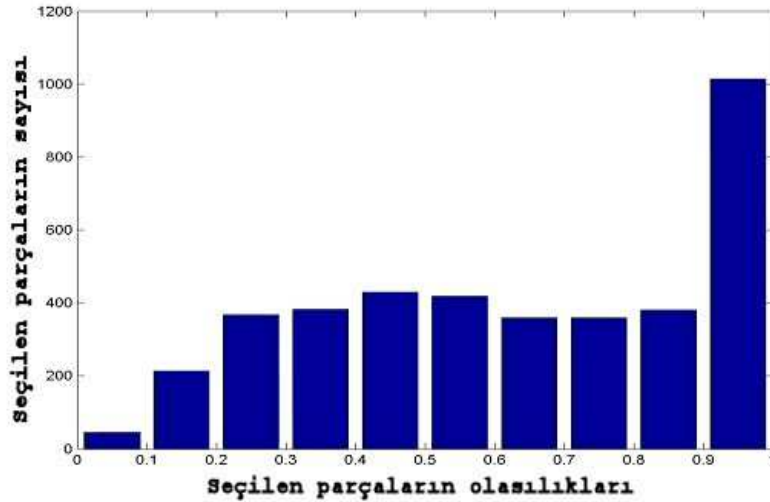
Şekil 5.20 : Veri Tabanında Seçilen Tüm Aday Parçaların Olasılıkları

Düşük çözünürlüklü görüntüdeki her bir parça için “Bike.sr” veri tabanından 10 adet aday parça denklem 4.1 kullanılarak seçilmiştir. Seçilen bu aday parçalar Markov Ağları ile modellenerek aralarındaki geçiş değerleri hesaplanmıştır. Şekil 5.20’de bu aday parçaların kestirim sırasında hesaplanan olasılık değerleri görülmektedir. Algoritmanın çalışması sonucunda aday parçalar arasından, geçiş fonksiyonları ile giriş görüntüsündeki her bir parçaya karşılık düşecek en yüksek olasılıklı parça seçilmiştir.

Tablo 5.2 : Düşük Çözünürlüklü Giriş Görüntüsü İçin Veri Tabanından Seçilen Aday Parçaların ve Sonuç Olarak Kestirilen Parçaların Sayısının Belirli Olasılık Aralıklarına Dağılımı

Olasılık	0-0.1	0.1-0.2	0.2-0.3	0.3-0.4	0.4-0.5	0.5-0.6	0.6-0.7	0.7-0.8	0.8-0.9	0.9-1
Tüm	30615	3312	1489	856	676	531	407	392	410	1046
Seçilen	41	161	333	394	424	433	375	362	407	1039

Tablo 5.2’de ilk satırda düşük çözünürlüklü giriş görüntüsü için veri tabanından seçilen tüm aday parçaların olasılıkları bulunmaktadır. İkinci satırda ise bu aday parçalar arasından, giriş görüntüsüne en uygun olarak seçilen parçaların olasılıkları görülmektedir. Şekil 6.21’de ise kestirilen yüksek frekanslı parçalarının olasılıkları görülmektedir. Aday parçalar arasından Markov ağları kullanarak yapılan eşlemede, veri tabanı içinden seçilebilecek en yüksek olasılıklı parçaların seçildiği görülmektedir.



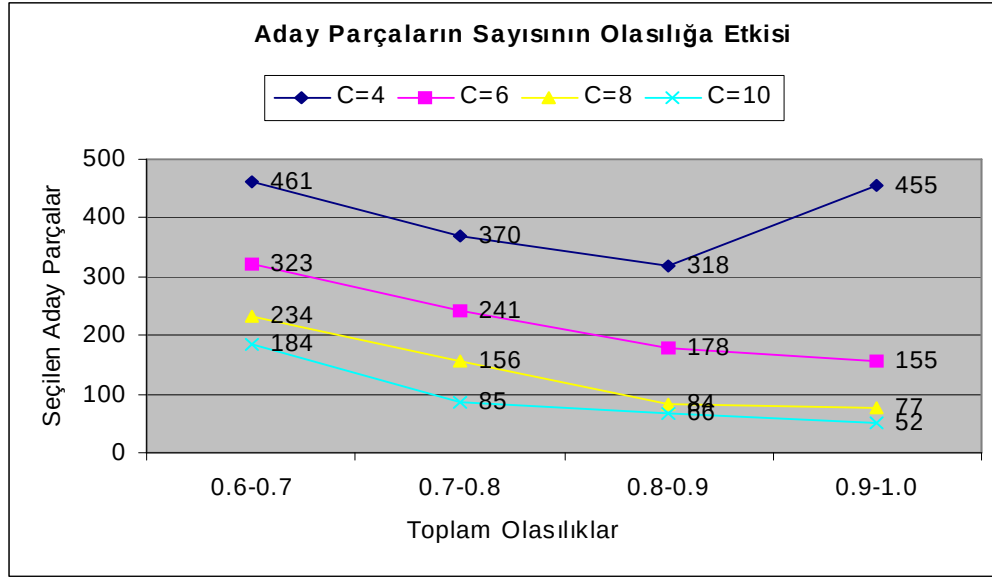
Şekil 5.21 : Giriş Görüntüsü İçin Veri Tabanından Seçilen Parçaların Olasılıkları

Tablo 5.3’te aday parça sayısına göre seçilme olasılığı 0.6 ve üstü olan parçaların sayısı görülmektedir. Bir düğüm için seçilen aday parçaların sayısı “C” arttıkça, eşleme sonunda bulunan parçaların seçilme parçaların olasılıkları da düşmektedir.

Tablo 5.3 : Veri Tabanından Seçilen Aday Sayısına Göre Parçaların Olasılıkları

Olasılık	0.6-0.7	0.7-0.8	0.8-0.9	0.9-1
C=4	461	370	318	455
C=6	323	241	178	155
C=8	234	156	84	77
C=10	184	85	66	52

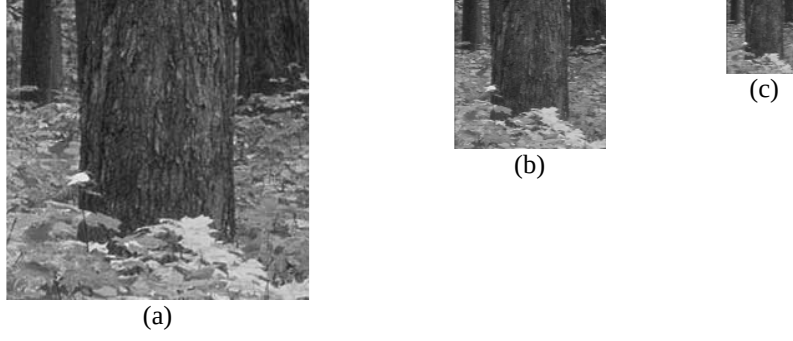
Şekil 5.22’de düşük çözünürlüklü görüntüden alınan her parça için veri tabanından eşleme yapılan aday parçaların seçilme olasılıkları görülmektedir.



Şekil 5.22 : Veri Tabanında Seçilen Tüm Aday Parçaların Olasılıkları

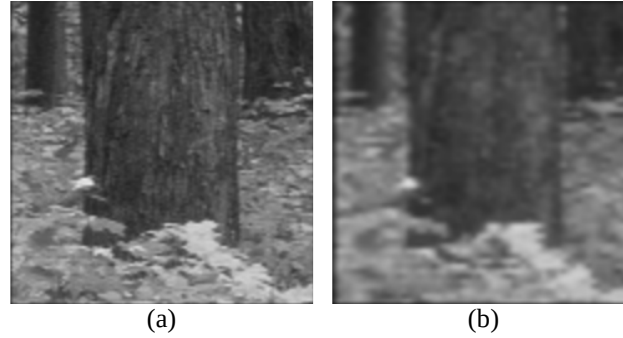
5.6 Örnekleme Sıklığının Değiştirilmesinin Yüksek Çözünürlüklü Görüntülerin Kestirimine Etkisi

Yüksek çözünürlüklü görüntü elde etmek için veri tabanını oluştururken yüksek çözünürlüklü görüntülerden örnekleme sıklığı değiştirilerek düşük çözünürlüklü görüntüler elde edilir. Bu işlem sırasında yüksek çözünürlüklü görüntü ilk önce yatayda ve dikeyde $\frac{1}{2}$ oranında küçültülür. Tekrar enterpolasyon ile eski boyutlarına getirilir. Örnekleme sıklığının küçültülme oranı algoritmanın başarısına etki etmektedir. Bu bölümde yatayda ve dikeyde $\frac{1}{2}$ ile $\frac{1}{4}$ oranında küçültmenin algoritmaya etkisi incelenecektir.



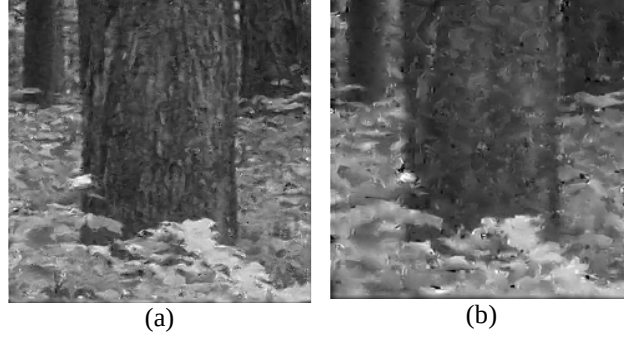
Şekil 5.23 : Örnekleme Sıklığı Değiştirilen Görüntüler (a) Yüksek Çözünürlüklü Görüntü, (b) 1: 2 Oranında Küçültülmüş Görüntü, (c) 1: 4 Oranında Küçültülmüş Görüntü

Şekil 5.23(a)'da yüksek çözünürlüklü giriş görüntüsü görülmektedir. Bu görüntü önce sırasıyla ayrı ayrı $\frac{1}{2}$ ve $\frac{1}{4}$ oranlarında küçültülerek şekil 5.23(b) ve şekil 5.23(c)'de görülen görüntüler elde edilmiştir. Küçültülmüş görüntüler enterpolasyon yapılarak orijinal boyutlarına getirilir. Orijinal boyutlarına gelen görüntüler artık içlerinde yüksek frekans bilgisini taşımamaktadırlar. Şekil 5.24'de enterpolasyon işlemi sonrası orijinal boyutuna getirilen görüntüler bulunmaktadır.



