

## Turbo kodlamalı resimlerin aşamalı iletimi ve resim bağışımı sıkıştırılması

**Kenan BÜYÜKATAK\***, Sedef KENT, O. Nuri UÇAN

*İTÜ Elektrik-Elektronik Fakültesi, Elektronik ve Haberleşme Mühendisliği Bölümü, 34469, Ayazağa, İstanbul*

### Özet

*Bu çalışmada, haberleşme kanalında turbo kodlamalı olarak iletilen görüntü kalitesinin alıcıda aşamalı bir biçimde ayarlanabilmesine olanak sağlayan bir metot önerilmiştir. Görüntü önce maskelenmiş sonra her bir maske belli sayıda bit düzlemleriyle ifade edilmiş, kodlanmış daha sonra da iletim kanalına verilmiştir. Maskelere ait bit düzlemi sayısı görüntüye ait detaylar tarafından belirlenmiştir. İstenen görüntü kalitesinin aşamalı olarak elde edilebilmesi amacıyla, verici, görüntüye ait bilgileri kodlanmış olarak göndermeye devam eder. Alıcıya bu bilgiler aşamalı olarak geldikçe En Büyük sonsal algoritması yardımıyla kod çözme işlemi yapılır ve bit düzlemleri yeniden oluşturulur. Böylece görüntü detaylarının (www tarayıcılarında olduğu gibi) ard arda elde edilmesi sağlanmış olur.*

**Anahtar Kelimeler:** Turbo kodlama, bit düzlemi, aşamalı resim iletimi, en büyük sonsal.

### Progressive transmission and image oriented compression of turbo coded images Abstract

*In this chapter, we propose a method, that allows turbo coded images to transfer efficiently, in which the data amount and the quality passed to the receiver can be adjusted. The procedure is based on the idea that only partial information from the original image is provided to the receiver at each step. The receiver will use the limited information and perform the enhancement. At first, the image is partitioned (masked), bit sliced, turbo coded and transmitted progressively through the communication channel. The number of bit planes of each mask is determined according to the details. In order to extract the quality of the image, the transmitter sends the coded bit planes of each mask continuously. Since the coded planes are collected at the receiver progressively, they are decoded via MAP (Maximum A posteriori Probability) algorithm reconstructing the bit planes and then the masks. So the whole image is built at the receiver. By this way, the details or additional information about the image is updated (like www browsers) at each step. So, the new data, together with data already on the receiver side, is used to build a new and more complete image at the receiver, achieving a better image representation. This procedure is iterated, according to the need of quality satisfaction.*

**Keywords:** Turbo coding, bit plane, progressive image transmission, maximum a posteriori probability.

---

\*Yazışmaların yapılacağı yazar: Kenan BÜYÜKATAK. k.buyukatak@hho.edu.tr; Tel: (212) 663 24 90 dahili: 4345.

Bu makale, birinci yazar tarafından İTÜ Elektrik-Elektronik Fakültesi'nde tamamlanmış olan "Transmission of 2-D images for turbo codes" adlı doktora tezinden hazırlanmıştır. Makale metni 04.08.2005 tarihinde dergiye ulaşmış, 14.10.2005 tarihinde basım kararı alınmıştır. Makale ile ilgili tartışmalar 30.09.2006 tarihine kadar dergiye gönderilmedi.

## Giriş

Bu çalışmada, iki boyutlu görüntü işaretlerinin az veri kullanılarak, internet ortamında olduğu gibi aşamalı olarak ve hata başarımı yüksek olacak şekilde iletilmesi için yeni bir yöntem geliştirilmiştir (Sherwood ve Zeger 1998). İletişim esnasında haberleşme kanalında oluşan istenmeyen etkilerin (Beyaz gürültü, AWGN) yok edilmesinde Turbo kodlama tekniği kullanılmış, böylece hatasız haberleşmenin önemli olduğu, band genişliğinin sınırlı olduğu ve hızlı haberleşmenin gerekli olduğu durumlarda (Naguib vd., 2000) görüntü iletimi gerçekleştirilebilmiştir (Büyükkatak vd., 2003).

Bu çalışmada, “İsteğe bağlı detay”, “Hata düzeltme” ve “Aşamalı iletim” olacak şekilde üç amacın gerçekleştirilmesi hedeflenmiştir. İlk aşamada, görüntüye ait ana hatların iletilmesinin yanında alıcının detay talebine göre bit düzlemi tahsis işlemi, ikinci aşamada turbo kodlama ve En Büyük Sonsal (EBS) algoritması yardımıyla hatanın azaltılması, son aşamada ise detay bilgilerinin ard arda gönderilmesi ve alıcıda görüntünün aşamalı olarak elde edilmesi işlemi gerçekleştirmiştir. Kayıplı veya kayıpsız iletimin söz konusu olabildiği bu metot da, kayıpsız iletimin elde edilebilmesi için alıcı tarafından tüm detay bilgilerinin talep edilmesi ve görüntüye, gönderilmeden önce maksimum sayıda bit düzlemi tahsis edilmesi gerekmektedir.

İletilmesi için daha fazla zamana ihtiyaç olması anlamına gelen bu durumda gürültünün belirli seviyede olmasıyla görüntünün tam olarak yeniden elde edilmesi mümkün olabilmektedir

## Sistem modeli

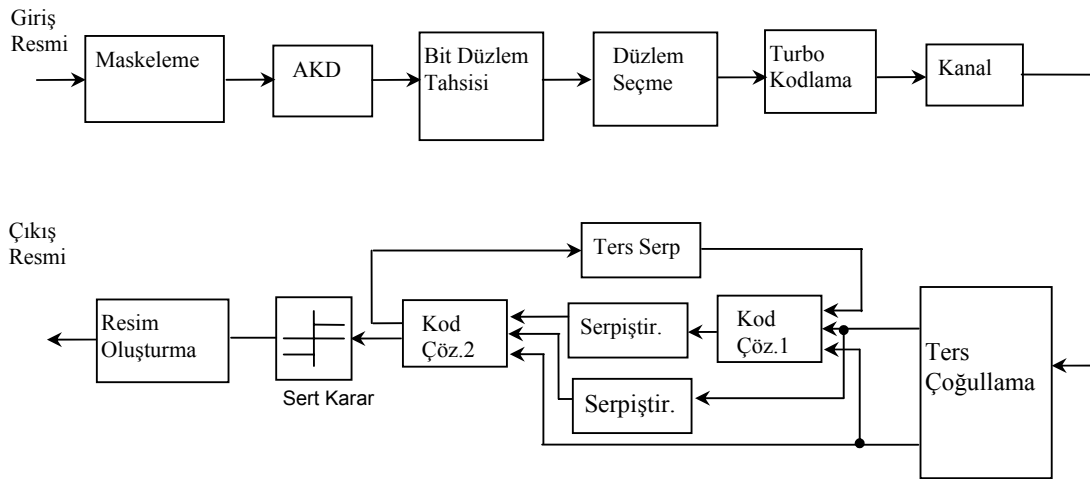
Sistem, Şekil 1’de görüldüğü gibi, Maskeleye, Ayrık Kosinüs Dönüştürme (AKD), Bit Düzlem Tahsisi, Düzlem Seçme, Turbo Kodlama, Turbo Kod Çözme ve Resim Oluşturma bölümlerinden oluşmaktadır.

