

Bitişik elemanlardan oluşmayan vektör gösterilimi ve öngörülü vektör kodlama ile görüntü sıkıştırma:

Mehmet YAKUT
Kocaeli Üniversitesi Mühendislik fakültesi
Elektronik ve Haberleşme Mühendisliği Bölümü
Veziroğlu Kampüsü 41040 İzmit
myakut@kou.edu.tr

Abstract: In this study different vector formation structure is proposed for high correlation between successive vectors. Switched codebook Predictive VQ algorithm is applied using this high correlation where starting vector is also used as codebook selector. Multiscale VQ representation is achieved while compression. Proposed vector formation has attractive structure for image compression. Different vector formation structures can be proposed depending on the application area.

I. Giriş:

Haberleşme tekniklerindeki gelişmeler ve çoklu ortam uygulamalarının yaygınlaşması sonucu görüntülerin iletimi ve depolanması ihtiyacı da buna paralel olarak artmıştır. Bilgisayar ve yarıiletken teknolojisindeki hızlı gelişmeler de görüntülerin iletimi ve depolanması ile ilgili daha etkin sıkıştırma tekniklerinin uygulanmasına imkan vermiştir. Sıkıştırma tekniklerinde gözlemlenen bu iyileşme iletim ve depolama ortamlarının daha verimli kullanılabilmesine imkan sağlamıştır. Vektör kodlama da görüntü sıkıştırmadaki etkinliği nedeniyle yaygın kullanım alanı bulmuş ve ilgi çekiciliği sürekli devam eden bir konu olarak karşımıza çıkmaktadır.

Vektör kodlama uygulamalarında genellikle birbirlerine bitişik kare bloklar seçilmektedir. Bu seçim şekli ardışıl vektörler arasındaki ilişkiyi dolayısıyla öngörülebilirliği azaltmaktadır. Bu çalışmada ise ardışıl vektör bileşenleri arasında sıkı ilişki olan bir tarama düzeni ve vektör seçimi düşünülmüştür. Böylece ardışıl vektörler kodlanırken bir önceki vektörden kodlanacak vektörün öngörülmesi kolay olacaktır. Öngörünün doğruluğu artacağı için kodlanacak hata vektörü daha az sayıdaki kod vektör yardımı ile kodlanabilecektir. Ayrıca başlangıç vektörü izleyen vektörler için kullanılacak kod tablosunu seçici özelliğe sahiptir. Önerilen yöntemin genel yapısı değişik bir vektör tarama düzeni üzerinden anahtarlamalı ve öngörülü vektör kodlayıcı şeklinde açıklanabilir.

II. Sayısal kuantalama ve vektör kodlama:

Kuantalayıcı temelde sürekli rasgele değişkeni ayrık rasgele değişkene dönüştüren mekanizmadır. Kuantalayıcı ile sürekli giriş değişkeninin ayrık değişkenlerden hangi değeri alacağı ve daha sonra

tekrar hangi değere dönüştürüleceğine karar verilir[1][5]. Giriş değişkeni olasılık dağılımı ve en iyileme kuralı verildiğinde, ortaya çıkacak bozulmayı en küçük yapacak, en uygun geçiş ve yeniden elde etme seviyelerinin belirlenmesi en uygun kuantalayıcı tasarımı olarak adlandırılabilir[5]. Vektör kuantalama (blok kodlama) bir boyutta yapılan kuantalama işleminin daha çok boyuta genelleştirilmesi şeklinde açıklanabilir[1]. Görüntü bilgisinde noktasal değerlerin kodlanması yerine veri bloklarının vektörler halinde kuantalanmasıyla daha yüksek sıkıştırma oranlarına erişilebilir[2][3]. Vektör kuantalama görüntü bilgisindeki veri bloklarının (vektörler) herbirinin daha önceden belli bir kurala göre belirlenmiş standart veri bloğu içindeki (kod tablosu) bir eleman ile temsil edilmesi şeklindedir. Vektör kuantalamada görüntü vektör uzayı daha az sayıda kod ile temsil edilen bir diziyi dönüştürüldüğü için kayıplı bir sistem olduğu açıktır. Önceki vektör kuantalama yöntemlerinde bozulmanın küçültülmesi ve sistemin gerçekleştirilebilirliğinin artırılması için karmaşıklık, gerekli bellek ve kabul edilebilir bozulmalar gözönünde bulundurulmuştur[2]. Daha sonraları geribesleme kullanılarak bellekli yapı ile vektörlerin kestirimine dayalı kuantalayıcılar geliştirilmiştir. Geribeslemeli yapılar VLSI uygulamaları için oldukça uygundur[4].