Şekil 5.24 : Enterpolasyon Kullanılarak Orijinal Boyuta Getirilen Görüntüler (a) 1: 2 Oranında Küçültülmüş ve Tekrar Orijinal Boyutuna Getirilmiş Görüntü, (b) 1: 4 Oranında Küçültülmüş ve Tekrar Orijinal Boyutuna Getirilmiş Görüntü

Bu görüntülere ayrı ayrı çözünürlük yükseltme işlemi uygulandıktan sonra kestirilen yüksek çözünürlüklü görüntüler şekil 5.25'de görülmektedir. Bütün testler aday parça sayısı 10, " $\alpha_s = 300$ " ve " $\alpha_i = 60$ " değerleri kullanılarak yapılmıştır. Ara basamaklarda kullanılan küçültme oranları arttıkça sonuçlar daha kötüleşmektedir.



Şekil 5.25 : Farklı Örnekleme Sıklığındaki Görüntüler Kullanılarak Kestirilen Yüksek Frekans Görüntüleri (a) Örnekleme Sıklığı 1: 2 Oranında Değiştirilmiş Giriş Görünüsü İçin Kestirilen Sonuç, (b) Örnekleme Sıklığı 1:4 Oranında Değiştirilmiş Giriş Görüntüsü İçin Kestirilen Sonuç

6. SONUÇLAR VE TARTIŞMA

Görüntüler kayıt edildikleri ortamdan daha yüksek çözünürlüklü bir ortamda gösterilmek istenirse büyütülmeleri gerekmektedir. Bu işlem için çeşitli enterpolasyon teknikleri kullanılır. Enterpolasyon düşük çözünürlüklü görüntülerde eksik olan detayları ekleyemez. Tez çalışması kapsamında incelenen yöntem, düşük çözünürlüklü görüntülerde eksilen detay bilgisinin kestiriminde kullanılabilecek öğrenmeye dayalı bir yöntemdir. Markov Ağları kullanılarak problem modellenmiş, olasılıksal yöntemler ile çözüme gidilmiştir. Önerilen bu yöntem önce sabit görüntüler üzerinde denenmiştir. Düşük çözünürlüklü görüntülerden algısal olarak kabul edilebilir yüksek çözünürlüklü görüntüler yaratılmıştır. İkinci bir yöntem olan Tek Geçiş Algoritması kullanılarak aynı başarımla elde edilmeye çalışılmıştır. Markov Ağları kullanılarak bulunan sonuçlar Tek Geçiş Algoritması ile bulunan sonuçlara göre daha fazla yüksek frekans bilgisi içermektedir. Tek Geçiş Algoritması ise daha hızlı çalışmaktadır.

Bu aşamadan sonra geliştirilen algoritma video kliplerine uygulanmış, düşük çözünürlüklü videolar kullanılarak karşılık düşen yüksek çözünürlüklü videolar elde edilmeye çalışılmıştır. Videolar üzerinde işlem yapılırken çerçeveler ayrı ayrı çözünürlük yükseltme işlemine tabi tutulmuştur. Hareket bilgisi bu hesaplamalara dahil edilmemiştir. Ortaya çıkan sonuçlarda video çerçeveleri içinde bulunan sabit nesneler için algısal olarak kabul edilebilir yüksek çözünürlüklü karşılıkları görülürken, hareket eden nesneler üzerinde kırışımlar görülmektedir.

Markov ağları kullanarak yüksek çözünürlüklü görüntü üretme işleminin başarımında önemli bir etken yüksek ve düşük çözünürlüklü ikililerin saklandığı veri tabanı içinde yüksek çözünürlüklü bol detay içeren örneklerin kullanılmasıdır. Öte yandan veri tabanı oluşturmak için kullanılan video çerçevelerin, çözünürlüğü yükseltmek istenilen video çerçevelerine benzerliği arttıkça algoritmanın başarımı da artmaktadır.

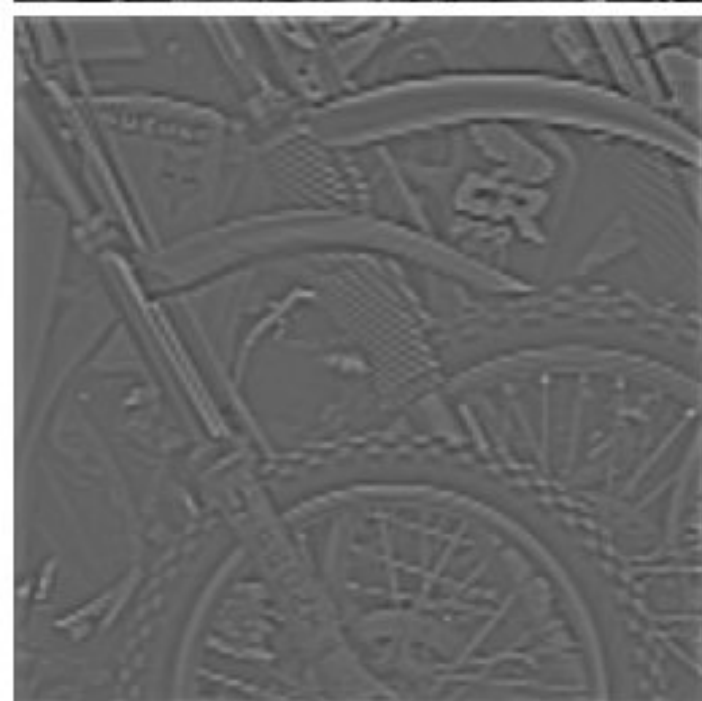
KAYNAKLAR

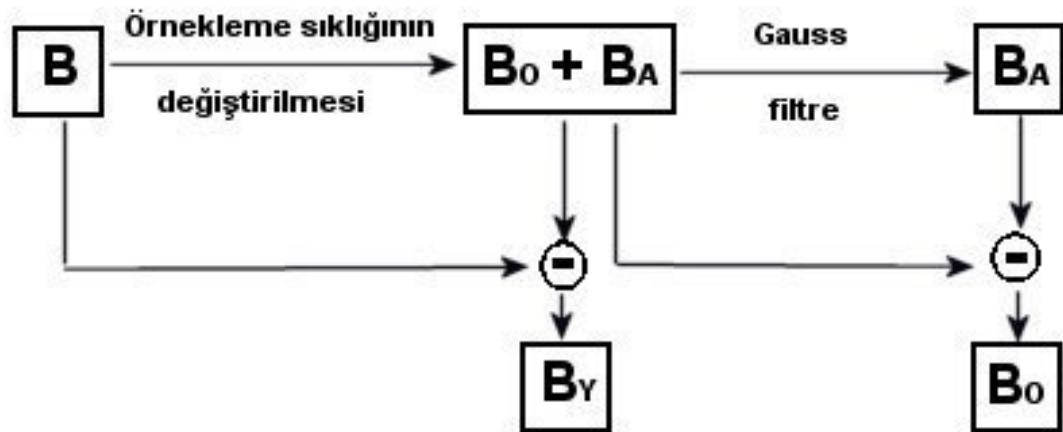
- [1] **Freeman W. T., Pasztor E. C. and Carmichael O. T.,** 2000. Learning low-level vision. *Intl. J. Computer Vision*, 40:24-47
- [2] **Freeman W. T., Jones T. R. and Pasztor E. C.,** 2001. Example-based super resolution. *MERL*, TR-2001-30
- [3] **C. M. Bishop, Blake A. and Marthi B.,** 2002. Super-resolution Enhancement of Video Microsoft Research, Cambridge, CB1 2 HN, UK
- [4] **Hong L.,** 2000 *Super Resolution from Low Resolution Videos* ST Journal of System Research
- [5] **Freeman W. T. and Pasztor E. C.,** 1999. Learning to estimate scenes from images. In M. S. Keaarns, S.A. Solla, and D. A. Cohn, editors *Adv Neural Information Processing Systems*, volume 11, Cambridge, MA, 1999, MIT Press
- [6] **Geman S. and Geman D.,** 1984. Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images. *IEEE Pattern Analysis and Machine Intelligence*, 6:721-741
- [7] **Bell A. J. and Senjowski T. J.,** 1997. The independent components of natural scenes are edge filters. *Vision Research*, 37(23):3327-3338
- [8] **Kersten D., O'Toole A. J., Sereno M. E., Knill D. C. and Anderson J. A.,** 1987. Associative learning of scene parameters from images. *Applied Optics*, 26(23):4999-5006
- [9] **Szeliski R.,** *Bayesian Modeling of Uncertainty in Low-level Vision.*, 1989. Kluwer Academic Publishers, Boston
- [10] **Frey. B. J.,** 1997. *Bayesian networks for pattern classification.* MIT Press

- [11] **Freeman W. T. and Pasztor E. C.**, 2000. "Markov Networks for Superresolution," *Proc. 34th Ann. Conf. Information Sciences and Systems (CISS 2000)*, Dept. Electrical Eng., Princeton Univ
- [12] **Baker S. and Kanade T.**, 2000. "Limits on Super-Resolution and How to Break Them," *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, vol. II, IEEE CS Press, Los Alamitos, Calif., pp. 372-379.
- [13] **Yedidia J.S., Freeman W. T. and Weiss Y.**, 2001. "Generalized Belief Propagation," *Advances in Neural Information Processing Systems*, T.K. Leen, T.G. Dietterich, and V. Tresp, eds., vol. 13, MIT Press, Cambridge, Mass., pp. 689-695
- [14] **Fekri F., Mersereau R. M. and Schafer R. W.**, 1998. A generalized interpolative VQ method for jointly optimal quantization and interpolation of images. In *Proc. ICASSP*, volume 5, pages 2657–2660
- [15] **Nene S. A. and Nayar S. K.**, 1997, A simple algorithm for nearest neighbor search in high dimensions. *IEEE Pattern Analysis and Machine Intelligence*, 19(9):989–1003, September
- [16] **Schultz R. R. and Stevenson R. L.**, 1994, A Bayesian approach to image expansion for improved definition. *IEEE Trans. Image Processing*, 3(3):233–242
- [17] **Schultz R. R. and Stevenson R. L.**, 1994, Extraction of HighResolution frames from video sequences. *IEEE Trans. Image Processing*, 5:996-1011, June
- [18] **Schreiber W. F.**, 1986, Fundamentals of Electronic Imaging Systems, Springer-Verlag
- [19] **Eruhimov V. and Bradski G.**, 2003, Probabilistic Network Library, Intel Russia Research Center