## Maskeleye

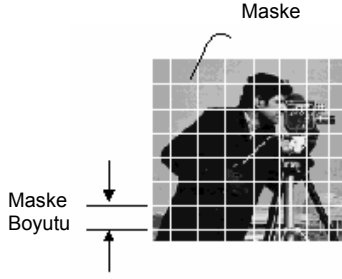
Maskeleye bölümünde resmin küçük bölümlere (alt resimlere) bölünmesi işlemi gerçekleştirilir. Alt resmin boyutları istenildiği gibi seçilebilir. En az 2X2 seçilebileceği gibi en fazla orijinal resmin boyutu kadar olabilir. Şekil 2, maskeleyenin nasıl yapıldığını ve resmin alt resimlere nasıl bölündüğünü göstermektedir.

Orijinal görüntü matrisel olarak  $[I(i, j)]$  ile gösterilebilir. Yani;

$$[I(i, j)]_{UXV} = \begin{bmatrix} I_{1,1}(i, j) & I_{1,2}(i, j) & \dots & I_{1,r}(i, j) \\ I_{2,1}(i, j) & \dots & \dots & \dots \\ \vdots & \dots & \dots & \dots \\ I_{p,1}(i, j) & \dots & \dots & I_{p,r}(i, j) \end{bmatrix}_{UXV} \quad (1)$$



Şekil 1. Sistem modeli



Şekil 2. Resmin maskelenmesi

Bu durumda  $I_{x,y}(i, j)$ , ( $x = 1..p, y = 1..r$ ) ifadesi, her bir maskelenmiş alt görüntüyü ifade edecektir. Alt görüntünün boyutunun  $L \times L$  seçildiği düşünülürse,  $p = U/L$  ve  $r = V/L$  olacaktır. Elde edilen her maskeye ilişkin AKD değerleri, “Ayrık Kosinüs Dönüştürme” bölümünde gerçekleştirilir. Bu değerlerin genliklerinin büyüklüklerine göre bit tahsis işlemi gerçekleştirilir. AKD değerleri ne kadar büyükse resmin o bölümünde o kadar fazla detay var demektir ve o bölüme daha fazla sayıda bit düzlemi tahsis edilir. Böylece daha az detaya sahip bölgeler daha az sayıda bit düzlemleriyle ifade edilecek ve o bölge daha fazla sıkıştırılmış olacaktır.

### Bit düzlem tahsisi

Gri-seviyeli bir resmin tüm pikselleri ikili sistemde  $n$  bitle ifade edilebilir (Gonzales ve Woods, 1992). Her pikselin en düşük anlamlı bitinden en yüksek anlamlıya doğru ayrıştırılmasıyla o resme ait  $n$  adet ikili (binary) resim elde edilir. En fazla komşuluk ilişkisine en yüksek anlamlı ikili resim sahip olmakla birlikte, anlam mertebesi azaldıkça bu ilişki de zayıflayacaktır. Bu işleme, “Resmin Bit Düzlemlerine ayrıştırılması” işlemi denir (Büyükatak vd., 2004). Eğer resim  $n$  bitle ifade edilmişse o resme ait  $N$  adet ikili resim elde edilir. Şekil 3, gönderilmesi istenen örnek resim için  $N=4$  durumunda ayrıştırılan bit düzlemlerini göstermektedir.

Ayrıştırma işlemi resmi bütünü yerine her maske için ayrı ayrı yapılır. Ancak her maskenin aynı sayıda bit düzlemlerine sahip olması gerekmez. Her maskeye detayı derecesinde bit düzlemi tahsis edilir.



(a)



(b)



(c)



(d)



(e)

Şekil 3. Görüntünün  $N=4$  için bit düzlemlerine ayrıştırılması. (a) Orjinal görüntü (b) 0.Bit-düzlemi (c) 1.Bit-düzlemi (d) 2.Bit-düzlemi (e) 3.Bit-düzlemi

Detay derecesi ise maskenin ayrık kosinüs dönüşümünün genlik değerlerinin toplamıyla ölçülür. Resimdeki her bir maskeye ait AKD katsayılarının maksimum ve minimum değerleri  $N$  aralığa bölünür. Bu aralık lineer olabileceği gibi logaritmik de olabilir. Bu aralıklardan maksimum olan aralığa karşılık gelen maskeye  $N$  adet (maksimum sayıda) düzlem tahsis edilir. Minimum aralığa düşen maskeye ise kullanıcı isteğine bağlı olarak hangi oranda sıkıştırma isteniyorsa o oran dikkate alınarak bit düzlemi tahsisi (En az 1) yapılır. Böylece detayı fazla olan bölgeler fazla düzlem ile az olan bölgeler ise derecelerine göre daha az sayıda düzlem ile oluşturulmuş resim görüntüleri şeklinde alıcıda meydana gelecektir.

(1)'de ifade edildiği gibi iletilmesi istenen resim  $[\mathbf{I}(i, j)]_{U \times V}$  ile maskeler ise  $\mathbf{I}_{x,y}(i, j)$  gösterilirse;

$$\mathbf{I}_{x,y}(i, j) = 2^{N-1} \mathbf{I}_{x,y}^{(N-1)}(i, j) + 2^{N-2} \mathbf{I}_{x,y}^{(N-2)}(i, j) + \dots + 2^0 \mathbf{I}_{x,y}^{(0)}(i, j)$$

$$\Rightarrow \mathbf{I}_{x,y}(i, j) = \sum_{k=1}^N 2^{N-k} \mathbf{I}_{x,y}^{(N-k)}(i, j) \quad (2)$$

Şeklinde bit düzlemlerinin toplamı şeklinde ifade edilebilir.

$$\bar{\mathbf{a}}^{(x,y)} \square \left[ \mathbf{a}_{N-1}^{(x,y)} \quad \mathbf{a}_{N-2}^{(x,y)} \quad \dots \quad \mathbf{a}_0^{(x,y)} \right] \in [0,1] \quad (3)$$

vektörünü tanımlayalım.  $\bar{\mathbf{a}}^{(x,y)}$  vektörünün elemanları (p,r)nci maskenin AKD genlikleriyle belirlenecektir. Bu durumda I(i,j) yi şu şekilde ifade edebiliriz;

$$\mathbf{I}(i, j) = \begin{bmatrix} \sum_{k=1}^M \overbrace{\mathbf{a}_{N-k}^{(1,1)} 2^{N-k} \mathbf{I}_{1,1}^{N-k}(i, j)}^{(1,1)\text{inci altresim}} & \dots & \sum_{k=1}^M \overbrace{\mathbf{a}_{N-k}^{(1,r)} 2^{N-k} \mathbf{I}_{1,r}^{N-k}(i, j)}^{(1,r)\text{inci altresim}} \\ \vdots & \ddots & \vdots \\ \sum_{k=1}^M \mathbf{a}_{N-k}^{(p,1)} 2^{N-k} \mathbf{I}_{p,1}^{N-k}(i, j) & \dots & \sum_{k=1}^M \underbrace{\mathbf{a}_{N-k}^{(p,r)} 2^{N-k} \mathbf{I}_{p,r}^{N-k}(i, j)}_{(p,r)\text{inci altresim}} \end{bmatrix}$$

$$= \sum_{k=1}^M 2^{N-k} \begin{bmatrix} \mathbf{a}_{N-k}^{(1,1)} \mathbf{I}_{1,1}^{N-k}(i, j) & \dots & \mathbf{a}_{N-k}^{(1,r)} \mathbf{I}_{1,r}^{N-k}(i, j) \\ \vdots & \ddots & \vdots \\ \mathbf{a}_{N-k}^{(p,1)} \mathbf{I}_{p,1}^{N-k}(i, j) & \dots & \mathbf{a}_{N-k}^{(p,r)} \mathbf{I}_{p,r}^{N-k}(i, j) \end{bmatrix} \quad (4)$$

ve

$$\bar{\mathbf{I}}_{x,y} \square \left[ 2^{N-1} \mathbf{I}_{x,y}^{(N-1)} \quad 2^{N-2} \mathbf{I}_{x,y}^{(N-2)} \quad \dots \quad 2^0 \mathbf{I}_{x,y}^{(0)} \right]^T \quad (5)$$

olarak tanımlanırsa;

