

# Vektör Nicemleme ile Video Görüntü Sıkıştırma

## Video Compression with Vector Quantization

<sup>1</sup>Erkan KOCAKAYA, <sup>2</sup>Mehmet YAKUT, <sup>3</sup>Aybike KOCAKAYA

Elektronik ve Haberleşme Mühendisliği

Kocaeli Üniversitesi, Kocaeli/İzmit

[ekocakaya@kou.edu.tr](mailto:ekocakaya@kou.edu.tr), [myakut@kou.edu.tr](mailto:myakut@kou.edu.tr), [adirikol@kou.edu.tr](mailto:adirikol@kou.edu.tr)

### Özetçe

Günümüzde birçok video görüntü dizisi kodlama teknikleri bulunmaktadır. İmgeler üzerinde başarılı sonuçlar veren vektör nicemleme yöntemi bu çalışmamızda imge dizilerinin kodlanmasında kullanılmıştır. Farklı vektör ve farklı kod kitabı boyutları için ölçümler alınmış bunların neticesinde durağan ve sabit imge çerçevelerinde diğer sıkıştırma algoritmalarına göre daha iyi neticeler elde edilmiştir. Hala tasarım aşamasındaki MPEG-7 algoritmasına benzer bir yapıya sahip olan bu sistem ile objelerin belirlenmesi, arka plan gibi obje temelli kodlama yapısına sahip bir yapı geliştirilmiştir.

### Abstract

Nowadays there are many video coding techniques. We used vector quantization technique that is giving successful results on images. We took results for different vector and code vectors. With these results for stationary and fixed image frames we obtain better results than the other algorithms. Our systems has similar feature to MPEG-7 that is still under development. This system is based on object coding such as background.

### 1. Giriş

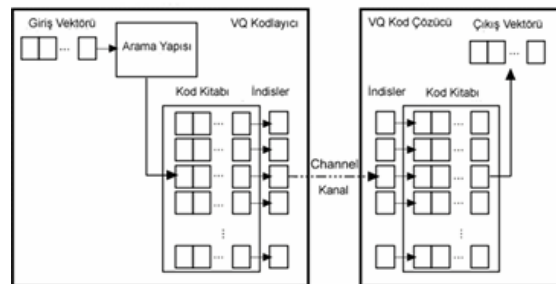
Günümüzün gelişen ses ve görüntü teknolojileri ile görüntü işleme ve sıkıştırma işlemleri önem kazanmıştır. İmge sıkıştırmasında JPEG, GIF ve TIFF ve görüntü dizileri için ise MPEG, DIVX gibi birçok kodlama algoritmaları bulunmaktadır. Vektör nicemleme(VQ) ise bunlardan bir tanesidir. Vektör Nicemlemenin pratik çalışmaları 1970'lerin sonu 1980'lerin başı gibi başlamıştır. Bundan sonrasında ise pratik yapılar ve tasarım algoritmaları geliştirilmiştir (Linde 1980).Bu çalışmaların sonucunda Vektör nicemleme başarılı bir şekilde imge, imge dizileri ve ses kodlamalarında kullanılmaya başlanmıştır. (Gersho 1982).

Video Kodlamada ise birçok algoritma bulunmasına karşın bunlardan en popüler olanı MPEG 'dir. "Moving Picture Expert Group" adı verilen bir gruba dayanmaktadır. Bu grup ISO/IEC olup 1988 yılında ses ve görüntü formatları üzerine çalışmaya başlamıştır. Her bir sıkıştırma standardı, özel uygulamalar ve bit oranları ile tasarlanmıştır. MPEG sıkıştırma ölçülü şekilde bir bit oran sıkıştırması gerçekleştirmektedir. MPEG video sıkıştırma algoritmaları genel olarak iki temel tekniğe dayanmaktadır. Tek düzelilik bilginin azaltılması amaçlı blok temelli hareket tahmini ve uzamsal tek düzeliliklerin azaltılması için ayrık kosinüs dönüşümü (DCT)'dir. Günümüzde kullanılmakta veya tasarım aşamasında olan beş çeşit MPEG standardı bulunmaktadır. Yapı sıkıştırma oranlarına ve yöntemlerine göre farklılıklar göstermektedir.