ÖZGEÇMİŞ

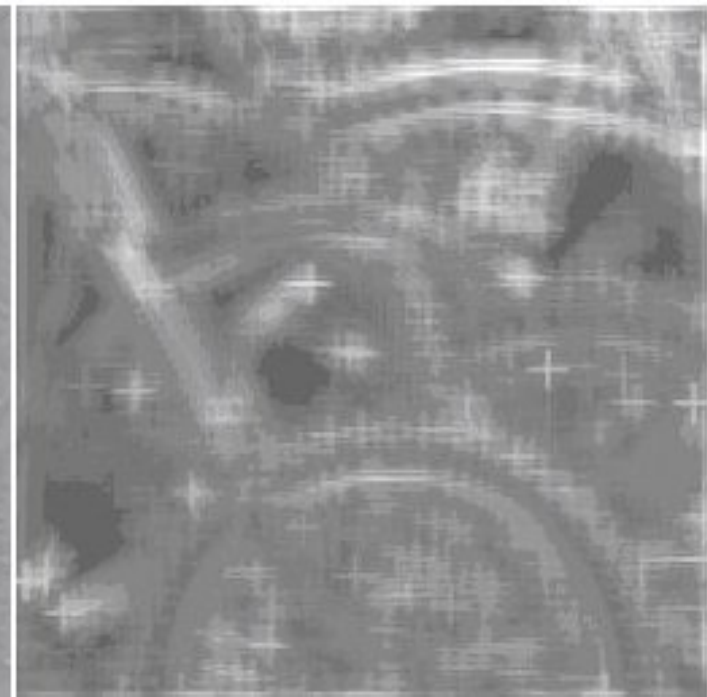
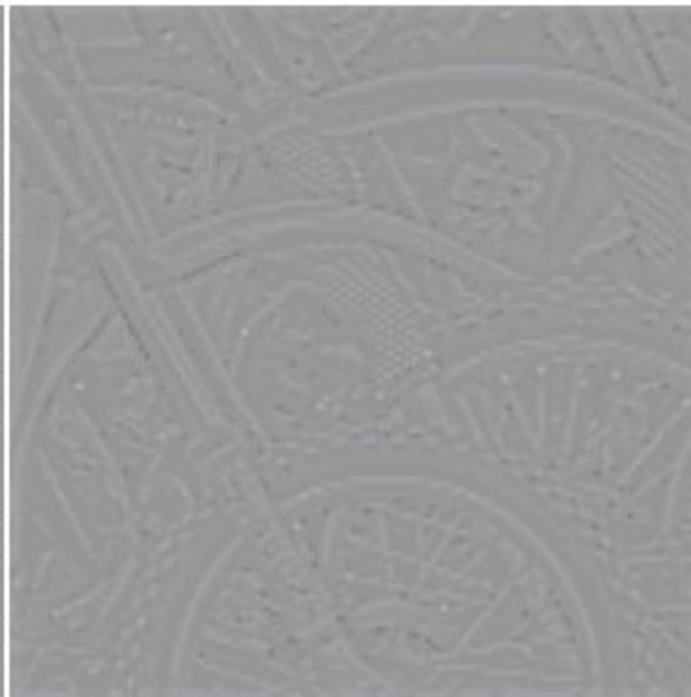
Bora TUNÇER, 1981 yılında İstanbul'da doğdu. 1999 yılında Vatan Anadolu Lisesi'nden mezun oldu. 2004 yılında İstanbul Üniversitesi Elektrik-Elektronik Mühendisliği Bölümünü bitirerek mühendis ünvanını aldı. Daha önce 2002 – 2004 yılları arasında Pasifik Yazılım Şirketi'nde gömülü sistemler için paralel işlem yapan işletim sistemleri tasarladı. 2004 yılından beri STMicroelectronics firmasında yonga test mühendisi olarak çalışmaktadır.