$\Rightarrow$  (p,r) nci maske,  $\bar{\mathbf{I}}_{p,r} * \bar{\mathbf{a}}^{(p,r)}$  olarak yazılabilir. (\*), iç çarpım operatörüdür. Bu durumda resim bilgisi,

$$\mathbf{I}(i, j) = \begin{bmatrix} \bar{\mathbf{I}}_{1,1} * \bar{\mathbf{a}}^{(1,1)} & \bar{\mathbf{I}}_{1,2} * \bar{\mathbf{a}}^{(1,2)} & \dots & \bar{\mathbf{I}}_{1,r} * \bar{\mathbf{a}}^{(1,r)} \\ \vdots & \ddots & & \vdots \\ \bar{\mathbf{I}}_{p,1} * \bar{\mathbf{a}}^{(p,1)} & \dots & & \bar{\mathbf{I}}_{p,r} * \bar{\mathbf{a}}^{(p,r)} \end{bmatrix} \quad (6)$$

olacaktır. Tüm Maskelere maksimum sayıda bit düzlemleri tahsis edilmemesi sıkıştırma anlamına gelir ve bu oran  $\% \varphi / N$  ( $\varphi$  vektörde bulunan sıfırların sayısı olmak üzere) ile ölçülür. Hangi maskeye ne kadar bit düzleminin tahsis edileceğini ilgili maskenin AKD katsayıları belirler ve bu katsayıların genliklerinin büyüklüğü doğrultusunda  $\bar{\mathbf{a}}^{(x,y)} \in [0,1]$ , ( $x = 1 \dots p$ ,  $y = 1 \dots r$ )

vektörü oluşturulur. Örnek vermek gerekirse  $\bar{\mathbf{a}}^{(1,1)} = [1 \ 1 \ 0 \ 0]^T$ , ve  $N=4$ , (4 düzlem olup resim 16 seviyede ifade ediliyor) olsun. (1,1)nci maskeye ilişkin sıkıştırma oranı  $\% \frac{2}{4} \cdot 100 = \%50$

olacaktır. Bu durumda resim oluşturmada 3. ve 2. düzlemler dikkate alınmakta, 1. ve 0. düzlemler ihmal edilmekte, dolayısı ile verici tarafından gönderilmeye ihtiyaç bulunmamaktadır ve bu Maske 4 seviyede detaylandırılmaktadır. Kısacası bu maske 4 yerine 2 bit düzlemiyle ifade edilmiş, bu da  $\%50$ 'lik bir sıkıştırma sağlamış ancak bu maske 16 yerine 4 seviyede detaylandırılarak biraz kontrast kaybına yol açılmıştır. Ancak bu kaybın çok önemli olmaması gerekir, çünkü  $\%50$ lik sıkıştırma yapabileceğimizi AKD katsayıları belirlemiş, bu katsayılarla zaten o bölgede çok fazla detay olmadığını ifade etmiş ve o bölgenin 4 yerine 2 bit düzlemiyle ifade edilmesinin (o bölge için) önemli bir kontrast kaybına yol açmayacağını başlangıçta bize söylemiş olmaktadır.

### Düzlem seçme

İletim anında düzlemlerin gönderilme sırası önemlidir. En genel durumda örneğin herhangi bir maske N düzlemle ifade ediliyorsa bu sıra, (N-1)nci, (N-2)nci, ..., 0nci düzlemlerin gönderilmesi şeklinde olmalıdır. (N-1)nci düzlem gönderildiğinde resim alıcıda en düşük detay seviyesinde oluşturulacaktır. Diğer düzlemler de geldikçe her bir maskenin detayında aşamalı olarak artışlar görülecektir. Tüm düzlemlerin gönderilmesi durumunda alıcıda, orijinal resimle aynı detaya sahip resim elde edilmiş olacaktır. Bu algoritma özellikle sınırlı-band genişliğine sahip olma durumunda ve kısa zaman aralığında orijinal resmin daha az doğrulukta olsa bile yaklaşımının elde edilmesi uygulamaları için faydalıdır. Bu doğrulukta resme bakarak, kul-

lanıcı, resmin detaylarının tümünü göndermenin gerekli olup olmadığına karar verebilir. Böylece birim zamanda iletilen bilgi akışının kontrolü yapılmış olur.

### Turbo kodlama

Şekil 1'deki sistem şematik olarak ifade edilirse Şekil 4'deki akış diyagramı elde edilir. Turbo kodlayıcı kısmı iki tane genellikle aynı yapıdaki "Yinelemeli Katlamalı Kodlayıcı" (YKK)'dan oluşur. Her iki kodlayıcı da aynı veriyi alır, fakat ikinci kodlayıcı giriş verisini serpiştiriciden geçtikten sonra oluşan yeni dizilimli veriyi alır. Turbo kodlarının rasgele gibi görünmesini sağlayan bu serpiştirme işlemidir. M hafıza genişliğine sahip bir kodlayıcı düşünelim. Eğer k anındaki giriş ( $d_k$ ) k anındaki çıkışa eşit ise;

$$X_k = d_k \quad (7)$$

kalan  $r(D)$ , geri besleme  $g_{(0)}(D)$  ve ileri besleme polinomu  $g_{(1)}(D)$  ile bulunur. Geri besleme değişkeni;

$$r_i = d_k + \sum_{j=1}^K r_{k-j} g_j^{(0)} \quad (8)$$

ve YKK kodlayıcı çıkışı  $Y_k$ , parite veri olarak adlandırılır.