Bir Vektör Nicemleme işlemi üç kısımdan oluşmaktadır. Kod kitabı oluşturulması, kodlama ve kod çözme adımlarıdır. Vektör nicemleme işleminde kod kitabı hem kodlayıcı hemde kod çözücü kısımda bulunması gerekmektedir. Kodlanacak olan imge belirlenen boyutta vektör gruplarına bölünür bunlar 2x2, 4x4 vb. çeşitli matris boyutlarında olabilir. Her bir giriş vektör grubuna bağlı olarak ona en yakın kod kelimesi belirlenir ve bu belirlenen kod kelimesi kod kitabı içerisine yazılır. Bu işlem imge içerisindeki tüm vektör gruplarının kodlanmasına kadar devam eder. Kodlama işleminin tamamlanması ile beraber kod kitabı kod çözücü kısma iletilir arkasından ise kodlanan vektör gruplarının kod kitabındaki indisleri gönderilir. Kod çözücü kısmın görevi gelen indislere bağlı olarak kod kitabından vektör gruplarını alarak resmi oluşturmaktır. Bu kodlama ve kod çözme işleminin blok diyagramı şekilde görüldüğü gibidir.

Bir Vektör nicemleme işleminde en önemli noktalardan bir tanesi iyi bir kod kitabının oluşturulmasıdır. Kod kitabının kalitesi kod çözücü kısımda oluşturulacak olan görüntünün kalitesini belirlemektedir. Bu anlamda kod kitabının oluşturulmasına ilişkin çeşitli algoritmalar geliştirilmiş olup bunlardan en ünlüsü LBG (Linda , Buzo , Gray [1]) Algoritmasıdır.

Vektör nicemlemenin temel dezavantajı kaynağın istatistiksel davranışının çok iyi bir şekilde bilinmesi gereklidir. Bu sistem sabit olmayan kaynaklar için uygun değildir. Kodlayıcı ve kod çözücü, tanımlı olan kod kitabına bağlı olarak kodlama ve kod çözme işlemlerini gerçekleştirir. Bundan dolayı kodlayıcı kod kitabının içeriğini bağımsız olarak değiştiremez. Değiştirilmesi durumunda ise kod çözücü eşzamanlı olarak kod kitabındaki değişikliklerden haberdar edilmelidir. Kod kitabındaki değişikliklerin bildirilebilmesi için özel bir haberleşme yapısına ihtiyaç vardır. Bu iletilen ek bilgiye kısım bilgisi adı verilir. Şekil 1'de vektör nicemleme ile kısım bilgi yapısı gösterilmektedir. Bu yapıya örnek vektör nicemleme, Adaptif (uyarlamalı) vektör nicemleme (AVQ) gösterilebilir.



Şekil 1 Vektör Nicemleme Yapısı

## 2. Temel Tanımlamalar ve LBG Algoritması

Bir Vektör Niceleme sistemi, N vektörlü  $c_i \hat{I} \ i^L$  'den ,  $I = \{0, \dots, N-1\}$  indeks dizisi ve N kod kelimesinin  $V$  dizisi ile tanımlanan değişken uzunluklu kod'dan (VLC) oluşan bir  $C = \{c_0, \dots, c_{N-1}\}$  kod kitabına dayanır. Bu diziler arasındaki bağlantı  $\alpha$  vektör kodlayıcı ,  $\beta$  vektör kod çözücü ve bire bir indeks kodlayıcı  $\gamma$  ile tanımlanır.

- $\alpha : i^L \rightarrow I$  vektörü indekse eşler
- $\beta : I \rightarrow C$  bir kod kitabı içerisindeki indeksi ise vektöre eşleştirir.
- $\gamma : I \rightarrow V$  İndeksini bir değişken uzunluklu kod (VLC) kelimesine eşler

$\alpha : i^L \rightarrow I$  kodlayıcı  $\gamma \circ \alpha$  ile verilir ve  $\beta : I \rightarrow C$  ise  $\beta \circ \alpha$  ile tanımlıdır.  $Q = i^L \rightarrow C$  Şeklinde gösterilen niceleme şöyle tanımlanabilir;

$$Q = \beta \circ \alpha = \beta \circ \gamma^{-1} \circ \gamma \circ \alpha \quad (2.1)$$

$$x = (x^{(0)}, \dots, x^{(L-1)})^t \quad (2.2)$$

ve

$$Q(x) = \hat{x} = (\hat{x}^{(0)}, \dots, \hat{x}^{(L-1)})^t \quad (2.3)$$

Arasındaki bozulmayı belirlemek için bir bozulma ölçüsü  $d(x, \hat{x})$  gereklidir. Genellikle  $d(\dots)$  şöyle tanımlanır.