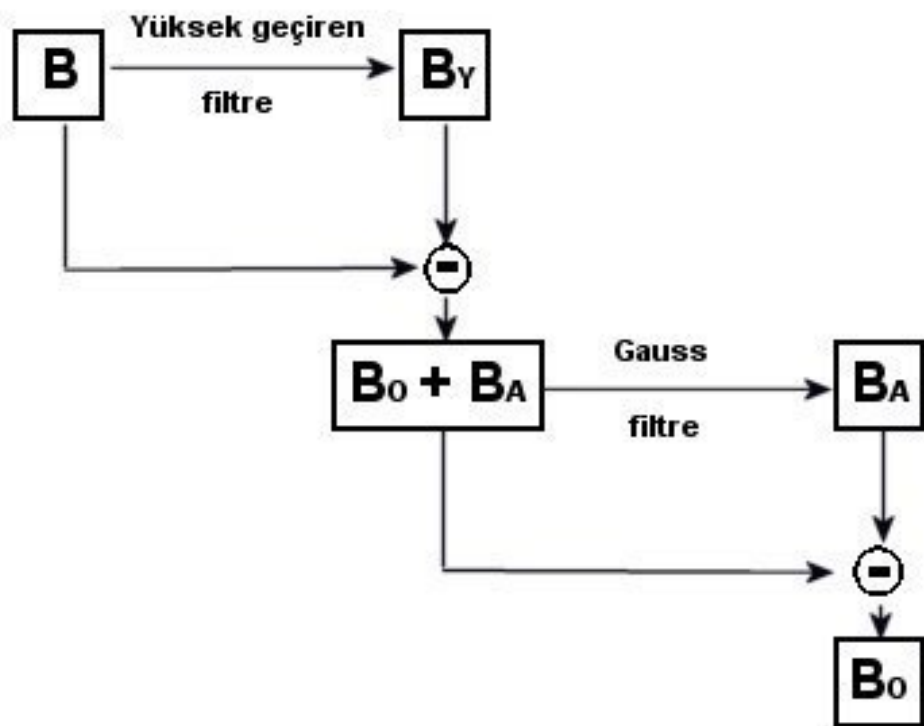


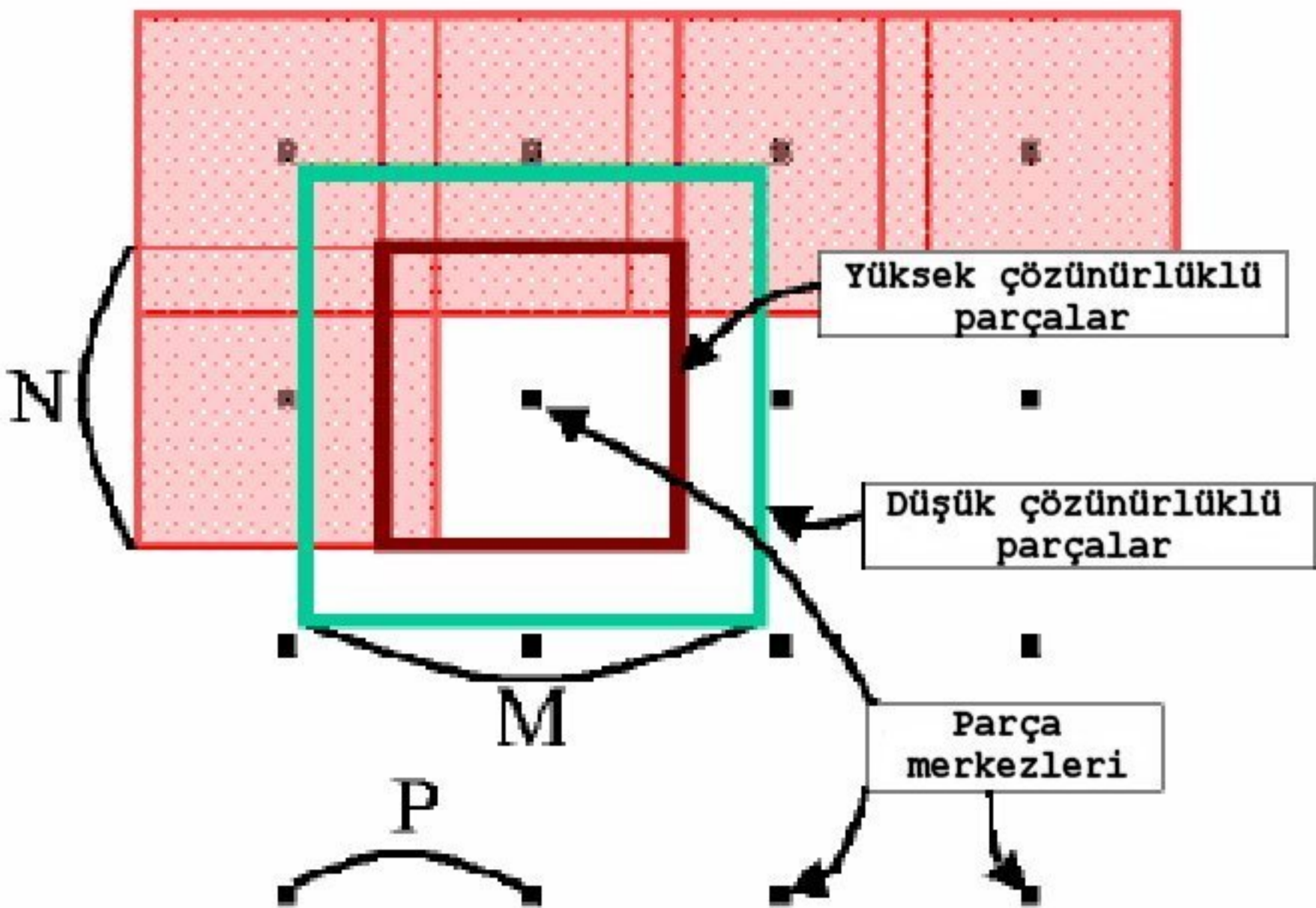


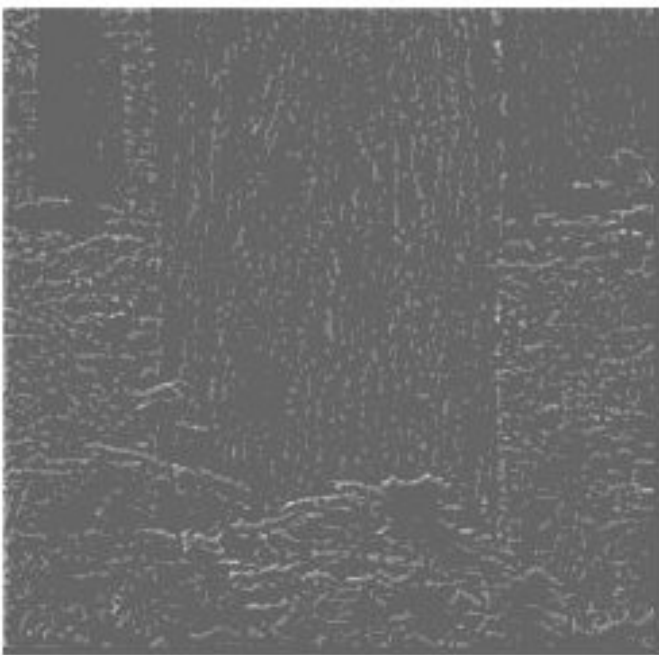
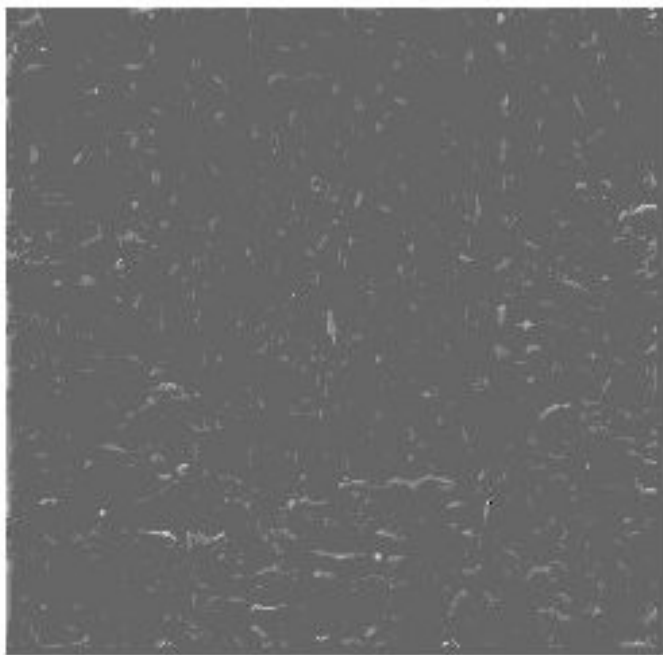


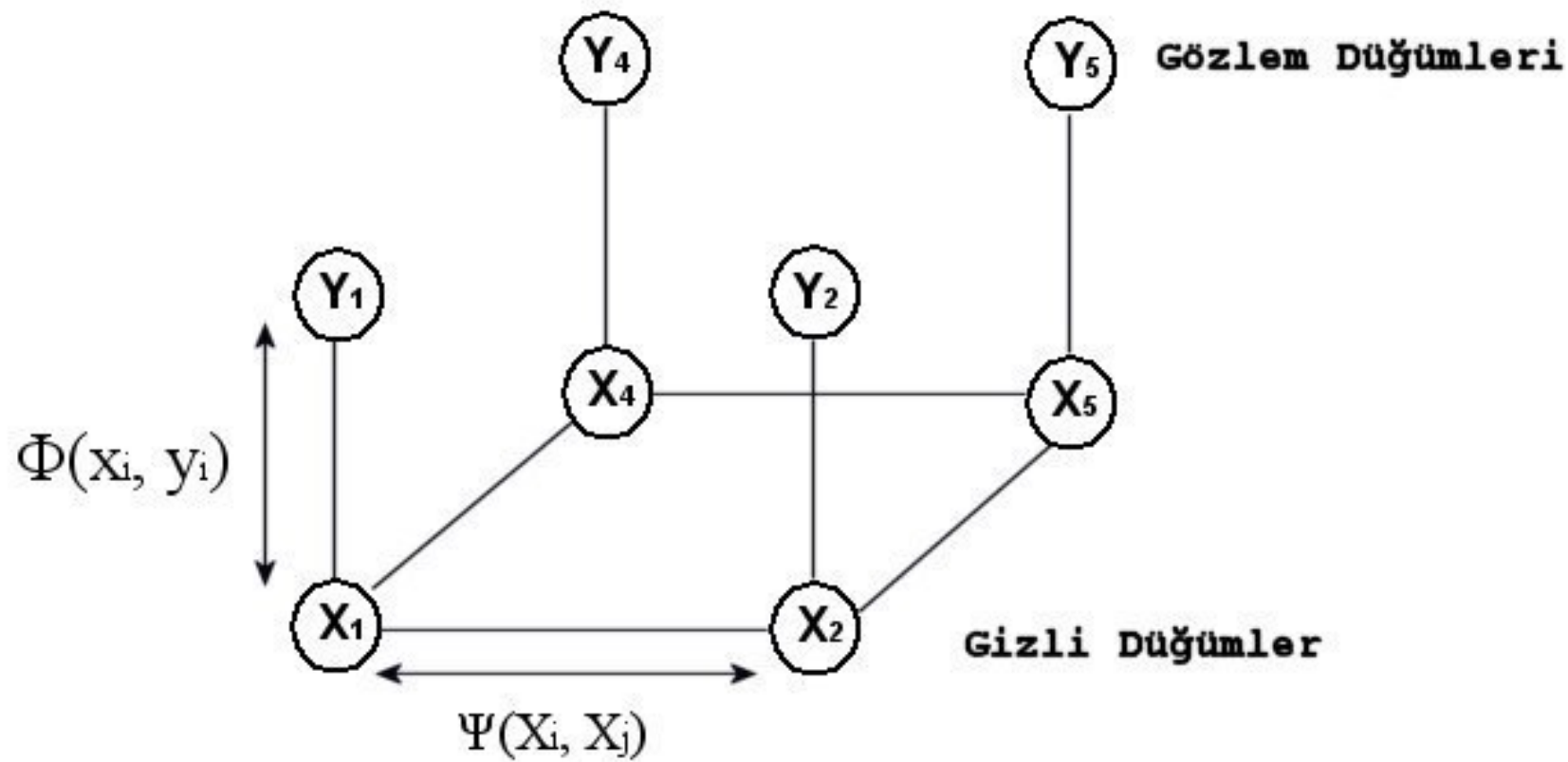


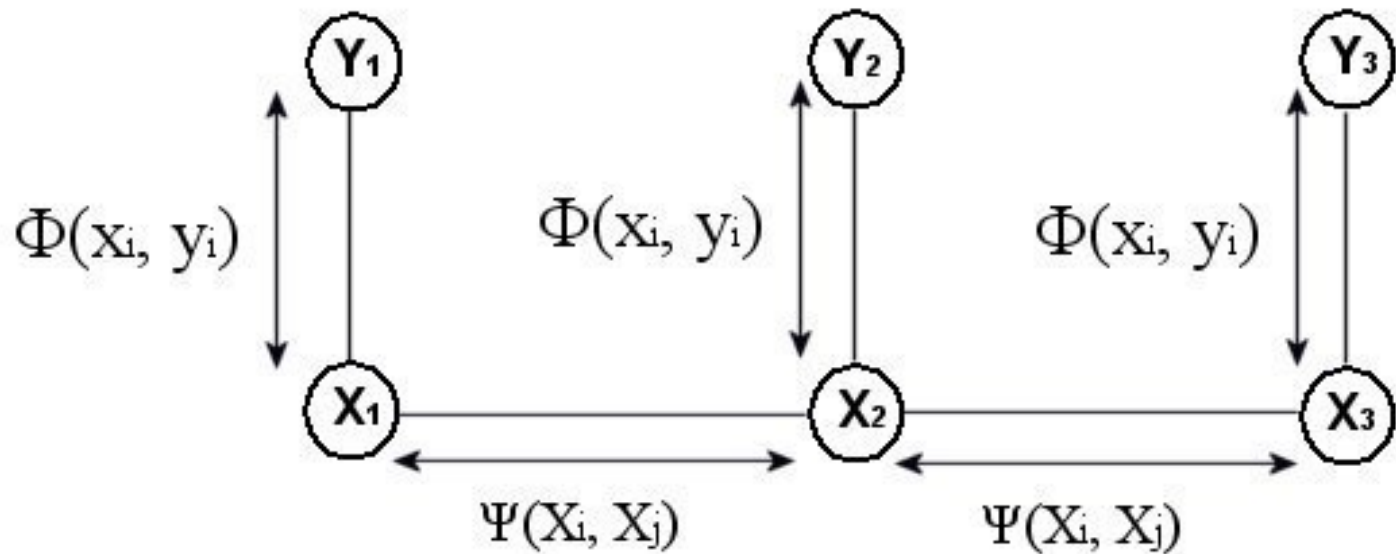


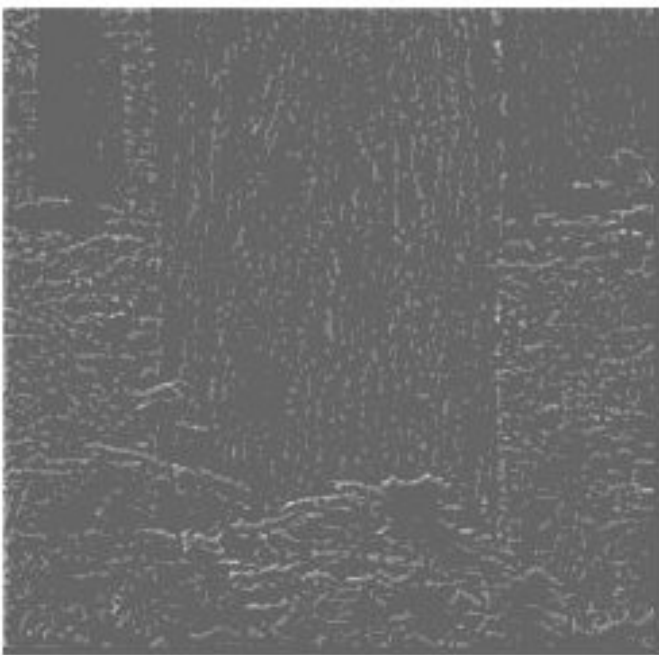
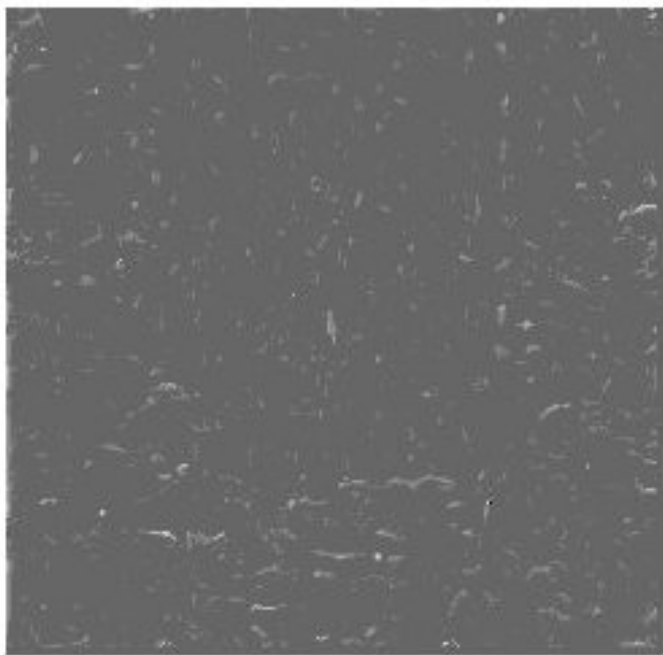