$$Y_k = \sum_{j=0}^K r_{k-j} g_j^{(1)} \quad (9)$$

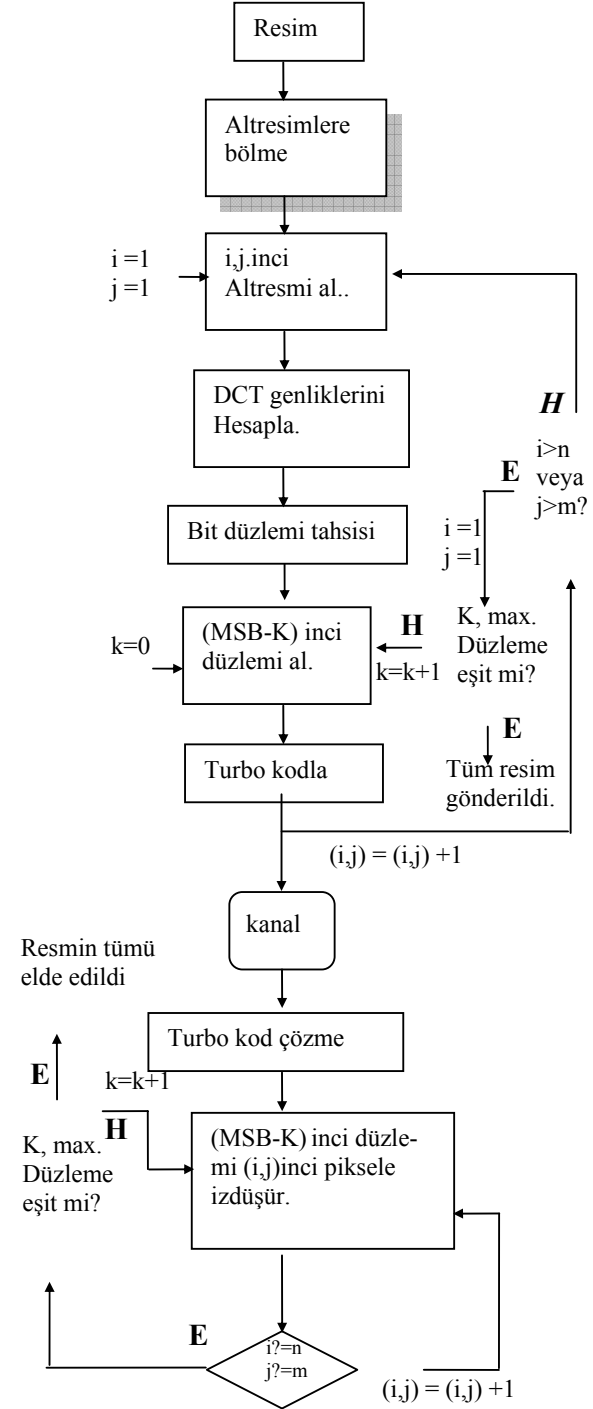
İki hafızalı ( $M=2$ ), kodlama oranı  $1/2$  olan ( $R=1/2$ ), geri besleme polinomu  $g(0)=7$  ve ileri besleme polinomu  $g(1)=5$  olan bir YKK kodlayıcının üretici matrisi ise aşağıda gösterilmektedir (Valenti, 1998).

$$G(D) = \begin{bmatrix} 1 & \frac{1+D+D^2}{1+D^2} \end{bmatrix} \quad (10)$$

### Turbo kod çözücü

Turbo kod çözücü olarak EBS algoritması kullanılmaktadır (Valenti, 1998; Gross ve Gulak, 1998). EBS algoritmasının amacı gürültülü bir

ortamda Markov işlevi tarafından üretilen durum geçişi, mesaj biti veya sembolü'ne ait En Büyük Sonsal olasılık değerini bulmaktır (Benedetto and Montorsi, 1996; Benedetto vd., 1996a).



Şekil 4. Sistemin akış diyagramı

Her iki kod çözücünün verileri kullanılarak kod çözme işlemi yinelemeli olarak yapılır. Şekil 5'te turbo kod çözücü ve düzlemlerden resmin elde edilmesinin sistematik yapısı görülmektedir. Sistemde iki adet kod çözücü vardır. Sembol-sembol EBS kod çözücüsünde çıkış

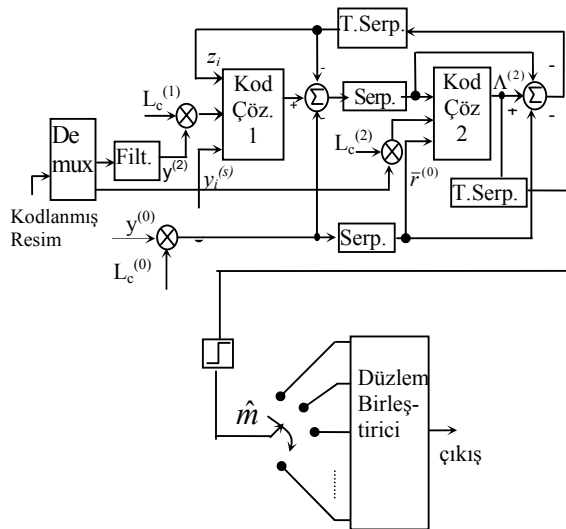
$$\Lambda_k = \ln \frac{P[m_k = 1 | y]}{P[m_k = 0 | y]} \quad (11)$$

şeklinde dir. m mesaj bitlerini, y ise alıcıdaki dizi yi ifade etmektedir. SISO (Soft Input Soft Output, yumuşak-girişli yumuşak-çıkışlı) kod çözücüsüne üç giriş gelmektedir. Bunlar; sistematik bilgi  $y_i(s)$ , parite veri  $y_i(p)$  ve diğer kod çözücünden gelen  $z_i$  bilgisidir. Logaritmik olasılık oranı (log-likelihood ratio LLR) şu şekilde ifade edilir.

$$\Lambda_k = \ln \frac{\sum_{S_1} \exp[\bar{\alpha}(s_k) + \bar{\gamma}(s_k \rightarrow s_{k+1}) + \bar{\beta}(s_{k+1})]}{\sum_{S_0} \exp[\bar{\alpha}(s_k) + \bar{\gamma}(s_k \rightarrow s_{k+1}) + \bar{\beta}(s_{k+1})]} \quad (12)$$

Burada  $S_1 = \{s_k \rightarrow s_{k+1} | m_k=1\}$  mesaj bitinin 1 olması durumundaki tüm durum geçişlerini ve  $S_0 = \{s_k \rightarrow s_{k+1} | m_k=0\}$  mesaj bitinin 0 olması durumundaki tüm durum geçişlerini gösterir. Son iterasyonda, ikinci kod çözücünün çıkışı kullanılarak sert karar gerçekleştirilir.

$$\hat{m}_k = \begin{cases} 1 & \text{if } \Lambda^{(2)} \geq 0 \\ 0 & \text{if } \Lambda^{(2)} < 0 \end{cases} \quad (13)$$



Şekil 5. Kod Çözme Ünitesinin yapısı

## Aşamalı iletim

Aşamalı iletimde zaman geçtikçe (her bir iterasyonda) alıcıda oluşturulan resmin detaylarında iyileşme görülecektir. 1.iterasyon için denklem (4)'de  $M=1$  alınırsa;

$$\Rightarrow \mathbf{I}(i,j) = 2^{N-1} \begin{bmatrix} a_{N-1,1}^{(1)} \mathbf{I}_{1,1}^{N-1}(i,j) & a_{N-1,2}^{(2)} \mathbf{I}_{1,2}^{N-1}(i,j) & \cdots & a_{N-1,r}^{(r)} \mathbf{I}_{1,r}^{N-1}(i,j) \\ \vdots & \ddots & & \vdots \\ a_{N-1,p,1}^{(p,1)} \mathbf{I}_{p,1}^{N-1}(i,j) & \cdots & & a_{N-1,p,r}^{(p,r)} \mathbf{I}_{p,r}^{N-1}(i,j) \end{bmatrix} \quad (14)$$

Bu alıcıda iki seviyeli (Binary Image) resmin oluşması anlamına gelir. Tüm pikseller 0 veya 1 değerlerinden birine sahiptir.