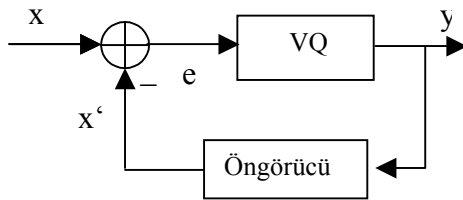
Temel olarak vektör kuantalayıcı kodlama ve kod çözmede iki eşleme işlemi gerçekleştirir. k boyutlu bir kodlayıcıda herbir giriş vektörünü $x=(x_0, x_1, \dots, x_{k-1})$ kod tablosundaki M tane değerden birisi ile en az bozulma olacak şekilde eşleştirir. Benzer şekilde k boyutlu kod çözücü de kanaldan gelen eşleştirme bilgisini gösteren kod tablosu indeksi ile kod tablosundaki yerini eşleştirir. Kod tablosunda M eleman bulunursa kuantalayıcı hızı vektör başına bit olarak $R=\log_2 M$ olarak verilir. Örnek başına bit şeklindeki hız ise $r=R/k$ dir. Sayısal kuantalamadan farklı olarak vektör kuantalamada kesirli iletim hızları elde edilebilmektedir[6].

VQ tasarımı için önemli aşamalardan birincisi başlangıç kod tablosunun seçilmesidir. Başlangıç kod tablosu rasgele (random), budama (pruning), en yakın komşuların birleştirilmesi (PNN), ayrıştırma (splitting) gibi yöntemlerle seçilebilir[10]. Başlangıçta seçilen kod tablosu yeteri kadar iyi ise, kod tablosunu iyileştirmek kolay olur ya da hiç gerekmez. Değişik görüntü işleme konuları (şekil tanıma, sıkıştırma gibi) için çok çeşitli kod tablosu üretme

yöntemleri geliştirilmiştir[3]. Genelde vektör kuantalayıcı tasarlanırken kod tablosu başlangıç değerleri en yakın komşuluk kuralı ile belirlenmiş bölgelere göre seçilir, daha sonra bu bölmeler için yeni en uygun kod tablosu eğitimle elde edilir[10]. Elde edilen yeni kod tablosunun bozulma miktarı bir önceki durumdan daha az olur veya değişmez. Kod tablosu iyileştirme için kullanılabilecek kapalı formda çözüm yoktur. Genelde iyileştirme yöntemleri ard arda uygulanarak iki adım arası iyileşme belirsiz oluncaya kadar devam edilir[6][10]. Daha önce anlatılan kod vektör tablosu oluşturma işlemleri genelde en iyi kod vektör tablosu ile sonuçlanmaz, ancak elde edilen kod vektör tabloları Lloyd ardışıl işlemleri için başlangıç kod tablosu olarak seçilerek daha hızlı biçimde en iyi kod tablosuna ulaşılabilir. Bu yöntem ile kod tablosu $C_m = \{y_i ; i=1, \dots, N\}$ verildiğinde, giriş eğitim seti için en yakın komşuluk kuralını kullanarak kuantalama bölgeleri en iyi olacak şekilde bölmelenir. Yani x girişi ile y_i ler arasında birbirine en yakın olan y değeri seçilerek x bu bölmeye atanır. Yeni verilerden centroid kuralı kullanılarak yeni kod tablosu C_{m+1} daha önceki bölgelerle atanan giriş vektörlerinin orta noktalarında olacak şekilde belirlenir $C_{m+1} = \{\text{cent}(R_i); i=0, \dots, N\}$ [7].

III. Öngörü ile vektör kodlama:

Görüntü sıkıştırmak için belleksiz vektör kodlama kullanılırken birbirlerini izleyen vektörler istatistik açıdan bağımlılık gösterir. Buna göre vektörler arasındaki bu ilişki kullanılarak kodlayıcı gerçekleştirilirse daha iyi performans elde edilir[3]. Öngörü ile vektör kodlamada, görüntü bilgisi alt bloklara bölünür ve komşu bloklar arası ilişki yardımıyla yeni vektörün öngörüsü yapılır. Vektöre ilişkin öngörü yapıldıktan sonra asıl bloğu kodlamak yerine, öngörülen değerle asıl blok arasındaki farkı kodlayabiliriz. Kodlayıcıda yapılanın benzeri şekilde alıcı tarafta da bloğa ilişkin öngörü yapılarak gelen fark bilgisi ile toplanır ve asıl bloğun belli yaklaşıklıkla kodlanması sağlanmış olur. Genel vektör kodlama kuralı olarak vektör boyutu