$$d(x, \hat{x}) = \|x - \hat{x}\|_2^2 = \sum_{i=0}^{L-1} (x^{(i)} - \hat{x}^{(i)})^2 \quad (2.4)$$

Bozulma ölçüsünü de belirledikten sonra, vektör kodlayıcı  $\alpha$  genellikle en yakın komşu (NN) kuralı ile gerçekleştirilir. Bu kurala göre Vektör kodlayıcı, Kod kitabı içerisindeki vektör indeksini en küçük bozulmaya sahip olanından seçer  $\alpha(x) = \arg \min_{i \in I} d(x, \beta(i))$ . Bu kuralın uygulanması için en yakın değer iki tane olması durumunda bunlardan bir tanesini seçecek bir kabul yapılır.

Vektör niceleyicisinin performansını değerlendire bilmek için, x vektörlerini rasgele değişken X ile tanımlandığını ve olasılık yoğunluk fonksiyonu ise  $f_X$  olduğu kabul edilsin. Ortalama bozulma D ve Ortalama bithızı da R ile gösterilsin bu durumda;

$$D = E[d(X, Q(X))] = \int_{i^L} d(x, Q(x)) f_X(x) dx \quad (2.5)$$

ve

$$R = E[|\gamma(\alpha(X))|] = \int_{i^L} |\gamma(\alpha(x))| f_X(x) dx \quad (2.6)$$

Denklem 2.6 'da tanımlanan  $|\gamma(i)|$  ,  $\gamma(i)$  'nin bit formatında VLC uzunluğudur. E ise beklenti operatörüdür. Vektör kodlayıcı  $i^L$  uzayını N parçalı hücrelere böler  $R_i = \{x \in i^L : \alpha(x) = i\}$  .Her bir bölüm

hücrenin içinde bulunduğu bölmenin olasılığına  $p_i$  ve bölmenin bozulması  $d_i$  ise şu şekilde tanımlanır;

$$p_i = \text{prob}(X \in R_i) = \int_{R_i} f_X(x) dx \quad (2.7)$$

ve

$$d_i = E[d(X, Q(X)) : X \in R_i] \quad (2.8)$$

Bu terimleri kullanarak bölmeler için bozulma ;

$$D_i = d_i \cdot p_i \quad (2.9)$$

LBG 'nin açıklana bilmesi için 2 duruma gerek vardır.

**Durum 1 :** (En yakın Komşu Durumu ) Oluşturulmuş bir kod kitabı için en uygun bölüm hücreleri  $R_i$   $R_i \subset \{x : d(x, \beta(j)), \forall j \in I\}$  'yi tamamlar.

**Durum 2 :** (Kütle merkez konumları) Verilen bir bölüm için  $\{R_i : i \in I\}$  , En iyi kod kitabı vektörleri  $\text{cent}(R) = E[X : X \in R]$  Beklenti operatörü E ile  $\beta(i) = \text{cent}(R_i)$  'yi tamamlamaktadır.

Birinci durumda vektör kodlayıcı  $\alpha$  sabit bir vektör kod çözücüsü  $\beta$  'nın olduğu kabul edilerek formül ize edilmiştir.2. durumda ise  $\beta$  kod çözücüsü sabit bir  $\alpha$  kodlayıcısının olduğu kabul edilerek kodlanmıştır. Bu kabuller şu iterasyon adımlarını önerir: Sabit bir  $C_n$  kod

kitabı ile başlanır. Vektör kodlayıcı  $\alpha_{n+1}$  en yakın komşu kuralına bağlı olarak bölüm hücreleri  $R_i$  'leri belirler.

Sonrasında yeni bir kod kitabı  $C_{n+1}$  ,  $\beta_{n+1}(i) = \text{cent}(R_i)$  'nin belirlenmesi ile hesaplanabilir.

Bu iterasyon adımları Lloyd İterasyonu olarak adlandırılır. Yukarıda gösterilen iterasyon formu  $f_X(x)$  'in bulunduğu

ve  $R_i$  bölüm hücrelerinin analitik olarak belirlendiğini kabul eder. Genelde böyle bir şey imkânsızdır. Olasılık dağılımı eğitim vektör dizileri ile bulunur. Bundan dolayı Lloyd iterasyonun'da kütle merkezleri hesaplanması ve  $R_i$  'lerin hesaplanması ayrı hesaplama ile tayin edilir.