İkinci iterasyonda denklem (4)'de  $M=2$  alınır.

$$\Rightarrow \mathbf{I}(i,j) = \mathbf{I}(i,j) \Big|_{M=1} + 2^{N-2} \begin{bmatrix} a_{N-2,1,1}^{(1,1)} \mathbf{I}_{1,1}^{N-2}(i,j) & \cdots & a_{N-2,1,r}^{(1,r)} \mathbf{I}_{1,r}^{N-2}(i,j) \\ \vdots & \ddots & \vdots \\ a_{N-2,p,1}^{(p,1)} \mathbf{I}_{p,1}^{N-2}(i,j) & \cdots & a_{N-2,p,r}^{(p,r)} \mathbf{I}_{p,r}^{N-2}(i,j) \end{bmatrix} \quad (15)$$

Bu durumda alıcıda 4 seviyeli resim bilgisi oluşturulmuş olur. İlk duruma göre kontrast daha iyi olacaktır. Nnci iterasyonda, denklem (4)'de  $M=N$  alınır.

$$\Rightarrow \mathbf{I}(i,j) = \mathbf{I}(i,j) \Big|_{M=N-1} + 2^0 \begin{bmatrix} a_0^{(1,1)} \mathbf{I}_{1,1}^0(i,j) & a_0^{(1,2)} \mathbf{I}_{1,2}^0(i,j) & \cdots & a_0^{(1,r)} \mathbf{I}_{1,r}^0(i,j) \\ \vdots & \ddots & & \vdots \\ a_0^{(p,1)} \mathbf{I}_{p,1}^0(i,j) & \cdots & & a_0^{(p,r)} \mathbf{I}_{p,r}^0(i,j) \end{bmatrix} \quad (16)$$

Bu ise  $2^N$  seviyede bir resim bilgisinin elde edilmesini sağlar (Maximum Contrast).

## N=4 için özel durum

$N=4$  (16 gri seviye) ve resim bilgisi (1,1), (1,2), (2,1), (2,2) olmak üzere 4 maskeye ayrılabilir. Yani;

$$\mathbf{I}(i, j) = \begin{bmatrix} \mathbf{I}_{11}(i, j) & \mathbf{I}_{12}(i, j) \\ \mathbf{I}_{21}(i, j) & \mathbf{I}_{22}(i, j) \end{bmatrix} \quad (17)$$

Her bir maske,

$$\mathbf{I}_{x,y}(i, j) = 2^3 \mathbf{I}_{x,y}^{(3)}(i, j) + 2^2 \mathbf{I}_{x,y}^{(2)}(i, j) + 2^1 \mathbf{I}_{x,y}^{(1)}(i, j) + 2^0 \mathbf{I}_{x,y}^{(0)}(i, j)$$

şeklinde yazılabilir.

#### N=4 özel durumu için sıkıştırma

AKD katsayıları gereği, (1,1)nci maske  $\mathbf{I}_{11}(i, j)$ 'e, 2 düzlem, (1,2)nci maske  $\mathbf{I}_{12}(i, j)$ 'e, 1 düzlem, (2,1)nci maske  $\mathbf{I}_{21}(i, j)$ 'e, 3 düzlem, son olarak (2,2)nci maske  $\mathbf{I}_{22}(i, j)$ 'e, 4 düzlem tahsis edilmiş olsun. Oluşturulan  $\bar{\mathbf{a}}^{(x,y)}$  vektörü

$$\begin{aligned} \bar{\mathbf{a}}^{(1,1)} &\square [1 \ 1 \ 0 \ 0], \bar{\mathbf{a}}^{(1,2)} \square [1 \ 0 \ 0 \ 0] \\ \bar{\mathbf{a}}^{(2,1)} &\square [1 \ 1 \ 1 \ 0], \bar{\mathbf{a}}^{(2,2)} \square [1 \ 1 \ 1 \ 1] \end{aligned} \quad (18)$$

şeklinde olacaktır. Bu duruma ilişkin  $\%(\varphi/N)$  gereği sıkıştırma oranları (1,1)nci maske için %50, (1,2)nci maske için %75, (2,1)nci maske için %25, (2,2)nci maske için %0 olacaktır. Sonuçta eşdeğer olarak %62 oranında bir sıkıştırma kazancına sahip olacağız. Denklem (5) ten;

$$\mathbf{I}_{x,y}(i, j) = \bar{\mathbf{I}}_{x,y} * \bar{\mathbf{a}}^{(x,y)} \quad (19)$$

Burada

$$\bar{\mathbf{a}}^{(x,y)} = \begin{bmatrix} \mathbf{a}_3^{(x,y)} & \mathbf{a}_2^{(x,y)} & \mathbf{a}_1^{(x,y)} & \mathbf{a}_0^{(x,y)} \end{bmatrix}^T \quad (20)$$

$$\begin{aligned} \bar{\mathbf{I}}_{x,y} &= \begin{bmatrix} 2^{4-1} \mathbf{I}_{x,y}^{(4-1)} & 2^{4-2} \mathbf{I}_{x,y}^{(4-2)} & 2^{4-3} \mathbf{I}_{x,y}^{(4-3)} & 2^{4-4} \mathbf{I}_{x,y}^{(4-4)} \end{bmatrix} \\ &= \begin{bmatrix} 8\mathbf{I}_{x,y}^{(3)} & 4\mathbf{I}_{x,y}^{(2)} & 2\mathbf{I}_{x,y}^{(1)} & \mathbf{I}_{x,y}^{(0)} \end{bmatrix} \end{aligned} \quad (21)$$

yazılabilir.

#### (1,1)nci maske için

$$\begin{aligned} \mathbf{I}_{1,1}(i, j) &= \bar{\mathbf{I}}_{1,1} * \bar{\mathbf{a}}^{(1,1)} = \begin{bmatrix} 8\mathbf{I}_{1,1}^{(3)} & 4\mathbf{I}_{1,1}^{(2)} & 2\mathbf{I}_{1,1}^{(1)} & \mathbf{I}_{1,1}^{(0)} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \\ &= 8\mathbf{I}_{1,1}^{(3)} + 4\mathbf{I}_{1,1}^{(2)} \end{aligned} \quad (22)$$

Alicıda bu maskenin elde edilmesinde  $\mathbf{I}_{1,1}^{(3)}$  ve  $\mathbf{I}_{1,1}^{(2)}$ , yani bu maskeye ait 3ncü ve 2nci düzlemler kullanılacak olup;

#### (1,2). maske için

$$\mathbf{I}_{1,2}(i, j) = \bar{\mathbf{I}}_{1,2} * \bar{\mathbf{a}}^{(1,2)} = \begin{bmatrix} 8\mathbf{I}_{1,2}^{(3)} & 4\mathbf{I}_{1,2}^{(2)} & 2\mathbf{I}_{1,2}^{(1)} & \mathbf{I}_{1,2}^{(0)} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = 8\mathbf{I}_{1,2}^{(3)} \quad (23)$$

Sadece  $\mathbf{I}_{1,2}^{(3)}$ , yani bu maskeye ait 3ncü düzlem kullanılacak olup,

#### (2,1). maske için

$$\begin{aligned} \mathbf{I}_{2,1}(i, j) &= \bar{\mathbf{I}}_{2,1} * \bar{\mathbf{a}}^{(2,1)} = \begin{bmatrix} 8\mathbf{I}_{2,1}^{(3)} & 4\mathbf{I}_{2,1}^{(2)} & 2\mathbf{I}_{2,1}^{(1)} & \mathbf{I}_{2,1}^{(0)} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} \\ &= 8\mathbf{I}_{2,1}^{(3)} + 4\mathbf{I}_{2,1}^{(2)} + 2\mathbf{I}_{2,1}^{(1)} \end{aligned} \quad (24)$$

$\mathbf{I}_{2,1}^{(3)}$ ,  $\mathbf{I}_{2,1}^{(2)}$  ve  $\mathbf{I}_{2,1}^{(1)}$ , yani bu maskeye ait 3ncü ve 2nci ve 1nci düzlemler kullanılacaktır.