Şekil 1. Öngörülü vektör kuantalayıcı

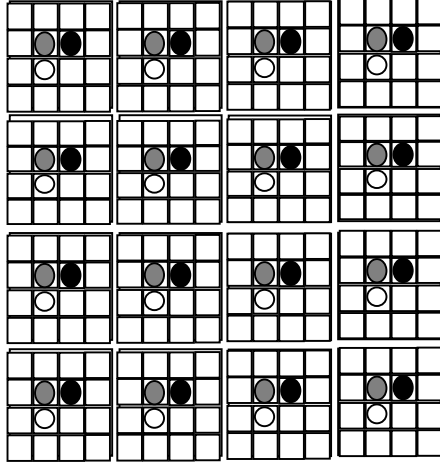
büyükçe kodlama performansı artmaktadır, ancak bu durum veri bloğunun öngörüsü söz konusu olduğunda geçerliliğini yitirir[8]. Vektör boyutu büyükçe bloklar arası ilişki azalacağından öngörü işleminin duyarlılığı azalacak ve kodlanması gereken hata vektörü genliği büyüterek vektör kodlama

performansı düşecektir. Vektör bileşenlerinin öngörüsünde basitliği ve yeterli kalitenin elde edilmesi sebebiyle doğrusal öngörü kullanılır. Çoğu uygulamada birinci derece doğrusal öngörü kullanılarak çok az karmaşıklıkla oldukça iyi sonuçlar elde edilmiştir[8]. Vektör blokları arasında vektör bileşenlerinin öngörüsü için kullanılabilecek değişik yapılar bulunabilir.

IV. Önerilen yöntem:

Vektör kuantalama uygulamalarında olabildiğince karmaşık vektör kodlama yöntemleri geliştirilmiş olmasına rağmen vektör blokları genellikle kare seçilmiş veya ender olarak satır şeklinde ele alınmıştır. Burada farklı bir vektör bloğu seçimi önerilmiştir. Vektör bloğu seçilirken şekil 2. de görüldüğü gibi birbirlerinden n adım uzaklıktaki bileşenler yardımıyla vektör oluşturulabilir. İlk bakışta birbirleriyle çok az ilişkili bileşenlere sahip bir vektör elde edilmiş gibi görülmeye rağmen asıl görüntü piramit kodlamadaki gibi az elemanla ve iyi bir yaklaşıklıkla temsil edilirse (seyrek örneklenip vektör kodlanarak), bu az ilişkili vektör bileşenlerini yeterli duyarlılıkta öngörmek mümkün olabilir. Ayrıca başlangıç vektörü sonraki vektörler için kod tablosu seçici özelliğe de sahiptir. Örnek olarak verilen düşük çözünürlüklü görüntüden alınan kare vektör bloğunun bileşenleri, asıl görüntüden alınan vektör bloğunun ilk bileşenleri gibi düşünülebilir. Burada seçilen başlangıç vektörü kod tablosu seçici olarak kullanılabilir. Doğal olarak asıl vektör ile seyrek örneklenip vektör kodlanmış gösterim arasında farklılık olacaktır. Kodlama işleminde fark vektör ve seyrek örneklenmiş vektör kullanılarak asıl vektöre yakın değer elde edilebilir. Bir sonraki adımda kodlanacak vektörün herbir bileşeni bir önceki vektör bileşenlerinin ayrı ayrı herbirine komşu olduğundan yeni vektör bileşenlerinin öngörüsü oldukça yaklaşık olacaktır. Yöntemin önemli özelliği bu adımda ortaya çıkmaktadır. Birbirlerine komşu bileşenler çok sıkı ilişkilidir. Bu yüzden öngörülmesi kolay olmaktadır. Oysa ki yaygın kullanılan yöntemlerde kare bloklar seçilmekte ve bitişik vektörler arasında bile ilişki oldukça azalmaktadır. Bu da vektör kodlamanın başarımını azaltmakta ya da çok büyük kodvektör tablolarına ihtiyaç duyulmasına neden olmaktadır[9]. Önerilen yöntemde özetle başarılı öngörü yapabilmek için ardışıl vektör bileşenleri arasındaki ilişkiyi büyük tutacak bir vektör yapısı vardır. Yöntemin diğer bir avantajı görüntü arşivlemede kullanılabilir. Düşük çözünürlüklü görüntü yardımıyla arşiv içinde aranan görüntüye hızlı erişim mümkün olacaktır. Yöntemin en önemli dezavantajı 16x16 bloklar arası bozulmalar ortaya çıkabilecektir. Bu durum düşük çözünürlüklü vektör kodlanmış görüntünün çok iyi elde edilmesiyle önlenemez. Vektör oluşturma şekli vektör elemanlarının öngörüsünde bir boyutlu öngörü kullanılmasına olanak tanımaktadır. Seyrek