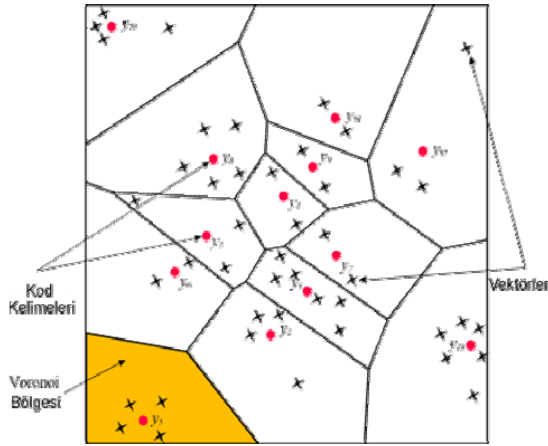
$$R_i = \{x \in B : d(x, \beta(i)) \leq d(x, \beta(j)), \forall j \in I\}$$

ve

$$\text{cent}(R_i) = \frac{1}{|R_i|} \sum_{x \in R_i} x \quad (2.10)$$

Bölüm bozulması  $d_i$  için bölüm olasılığı  $p_i$  ve kısmi bozulma  $D_i$  'yi benzer bir yolla bulabiliriz. Bu da bize gösterirki ortalama bozulma D bir Lloyd iterasyonun'dan sonra artmaz. Bundan dolayı Lloyd iterasyonunun sonrasında gelen uygulamalar artmayan ortalama bozulma ile birlikte  $C_m$  kod kitapları dizisini oluşturmaktadır. Bu iterasyon algoritmasına Genellenmiş Lloyd algoritması (GLA veya LBG) adı verilir. Eğer  $D_m$  ortalama bozulma değişimi yeteri kadar küçükse iterasyon durmaktadır. Bu bir

eğitim dizisi'nden sonra her zaman gerçekleşmektedir. Buda bir minimum değerine yaklaştığını gösterir.



Şekil 2 VQ'da Vektör ve Kod Kelimeleri

### Algoritma 1 : Genelleştirilmiş Lloyd Algoritması

Verilenler :

Başlangıç Kod Kitabı  $C_0$ ,

Vektör Çözücü  $\beta_0$ ,

Eğitim Dizisi B,

Durma Eşik Değeri,  $\epsilon$

İlk Vektör niceleyicisinin  $D_0$  ortalama dağılımı

$m=0$

Döngü

$$\alpha_{m+1}(x) = \arg \min_n d(x, \beta_m(n))$$

$$R_n \leftarrow \{x \in B : \alpha_{m+1}(x) = n\}, \forall n \in I$$

$$\beta_{m+1}(n) \leftarrow \frac{1}{|R_n|} \sum_{x \in R_n} x, \forall n \in I$$

$$C_{m+1} \leftarrow \{\beta_{m+1}(n) : n \in I\}$$

$$D_{m+1} \leftarrow \frac{1}{|B|} \sum_{x \in B} d(x, \beta_{m+1}(\alpha_{m+1}(x)))$$

$$m = m + 1$$

$$\frac{D_m - D_{m-1}}{D_m} < \epsilon \text{ Oluncaya Kadar Dön}$$

Bu bozulma dizileri küçük bir değerde bir araya gelmektedirler. Bundan dolayı  $C_0$  seçimi çok önemlidir. En basit başlatma tekniği N adet kod kitabı vektörlerini test dizisi B 'den seçmektir. LBG ortalama bozulma D 'yi optimize eder ve bunu yaparken ortalama oranı R'yi dikkate almaz. Bundan başka indeks kodlayıcısının sabit uzunlukta bir kodlayıcı olduğu kabul edilir. Bundan dolayı R iterasyon sırasında değişmez. Optimizasyon sırasında R önemli ise bu algoritma tipine Entropi Kısıtlı Vektör Niceleme (EKNV) adı verilir.

### 3. Vektör Nicemleme ile Video Kodlama

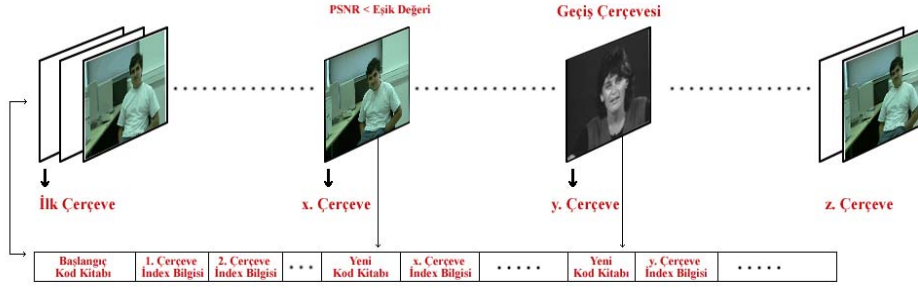
Sistemimizde farklı imge dizileri ve farklı boyutlarda kod kitapları için çalışmalarımız doğrulanmış ve sonuçları aşağıda verilmiştir. Sistemimizin yapısı şu aşamalardan oluşmaktadır.