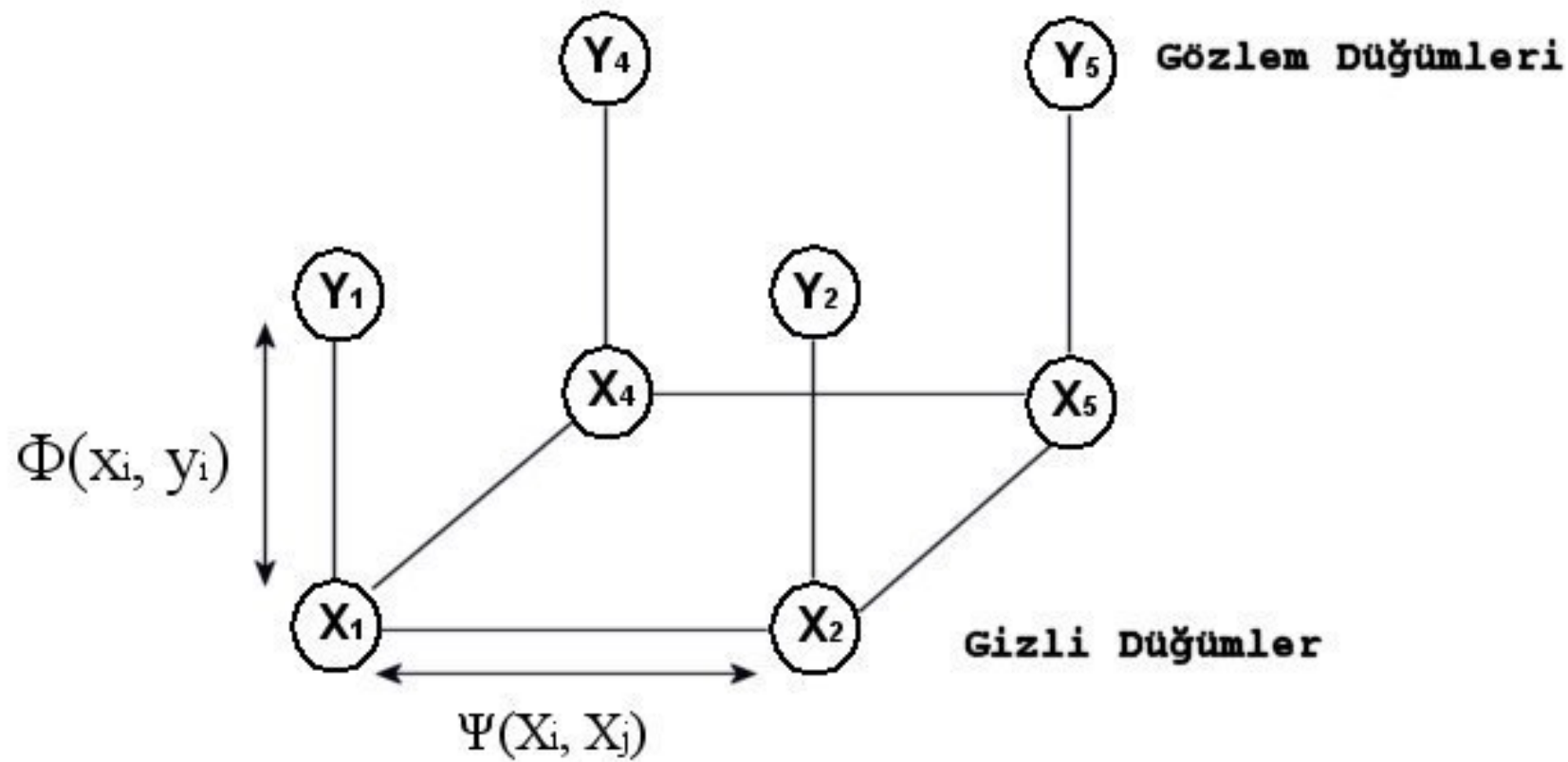


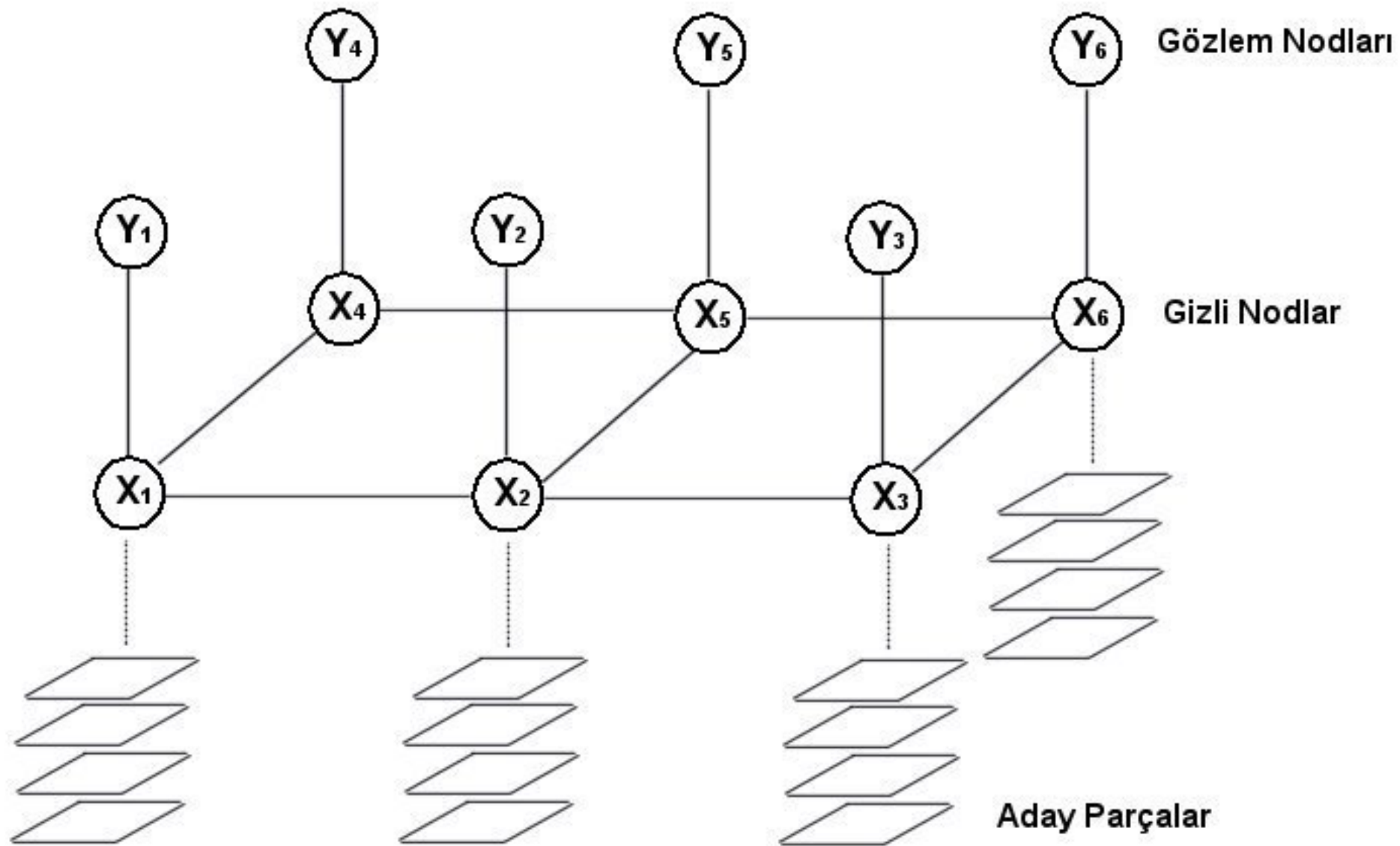






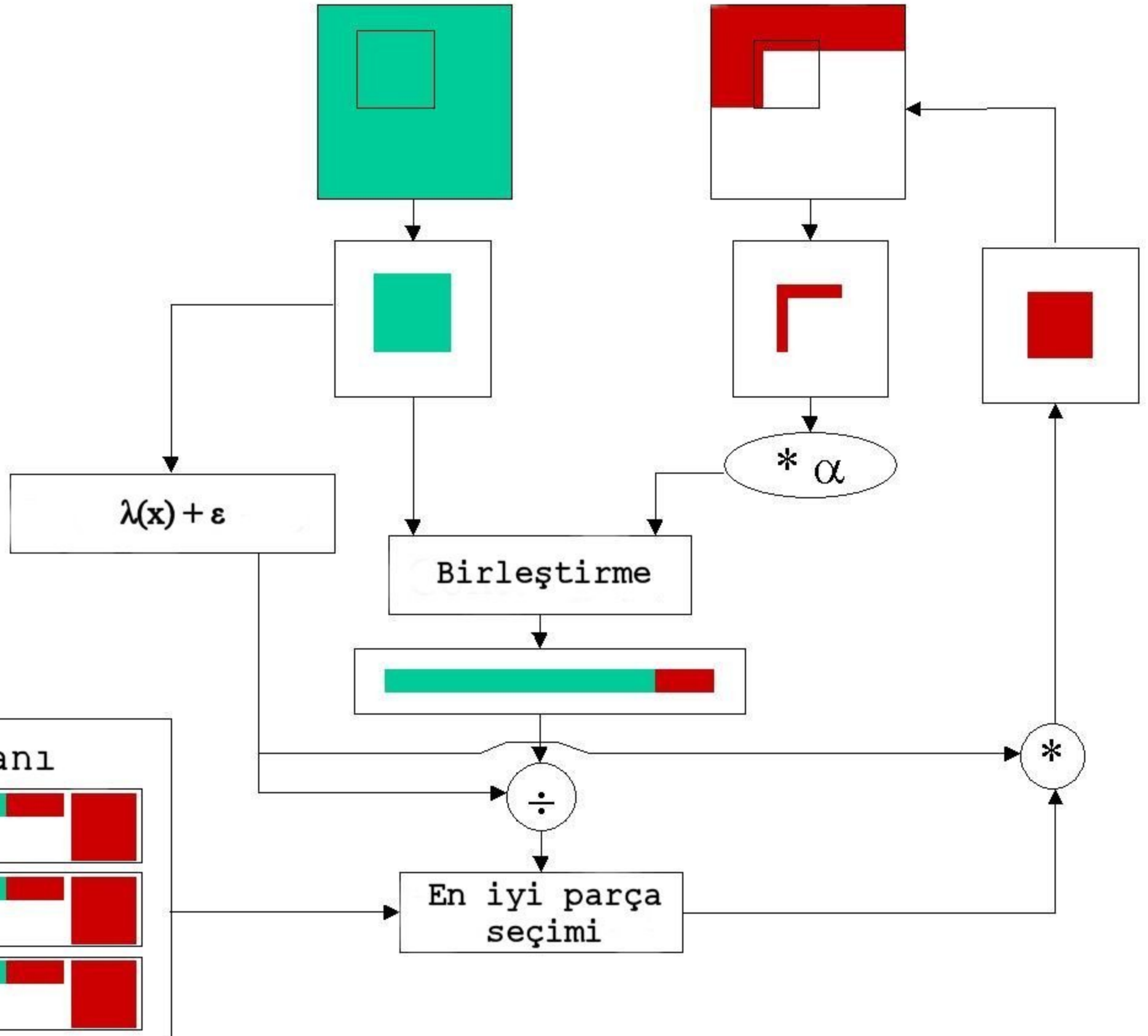


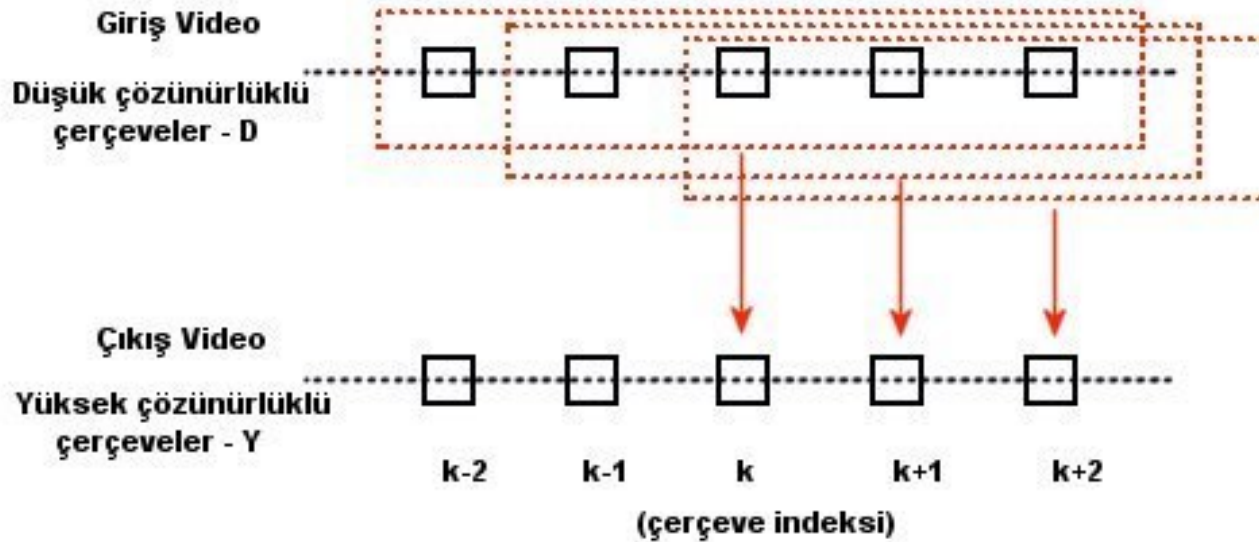


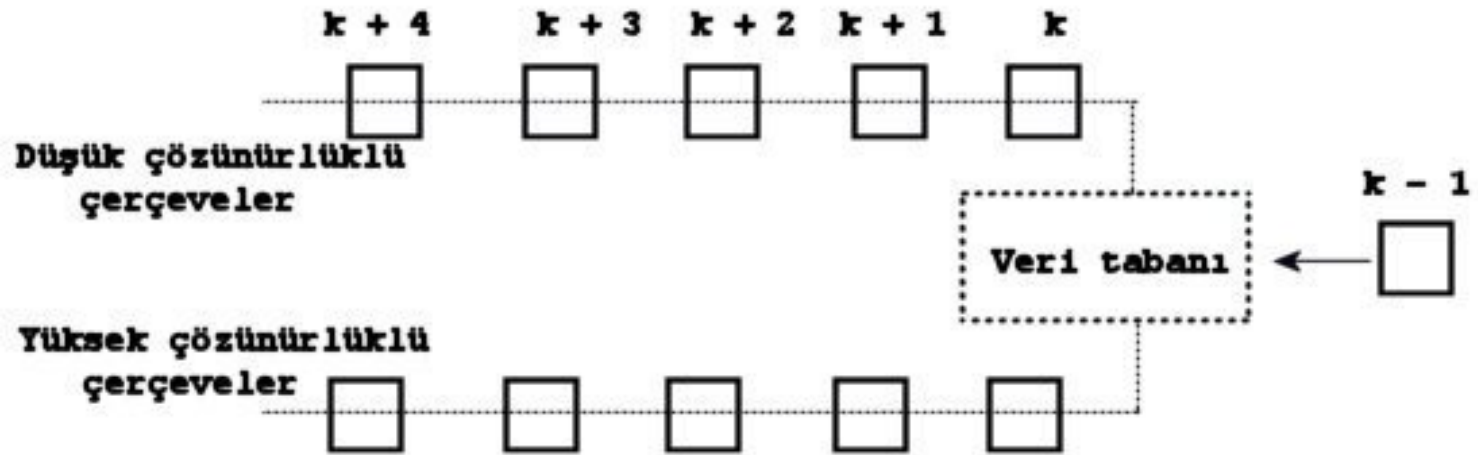


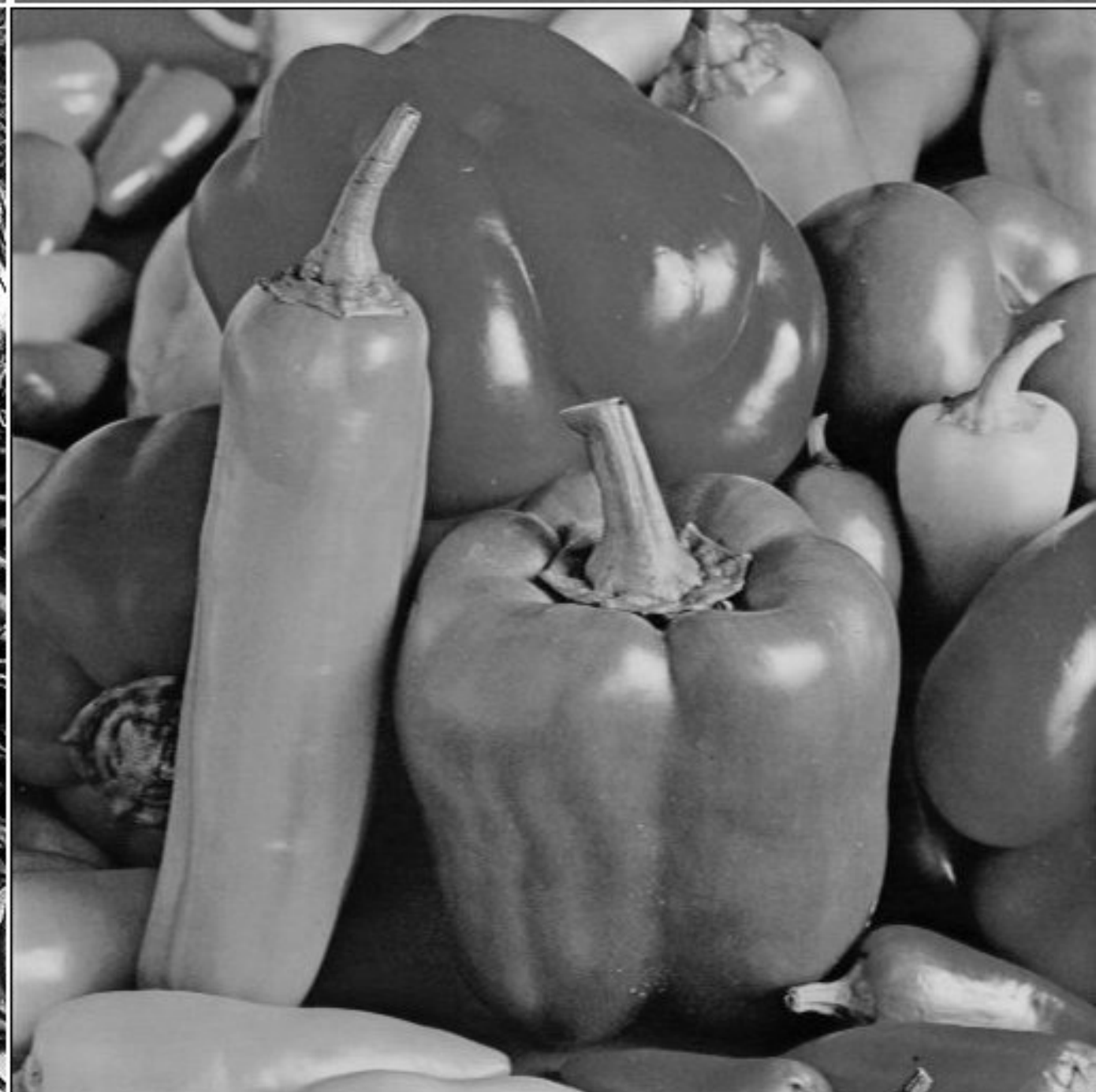
Giriş