- örneklenmiş görüntü bilgisinin kodlamada
- seyrek örnekleme ve vektör kodlanmış noktalar (başlangıç vektörü)
 - başlangıç vektöründen öngörülerek kodlanacak vektör
 - bir önceki vektörden öngörülerek kodlanacak vektör



Şekil 2. Önerilen vektör oluşturma yapısı

kullanılması özellikle görüntü veritabanlarında bazı özelliklere sahip görüntülerin aranmasında hız kazandıracaktır[9][12]. Benzer şekilde iletişim esnasında düşük çözünürlüklü görüntü ile çalışarak isteğe bağlı yüksek çözünürlük kullanımına olanak sağlanacaktır[11]. Kodlanacak görüntüden isteğe bağlı bir şekilde (seyrek örnekleme, alçak geçiren filtreleme v.b.g.) düşük çözünürlüklü görüntü (DÇG) elde edilir. Bu görüntü asıl görüntüyü belli bir yaklaşıklıkla temsil edecek nitelikte olmalıdır. Asıl görüntünün kodlanması için DÇG'den seçilen vektör bloğu çekirdek vektör gibi değerlendirilebilir. İleri adımlarda bu çekirdek vektör bileşenleri etrafında spiral şeklinde tarama yapılarak komşu vektörler oluşturulur. Taramanın bu şekilde yapılması ardışıl vektör bileşenleri arasındaki ilişkiyi en büyük yapmaktadır. Kodlanacak vektörleri oluşturma yöntemi bu şekilde seçildikten sonra çekirdek vektör olarak adlandırdığımız vektörle kodlanması istenen bir sonraki vektör arasındaki fark vektör kodlanır. Kodlama işleminde fark vektör kod tablosundan kodlanacak fark vektöre en yakın kod vektör bulunur. Sıra üçüncü vektörün kodlanmasına geldiğinde çekirdek vektör ve bir önceki vektör bileşenlerinin belli ağırlıklarla elde edilen bir kombinasyonu öngörülen vektör olarak seçilir. Çekirdek vektörün yakın komşuluğu için çekirdek vektör bileşenleri daha ağırlıklı alınır. Çekirdek vektörden uzaklaştıkça çekirdek vektörün öngörü vektöründeki ağırlığı azaltılıp ağırlık bir önceki kodlanan vektöre kaydırılır[3]. Ağırlıklar seçilirken çekirdek vektör ile kodlanacak diğer vektörler arası çıkarılmalıdır. Ancak genel görüntü istatistikleri kullanılarak ağırlıkların

başlangıç değerleri seçilebilir. Gerçekte bu başlangıç değerleri ile fark vektör kod tablosu ortalama karesel hata en küçük olacak şekilde eğitilir. Daha sonra yeni ağırlık katsayıları belirlenir ve kod tablosu eğitimi yinelenir. Bu işlem ortalama karesel hatadaki azalma belirsiz bir değere ulaşana kadar sürdürülür. Yapılan ön çalışmada seyrek örneklenmiş görüntü üzerinden asıl görüntü önerilen tarama şekline göre vektör kodlanırken seçilen ilk 16x16 blok için kodvektör tablosundaki bazı vektörler çok sayıda kullanılırken, bazılarının hiç kullanılmamakta olduğu gözlenmiştir (bkz tablo 1).