- İlk imge çerçevesinin kodlanması ve kod kitabının oluşturulması
- İlk frame ile karşılaştırılarak PSNR değeri eşik değerinin altında değil ise aynı kod kitabını kullanarak imgenin kodlanması
- Eğer görüntüde ani değişimler olması ve buna bağlı olarak PSNR değeri eşik değerinin altına düşmesi durumunda yeni bir kod kitabının oluşturulması ve bunun alıcıya ulaştırılması.

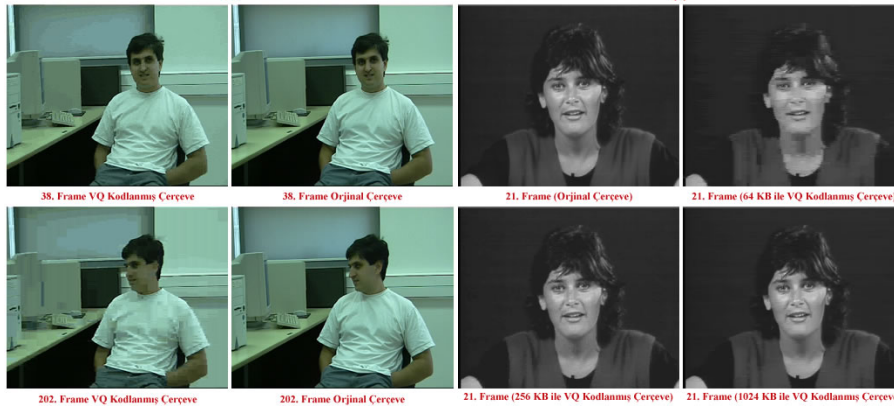
Sistemimiz temel olarak bu yapıya dayanmaktadır. Yapılan ölçümlerde görüntüde değişimin az olduğu imge dizilerinde 13 ile 14 frame'de tekrar bir kod kitabı oluşturulmaya ihtiyaç duyulduğu görülmüştür. Tabi bu durağan ve değişimin çok az olduğu görüntü dizileri için geçerlidir. Ani değişimlerin olduğu durumlarda her defasında kod kitabının yenilenmesi gerekmektedir. Diğer görüntü dizisi kodlama algoritmaları ile karşılaştırıldığında durağan görüntülerde daha iyi sonuçlar verdiği gözlenmiştir. Temelde bir MPEG formatında bir yapı ile karşılaştırmak gerekirse vektör nicemlemenin en büyük avantajı çok kapsamlı bir görüntüleme sistemine ve program yapısına ihtiyaç duymadan basit bir indeks bilgisine bağlı olarak vektör matrislerini kod kitabından alarak görüntüyü oluşturma yapısına sahiptir. MPEG'e baktığımızda bir görüntünün kodlana bilmesi ve gösterilebilmesi için DCT dönüşümünden Zig zag kodlamaya kadar birçok uygulamadan geçmesi gerekmektedir. Buda bize Vektör nicemlemenin ne kadar basit bir ara yüz sistemi ile görüntüleme sunacağını bir göstergesidir. Bu işlemleri gerçekleştirirken daha düşük kaliteli bir görüntü kalitesi elde edilebilir bu ise kod kitabının boyutunun artırılması ve belirlenecek hassasiyet değerleri ile düzeltilebilir bir yapıdadır.