#### (2,2). maske için

$$\begin{aligned} \mathbf{I}_{2,2}(i, j) &= \bar{\mathbf{I}}_{2,2} * \bar{\mathbf{a}}^{(2,2)} = \begin{bmatrix} 8\mathbf{I}_{2,2}^{(3)} & 4\mathbf{I}_{2,2}^{(2)} & 2\mathbf{I}_{2,2}^{(1)} & \mathbf{I}_{2,2}^{(0)} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \\ &= 8\mathbf{I}_{2,2}^{(3)} + 4\mathbf{I}_{2,2}^{(2)} + 2\mathbf{I}_{2,2}^{(1)} + \mathbf{I}_{2,2}^{(0)} \end{aligned} \quad (25)$$

Olup bu maske için tüm düzlemler dikkate alınacaktır.

#### N=4 özel durum için aşamalı iletim

##### 1. İterasyon için

$$\begin{aligned} \mathbf{I}(i, j) &= 2^3 \begin{bmatrix} \mathbf{a}_3^{(1,1)} \mathbf{I}_{11}^{(3)}(i, j) & \mathbf{a}_3^{(1,2)} \mathbf{I}_{12}^{(3)}(i, j) \\ \mathbf{a}_3^{(2,1)} \mathbf{I}_{21}^{(3)}(i, j) & \mathbf{a}_3^{(2,2)} \mathbf{I}_{22}^{(3)}(i, j) \end{bmatrix} \\ &= 8 \begin{bmatrix} 1 * \mathbf{I}_{11}^{(3)}(i, j) & 1 * \mathbf{I}_{12}^{(3)}(i, j) \\ 1 * \mathbf{I}_{21}^{(3)}(i, j) & 1 * \mathbf{I}_{22}^{(3)}(i, j) \end{bmatrix} \square \mathbf{I}(i, j) \Big|_{M=1} \end{aligned} \quad (26)$$

Aşamalı iletimde ise resim bilgisi (26) ile ifade edilebilir. Aşağıdaki şekilde ilk iterasyon sonucu maskelerin resim üzerindeki yerleşimi görülmektedir.

$\hat{I}_{1,1}^3(i,j)$	$\hat{I}_{1,2}^3(i,j)$
$\hat{I}_{2,1}^3(i,j)$	$\hat{I}_{2,2}^3(i,j)$

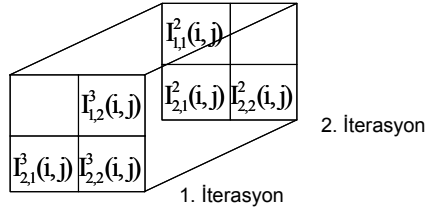
Şekil 6. 1.iterasyon sonucu maskelerin resim üzerindeki durumu

### 2nci İterasyon için

Bir önceki iterasyonda elde edilen bilgiler de dikkate alınarak;

$$\begin{aligned} \mathbf{I}(i,j) &= \mathbf{I}(i,j)|_{M=1} + 2^2 \begin{bmatrix} a_2^{(1,1)} \mathbf{I}_{11}^{(2)}(i,j) & a_2^{(1,2)} \mathbf{I}_{12}^{(2)}(i,j) \\ a_2^{(2,1)} \mathbf{I}_{21}^{(2)}(i,j) & a_2^{(2,2)} \mathbf{I}_{22}^{(2)}(i,j) \end{bmatrix} \\ &= 4 \begin{bmatrix} 1 * \mathbf{I}_{11}^{(2)}(i,j) & 0 * \mathbf{I}_{12}^{(2)}(i,j) \\ 1 * \mathbf{I}_{21}^{(2)}(i,j) & 1 * \mathbf{I}_{22}^{(2)}(i,j) \end{bmatrix} \\ &= 4 \begin{bmatrix} \mathbf{I}_{11}^{(2)}(i,j) & 0 \\ \mathbf{I}_{21}^{(2)}(i,j) & \mathbf{I}_{22}^{(2)}(i,j) \end{bmatrix} \square \mathbf{I}(i,j)|_{M=2} \end{aligned} \quad (27)$$

olacaktır. İki iterasyon sonucu maskelerin resim üzerindeki yerleşimi ise aşağıda görülmektedir.

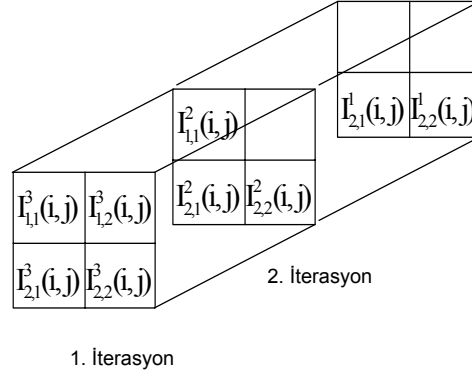


Şekil 7. 2.iterasyon sonucu maskelerin resim üzerindeki durumu

### 3nci İterasyon için

$$\begin{aligned} \mathbf{I}(i,j) &= \mathbf{I}(i,j)|_{M=2} + 2^1 \begin{bmatrix} a_1^{(1,1)} \mathbf{I}_{11}^{(1)}(i,j) & a_1^{(1,2)} \mathbf{I}_{12}^{(1)}(i,j) \\ a_1^{(2,1)} \mathbf{I}_{21}^{(1)}(i,j) & a_1^{(2,2)} \mathbf{I}_{22}^{(1)}(i,j) \end{bmatrix} \\ &= \mathbf{I}(i,j)|_{M=2} + 2 \begin{bmatrix} 0 & 0 \\ \mathbf{I}_{21}^{(1)}(i,j) & \mathbf{I}_{22}^{(1)}(i,j) \end{bmatrix} \square \mathbf{I}(i,j)|_{M=2} \end{aligned} \quad (28)$$

dır. Aşağıdaki şekilde üç iterasyon sonucu maskelerin resim üzerindeki yerleşimi görülmektedir.

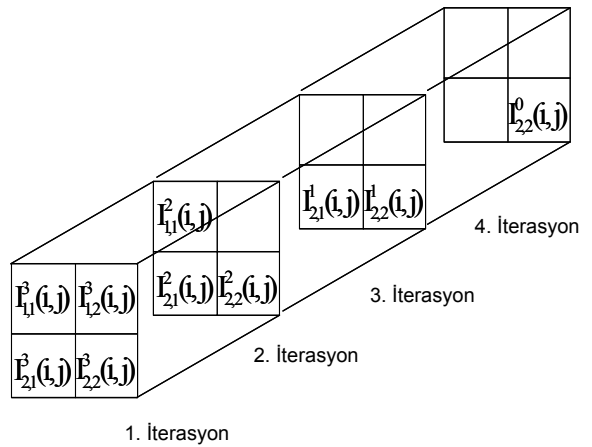


Şekil 8. 3.iterasyon sonucu maskelerin resim üzerindeki durumu

### 4nci İterasyon için

$$\begin{aligned} \mathbf{I}(i,j) &= \mathbf{I}(i,j)|_{M=3} + 2^0 \begin{bmatrix} a_0^{(1,1)} \mathbf{I}_{11}^{(0)}(i,j) & a_0^{(1,2)} \mathbf{I}_{12}^{(0)}(i,j) \\ a_0^{(2,1)} \mathbf{I}_{21}^{(0)}(i,j) & a_0^{(2,2)} \mathbf{I}_{22}^{(0)}(i,j) \end{bmatrix} \\ &= \mathbf{I}(i,j)|_{M=3} + 2 \begin{bmatrix} 0 & 0 \\ 0 & \mathbf{I}_{22}^{(0)}(i,j) \end{bmatrix} \end{aligned} \quad (29)$$

olacaktır. Aşağıdaki şekilde son (dört) iterasyon sonucu maskelerin resim üzerindeki yerleşimi görülmektedir.