Yüksek Frekans







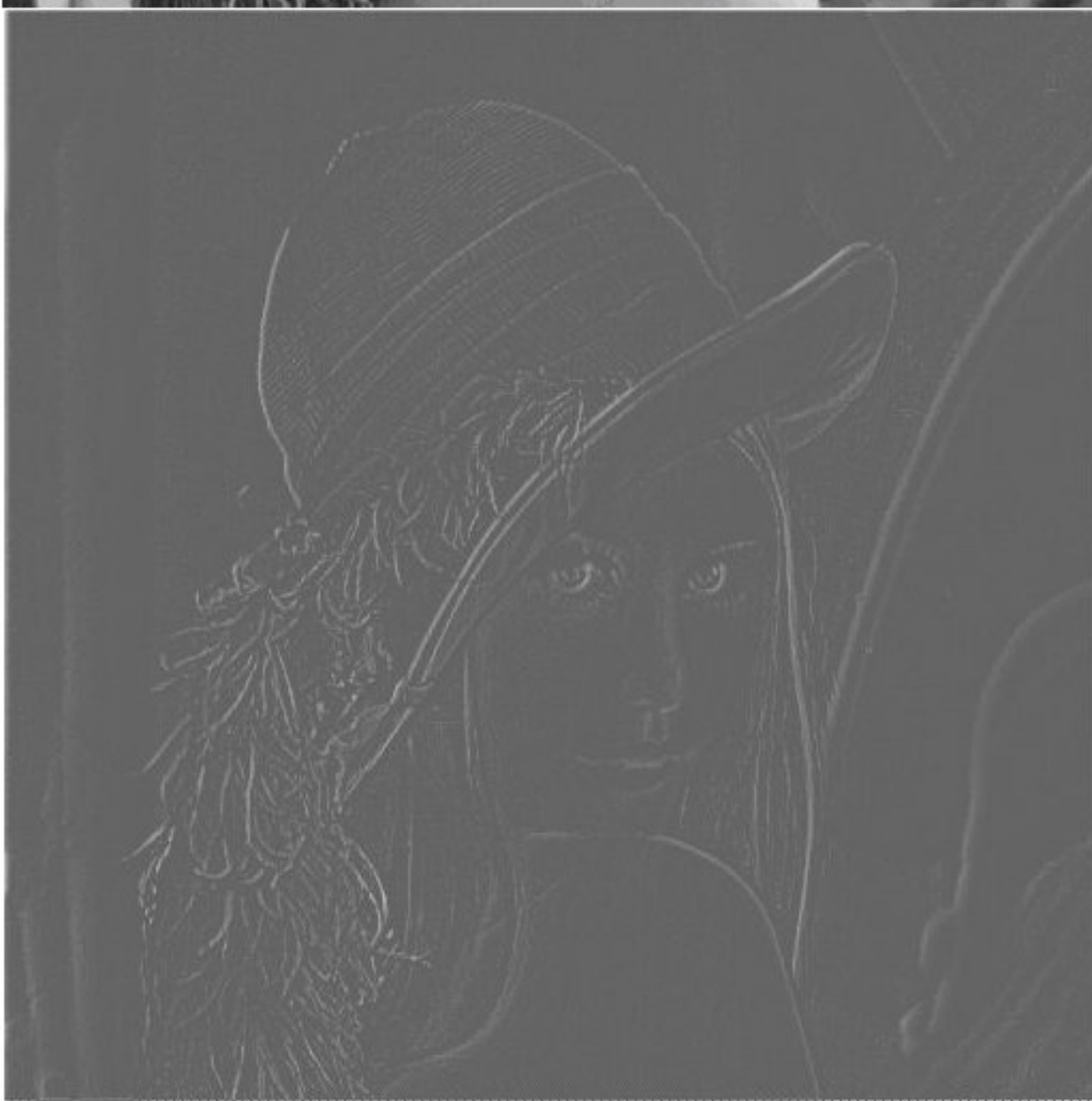


















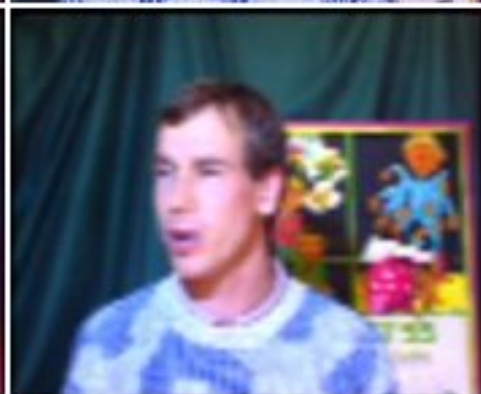
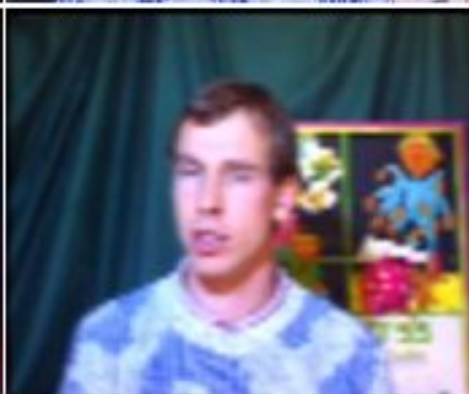
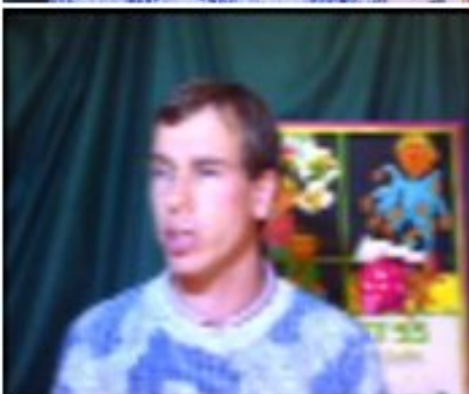


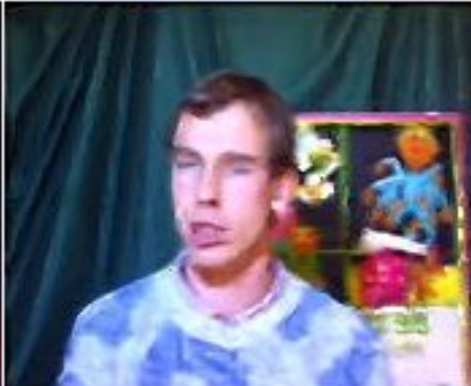


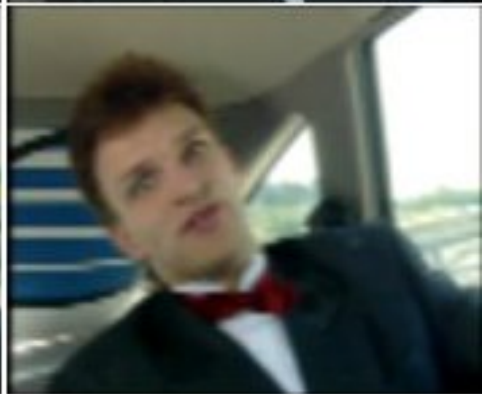