Vektör	1. blok	2.blok	7.blok	11.blok
0				
1			2	
2		3		1
3				
4	1		5	1
5	3	2		
6		4		
7				
8				2
9	1	1		
10		2		
11	4			
12			1	
13		3		
14				1
15	3			
16	1		2	
17			1	
18				
19				
20				1
21				
22	2			2
23				
24				
25				1
26				2
27			2	
28			2	3
29				
30				
31				1

Tablo 1: Standart VQ kullanılması durumunda kod vektörlerin kullanım oranları

Bu açıkça göstermektedir ki seyrek verideki vektörler o bölgeye ilişkin belirgin bir özelliği temsil ettiğinden asıl vektör bloğunun kodlanmasında bu özelliğe yakın olan kod vektörler daha fazla kazanmaktadır. Bu gözlem sonucu seyrek verideki özellikleri temsil eden kod vektörlere göre asıl kod vektör tablolarının oluşturulmasının fayda sağlayacağı ve daha iyi kodlama elde edilebileceği açıkça fark edilmektedir[2][9]. Ayrıca seyrek veride bulunan kod vektör sayısı kadar asıl vektörler için kod vektör tablosu kullanmak gerekli sabit belleği arttırmak dışında hiçbir işlemsel külfet getirmemektedir. Bu ilave ile yöntemin daha avantajlı olacağı açıktır. Tablo 1. de seçilen büyük vektör bloklarına karşılık tekli kod vektör tablosunda kullanılan vektörler ve kullanım sıklıkları örnek olarak verilmiştir. (Burada kullanılan tekli kod vektör tablosu tüm eğitim seti ve kodlanacak

görüntünün birlikte kullanılmasıyla elde edilmiş 32 elemanlı kod vektör tablosudur. 32 elemanlı kod vektör tablosu ile elde edilen kodlama sonuçları genelde kabul edilemeyecek kadar kötü olmakla birlikte kod vektörlerin kullanım sıklıkları açısından yeterli fikri vermektedir.)



Şekil 3. a) 256x256 8bit Lena b) 0.125bpp c) 0.25bpp VQ kodlanmış Lena

V. Sonuçlar:

Önerilen yöntem kullanılarak 4 farklı görüntü dosyası kullanılarak (Lady, Ape, Cameraman, Lena) kod tabloları eğitilmiş ve 16x16 lık bloklar üzerinden 4x4 lük vektörlerle kodlama yapılmıştır. Anahtarlamalı VQ özelliğini gözlemek için 16 adet anahtarlanabilir kod tablosu kullanılmıştır. Lena görüntüsü 0.125 bpp ile kodlandığında PSNR = 27.65 dB 0.25bpp ile kodlandığında PSNR = 28.31 dB işaret gürültü oranı ile yeniden elde edilmiştir. Önerilen yöntemle görsel açıdan yeterli kalitede görüntü elde edilmekte ve bunun düşük çözünürlüklü temsili ile hızlı arama işlemleri gerçekleştirilerek gerektiğinde daha yüksek çözünürlük seviyelerine ulaşılabilir. Hali hazırda uygulanan yapıda kaba ve ince olarak nitelenebilecek iki çözünürlük seviyesi vardır. Vektör oluşturma şekli üzerinde düzenlemeler yaparak ikiden fazla çözünürlük seviyeleri de elde edilebilir.



Şekil 4. Seyrek örneklenmiş 64x64 8bit Lena (Bu görüntü 4x4 VQ ile kodlanarak başlangıç ve seçici vektörler elde edilmiştir)

Referanslar:

- [1]. A. K. Jain "Image data compression; A review" Proceedings IEEE March 1981 p349-389
- [2]. M. Kunt, A. Ikonopoulou, M. Kocher " Second generation image coding techniques " Proceedings IEEE vol.73 no.4 April 1985
- [3]. N. M. Nasrabadi, R. A. King "Image coding using vector quantisation: A review" IEEE transactions on Communications vol.36 no.8 Aug. 1988
- [4]. C. F. Barnes, S. A. Rizvi, N. M. Nasrabadi "Advances in residual vector quantisation" IEEE transactions on image processing Feb. 1996
- [5]. R. M. Gray "Vector quantisation" IEEE ASSP magazine 1984
- [6]. Y.Linde A. Buzo R.M. Gray "An algorithm for VQ design" IEEE transactions on Communications Jan. 1980
- [7]. S. Panchanatan M. Goldberg "Mini-max algorithm for image adaptive VQ" IEEE Proceedings Feb. 1991
- [8]. H.M. Harry J.W. Woods "Predictive vector quantisation of images" IEEE transactions on Communications Nov. 1985
- [9]. B. Wegmann C. Zetzche "Feature spesific vector quantisation of images" IEEE transactions on image processing Feb. 1996
- [10]. A. Gersho, R. M. Gray "Vector quantisation and signal compression" Kluwer academic publishers 1992
- [11]. J. S. Lim "2-D signal and image processing" Prentice Hall 1990