Şekil 9. 4.iterasyon sonucu maskelerin resim üzerindeki durumu



En son iterasyon sonucu maskeler üst üste çakıştırılarak resmin tamamı oluşturulmuş olur. İterasyon 1'den 4'e kadar devam ettikçe resim bilgisi de yenilenmiş olacaktır.

### Deneyisel sonuçlar

150x150 boyutunda test resmi, ½ hızında YKK kodlayıcı, rasgele serpiştirici ve AWGN kanal modeli üzerinde simülasyon yapılmıştır. Resmin her bir pikseli 16 seviyeye ayrıştırılmıştır. Bu da resim için çalışılan düzlem sayısının N=4 olması anlamına gelir. Test resmine ait düzlemler Şekil 3'te görülmektedir. Buradan en yüksek anlamlı düzlem olan 3. düzlemin en fazla komşuluk ilişkisine sahip olduğu, dolayısıyla veri sıkıştırma işleminde en fazla dikkate alınması gerekenin bu düzlem olduğu sonucuna varılabilir. Eğer resme ait ince detaylar fazla bir önem teşkil etmiyorsa sadece 3. düzlem kodlanıp gönderilerek orijinal resme ait önemli bilgiler elde edilebilir. Özellikle hız ve hafıza gereksiniminin fazla olduğu durumlarda bu yöntem ilgi çekici neticeler verecektir. Sadece bu düzlemin gönderilmesiyle %75 oranında veri sıkıştırması yapılmış olur. Daha fazla detay istendiğinde 2. düzlem de gönderilebilir. Daha sonra alıcıda bu iki düzlem birbirleriyle üst üste bindirilerek detay resim oluşturulur. Bu durumda sıkıştırma oranı %50'ye inecek fakat ilkine göre daha fazla detay elde edilecektir. Dört düzlemin de gönderilmesi ile maksimum detay elde edilecek ancak bu durumda veri sıkıştırması mümkün olamayacaktır.

Resim 25x25 ve 10x10 boyutlarında maskelenmiş ve her birinin AKD katsayıları değerlendirilmiş ve her bir Maskeye bu katsayılar doğrultusunda düzlem tahsis edilmiş ve bu düzlemler, en yüksek anlamlı olandan başlamak üzere sırayla kodlanarak kanala verilmiştir. Resimde gökyüzü fazla detay ((1,1). Maske) içermediği için 1 düzlemle, bina ise ((6,7). Maske) fazla detay içerdiği için 4 düzlemle (resimde daire içine alınmış bölgeler), diğer bölgeler ise detay derecelerine göre farklı sayılarda düzlemlerle gönderilmiştir. Resimde bulunan şekillerin büyüklüğüne göre maske boyutu değiştirilebilir.

Eğer şekiller büyükse maske boyutu büyük seçilebilir. Resmin içinde bulunan küçük şekiller de farklı seviyelerde detaylandırılmak isteniyorsa o zaman maske boyutu daha da küçük seçilmelidir.

Şekil 10(a), (b) ve (c)'de, maske boyutunun 25x25 olması, Şekil 11(a), (b) ve (c)'de ise maske boyutunun 10x10 olması, durumunda alıcıda yeniden elde edilen resimler gösterilmektedir. Şekil 10, 11 (d), (e) ve (f) ise AKD katsayılarının farklı aralıklara bölünüp değerlendirilmeleri sonucu elde edilen düzlem histogramlarını göstermektedir.

Örneğin Şekil 10(a)'nın elde edilmesinde toplam 50x1 adet 0.düzlem, 14x2 adet 1.düzlem, 9x3 adet 2.düzlem ve 27x4 adet 3.düzlem kullanılmıştır. Dolayısıyla sıkıştırma oranı;

$$(10 \times 10 \times 4 - 50 \times 1 - 14 \times 2 - 9 \times 3 - 27 \times 4) / 10 \times 10 \times 4 = \%46 \text{ olacaktır.}$$

Eğer aynı sıkıştırma oranı resmin bölgesel detayları dikkate alınmadan gerçekleştirilseydi, o zaman resmin her maskesine aynı sayıda düzlem tahsis edilecek dolayısıyla bazı bölgelerin detayında kayıplar meydana gelecekti. Bu yolla istenen bölgenin istenen detay seviyesinde detaylandırılması mümkün hale gelir. Benzer şekilde Şekil 10(b) için elde edilen sıkıştırma;

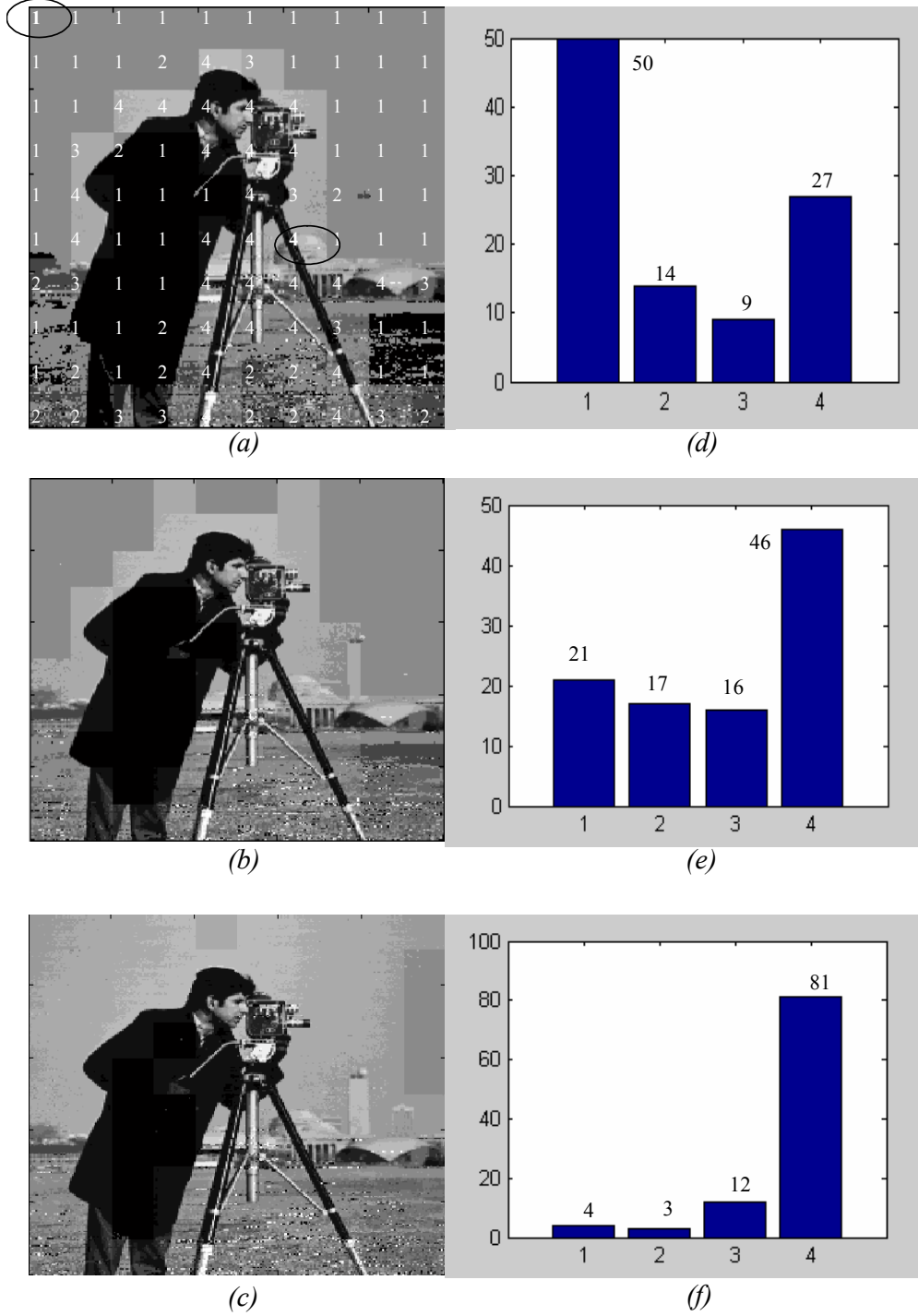
$$(10 \times 10 \times 4 - 21 \times 1 - 17 \times 2 - 16 \times 3 - 46 \times 4) / 10 \times 10 \times 4 = \%28 \text{ olup Şekil 10(c) için}$$