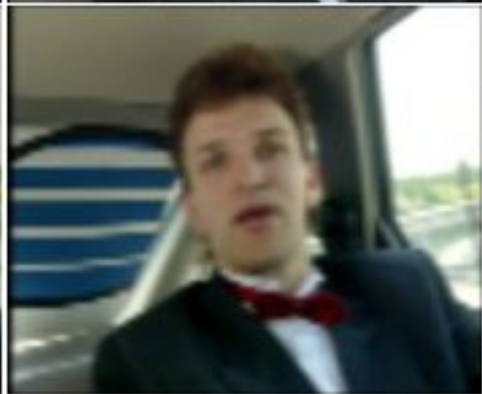


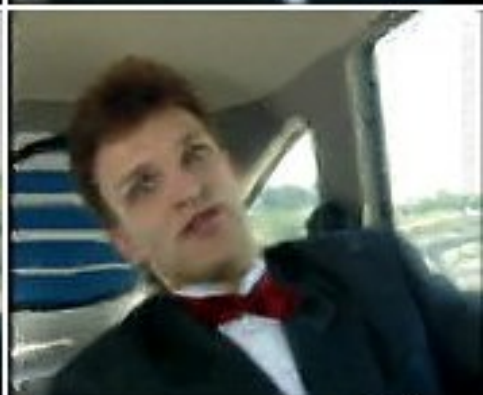
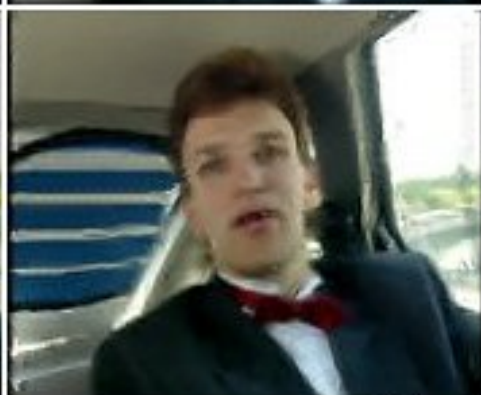
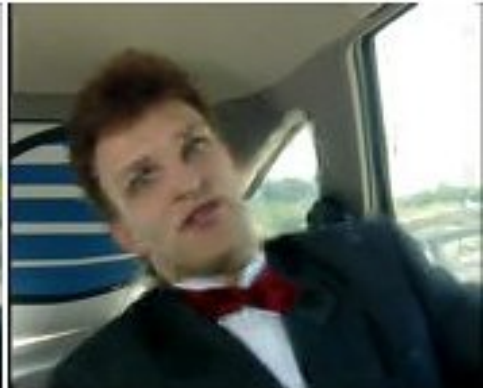


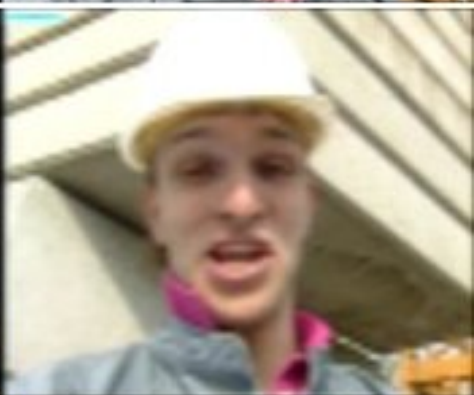
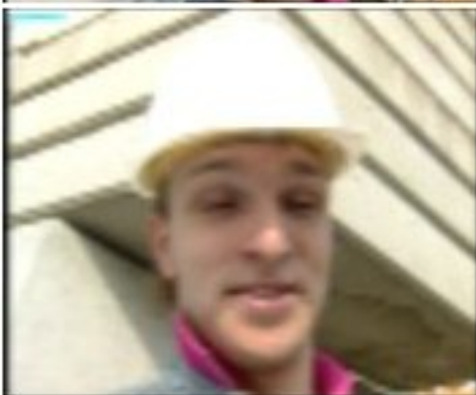