Tablo.1 Vektör Nicemleme ile MPEG Yapısının Karşılaştırılması

Kod Kitabı Boyutu	VQ			MPEG
	128	256	1024	-
Sıkıştırma Oranı	26.75	22,2	16.2	30
PSNR	29,8	31,1	33.8	25 - 26
İşlemler				
VQ	<ul style="list-style-type: none"> <li>• Kod Kitabının Oluşturulması ve Eğitim</li> <li>• İmge Çerçevesindeki Blokların Kod Kitabına bağlı olarak indeks değerlerinin atanması</li> <li>• Kod kitabı ve kodlanan imge çerçevelerinin bir dizi haline getirilmesi</li> </ul>			
MPEG	<ul style="list-style-type: none"> <li>• Ayrık Kosinüs Dönüşümü</li> <li>• Prediction / Blok Uyumlaması</li> <li>• Nicemleme</li> <li>• Zig Zag Kodlama</li> <li>• Entropi Kodlama</li> </ul>			



Şekil-3 Vektör Nicemleme ile gerçekleştirilmiş bilgi dizisi



Şekil-4 (a) 38. Frame'in 2x2 Boyutlu Kod Kitap Boyutu: 256 202. Frame'in 4x4 Boyutlu Kod Kitap Boyutu: 128 Sonuçları  
(b) 21. Frame'in 4x4 Boyutlu 64 ve 1024 Kod Kitabı ile Kodlama Sonuçları

#### 4. Sonuçlar ve Tartışmalar

Yaptığımız bu çalışmada çeşitli imge dizilerine bu yöntem uygulanmıştır. Şekil 4(a) imge dizisi için çeşitli vektör boyutlarında ve farklı büyüklükteki kod kitapları için uygulama yapılmıştır. Bu uygulamaların sonuçları Tablo 2 de gösterilmiştir.

Tablo 2 Ölçüm Sonuçları

İmge Dizisi	Frame	Matris Boyutu	Kod Kitabı	PSNR	Sıkış. Oranı
Laboratuvar Frame	38	2x2	64	29.2	15,378
			128	30.5	12,85
			256	34.6	10,70
Laboratuvar Frame	202	4x4	64	27.18	38,871
			128	29,7	26,020
			256	31.6	16,329
Miss America	21	2x2	64	38,6	15,402
		4x4	128	40.8	26,176
			256	42.6	16,376

Elde edilen veriler neticesinde imgelerin kodlamasında kod kitabının oluşturulmasının görüntü kalitesini belirlemede önemli bir özellik olduğu görülmüştür. Kod kitabındaki kod kelime sayısının artmasına bağlı olarak görüntü kalitesi artmakta fakat buna bağlı olarak sıkıştırma oranı düşmektedir. Böyle bir yapının en önemli özelliklerinden biri Jpeg ve Mpeg kod çözme işlemi için gerekli işlemci performansı yüksektir. Vektör nicemleme için ise düşük performanslı işlemciler ile video dizi kodları çözümlenebilir. Bu vektör nicemlemenin imge dizisine uygulanmasında ise yüksek sıkıştırma oranına sahip bir sistem elde edilebilmektedir. Aynı zamanda böyle bir

yapı kod çözücünün basitliği sayesinde mobil teknolojiden web uygulamalarına kadar birçok uygulamada imgelerin ve imge dizilerinin kodlanmasında kullanılabilir. Yeni geliştirilmekte olan MPEG-7 kodlama yapısına benzer bir yapıya sahip olup çerçeveler içerisinde obje tanımlama ve imge içerisine nesne yerleştirme gibi işlemlere olanak sağlamaktadır.

Gelecek çalışmalarımızda İmgelerin Vektör nicemleme ile kodlaması ve eğitimi için geçen sürelerin iyileştirilmesi, kod kitabının kalitesinin artırılarak daha yüksek PSNR değerlerinin elde edilmesi ve imge dizisi için oluşturulan kodlanmış bilgi dizilerinde iki değişim çerçevesi arasındaki çerçevelerdeki sadece değişim bilgisini göndererek daha iyi sıkıştırma elde edilmesi planlanmaktadır.

#### 5. Kaynakça

- [1] Y. Linde, A. Buzo, & R. M. Gray, " An Algorithm for Vector Quantization Design ," IEEE Transactions on Communication, v. COM-208, pp.84-95, 1980.
- [2] C. H. Hsieh, "DCT-based codebook design for vector quantization of images," IEEE Trans. Circuits and Systems for Video Technology, v. 2, no. 4, 1992
- [3] A. Gersho and B. Ramamurthi. Image coding using vector quantization In ICASSP'82, volume 1, pages 428{431, May 1982.
- [4]Khalil H. Predictive Multistage Vector Quantizer Design Using Asymptotic Closed Loop Optimization IEEE November 2001 Vol No: 10 Page 1765
- [5] Ricardo L. De Queiroz Senior Member IEEE , Very Fast JPEG Compression Using Hierarchical Vector Quantization IEEE May 2000 Vol : 7 Page 97
- [6] Yang H. Source Channel Prediction In Error resilient video coding IEEE 2003 Page :233