$$(10 \times 10 \times 4 - 21 \times 1 - 17 \times 2 - 16 \times 3 - x4) / 10 \times 10 \times 4 = \%7.5, \text{ Şekil 11(a) için}$$

$$(25 \times 25 \times 4 - 374 \times 1 - 64 \times 2 - 61 \times 3 - 26 \times 4) / 25 \times 25 \times 4 = \%52, \text{ Şekil 11(b) için}$$

$$(25 \times 25 \times 4 - 110 \times 1 - 126 \times 2 - 138 \times 3 - 51 \times 4) / 25 \times 25 \times 4 = \%29 \text{ ve son olarak, Şekil 11(c) için}$$

$$(25 \times 25 \times 4 - 70 \times 1 - 7 \times 2 - 9 \times 3 - 539 \times 4) / 25 \times 25 \times 4 = \%9 \text{ olacaktır.}$$

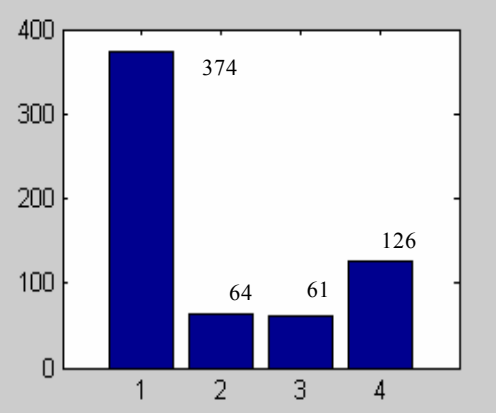


Şekil 10.  $M=25$  için alıcıda elde edilen resim bilgisi histogramları  
(a) Sıkıştırma oranı % 46 (b) Sıkıştırma oranı % 28 (c) Sıkıştırma oranı % 7.5  
(d) (a)'ya ait bit düzlem histogramı (e) (b)'ya ait bit düzlem histogramı (f) (c)'ye ait bit düzlem histogramı

*Turbo kodlamalı resimler*



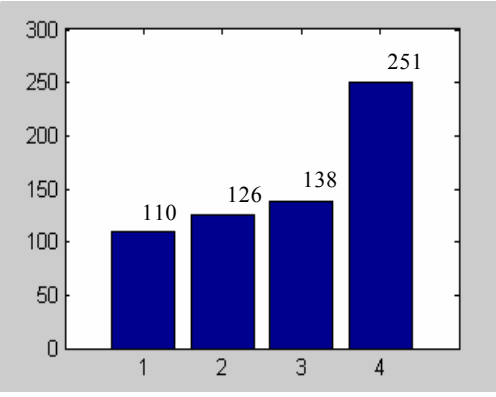
(a)



(d)



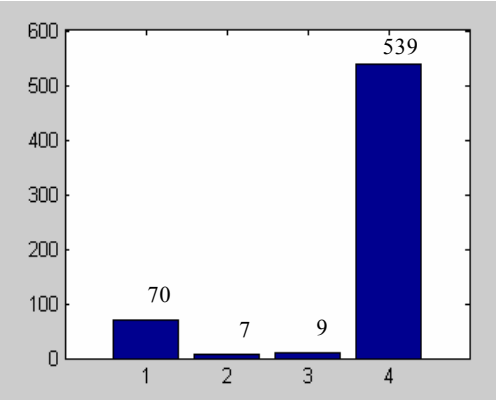
(b)



(e)



(c)



(f)

Şekil 11.  $M=10$  için alıcıda elde edilen resim bilgisi histogramları.  
(a) Sıkıştırma oranı % 52 (b) Sıkıştırma oranı % 29 (c) Sıkıştırma oranı % 9  
(d) (a)'ya ait bit düzlem histogramı (e) (b)'ya ait bit düzlem histogramı (f) (c)'ye ait bit düzlem histogramı

## Kaynaklar

- Benedetto S. ve Montorsi G., (1996). Unveiling turbo codes: some results on parallel concatenated coding schemes, *IEEE Trans. on Inform. Theory*, **42**, 2, 409-428.
- Benedetto S., Divsalar D., Montorsi G., and Pollara F., (1996a). A soft-input soft-output maximum a posteriori (map) module to decode parallel and serial concatenated codes. *TDA Progress Report*, 42-127, 1-20.
- Büyükatak K, Uçan O. N, Kargin S., Kent S., Göse E., (2004). Uyarlamalı wiener-turbo (UW-TS) sistemi ve bu sistemin iki boyutlu işaretlerin iletiminde kullanılması, *Havacılıkta İleri Teknolojiler ve Uygulamaları Sempozyumu (HITEK), İstanbul*.
- Büyükatak K., Göse G., Uçan O.N., Kent S., Osman O., (2003). Channel equalization and noise reduction based on turbo codes, *Proceedings of International Conference on Recent Advances in Space Technologies, İstanbul*.
- Gonzales R.C., Woods R.E., (1992) *Digital Image Processing*, ISBN 0-201-50803-6, 349-354.
- Gross W.J. ve Gulak P.G., (1998). Simplified MAP Algorithm suitable for implementation of Turbo Decoders, *Electronic Letters online*, **34**, 16, 1577-1578.
- Naguib A. F., Seshadri N., ve Calderbank A. R., (2000). Increasing data rate over wireless channels, *IEEE Signal Processing Magazine*, 76-92.
- Sherwood P. G. ve Zeger K., (1998). Error protection for progressive image transmission over memoryless and fading channels. *IEEE Trans. Communications*, **46**, 12, 1555-1559, December.
- Valenti M.C., (1998). Iterative detection and decoding for wireless communications, a proposal for current and future work toward Doctor of Philosophy degree. (<http://citeseer.ist.psu.edu/532677.html>), 48-52.