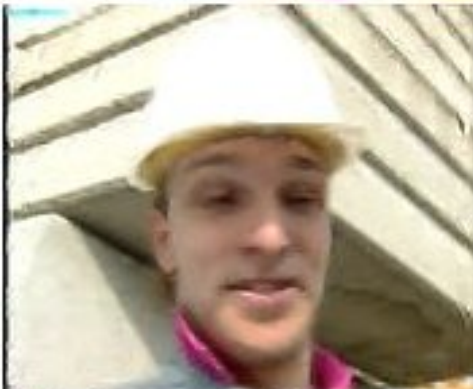






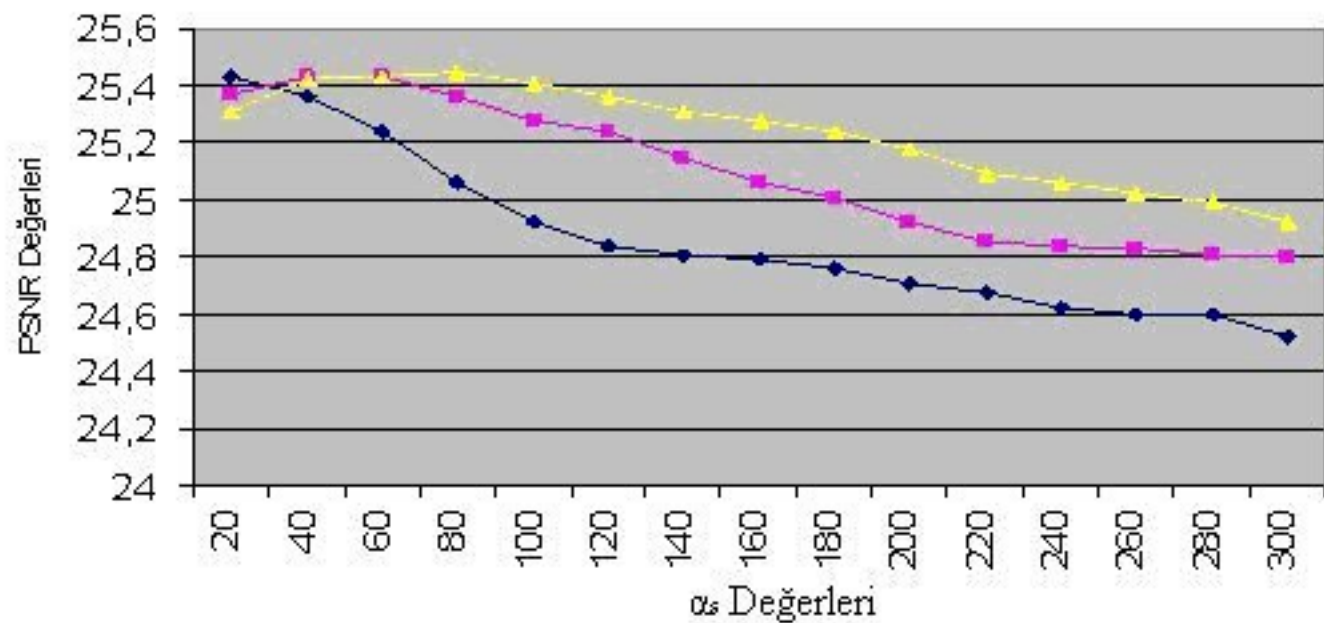


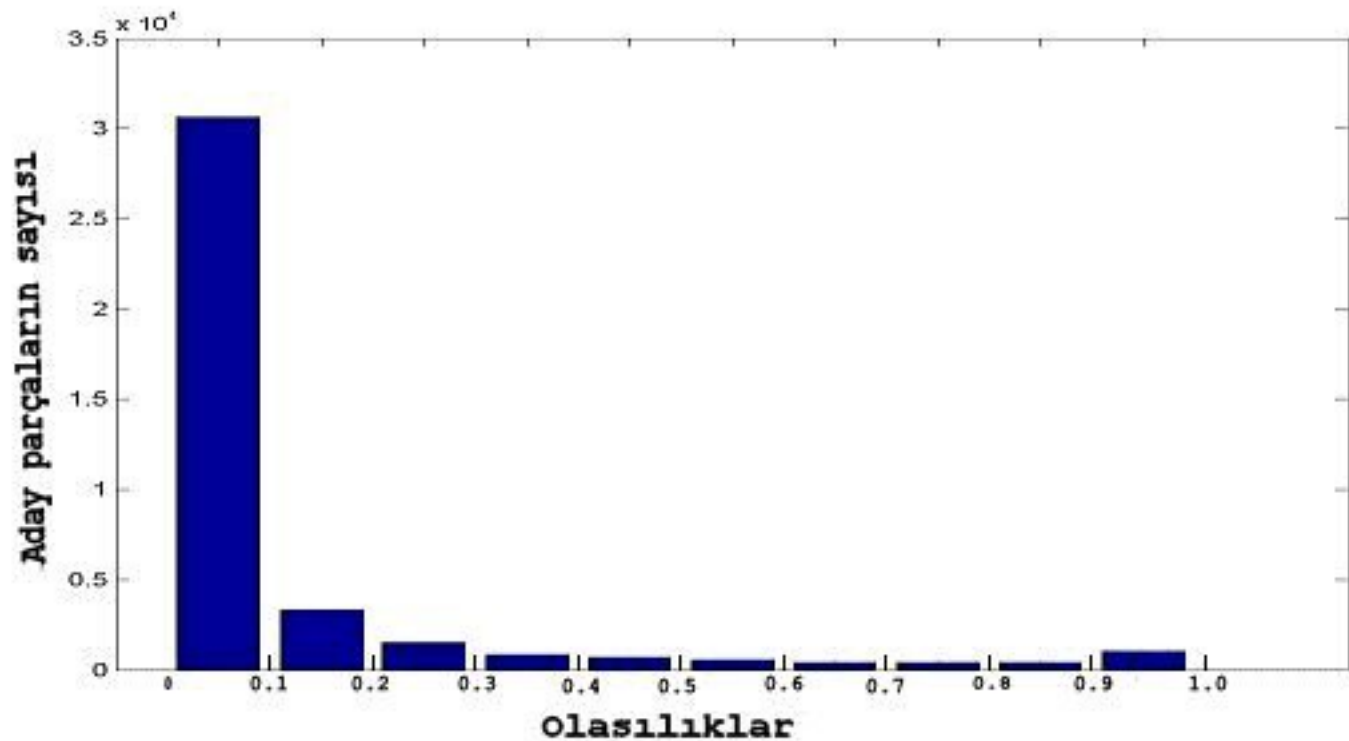


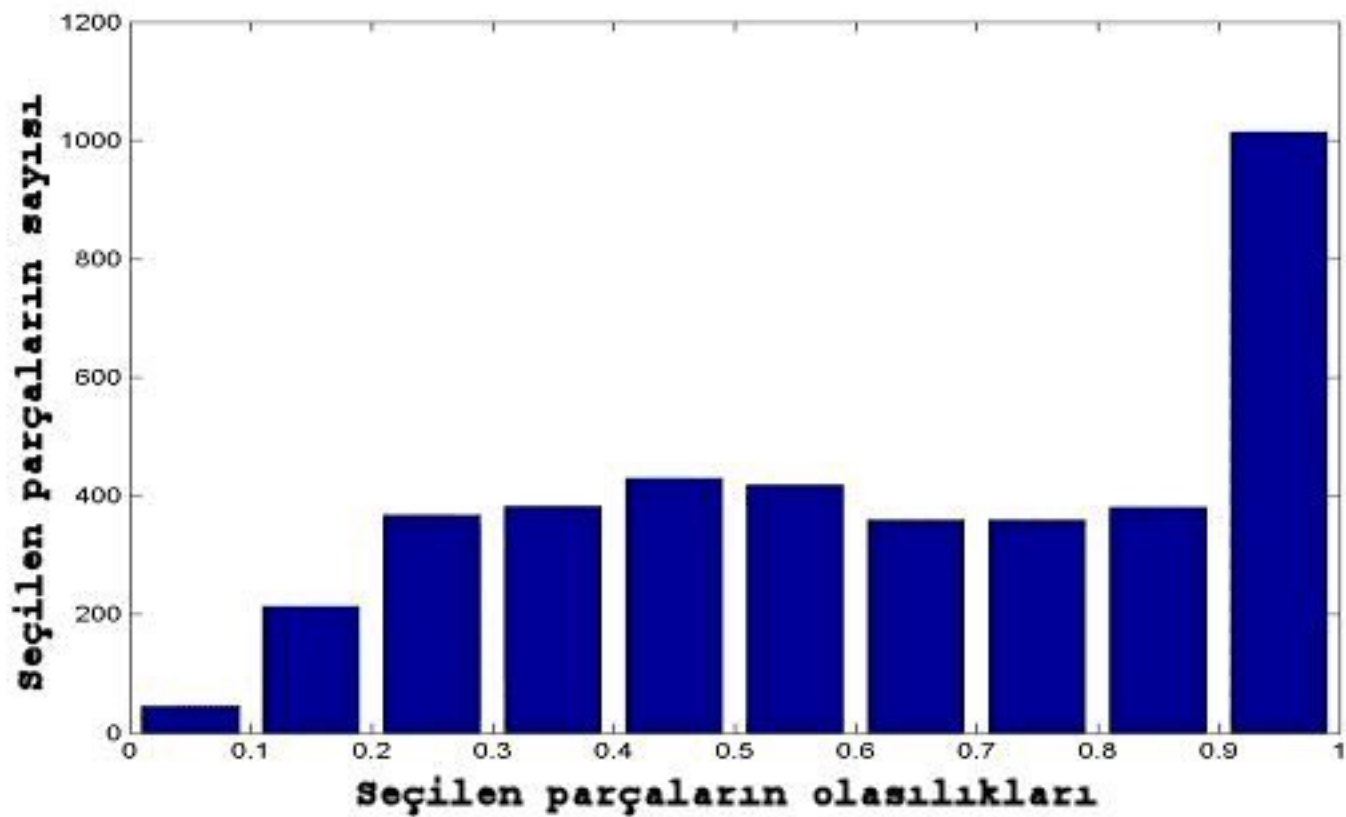


α_i ve α_s 'e bağı PSNR değerleri

—●— $\alpha_i = 100$ —■— $\alpha_i = 200$ —▲— $\alpha_i = 300$







Aday Parçaların Sayısının Olasılığa Etkisi

—◆— C=4 —■— C=6 —▲— C=8 —×— C=10

