

SÖZDE FAZ KULLANARAK DAYANIKLI BİR HAREKET TAHMİNİ

Mustafa ATALAR ve Hasan Şakir BİLGE

Bilgisayar Mühendisliği Bölümü, Mühendislik Mimarlık Fakültesi, Gazi Üniversitesi, 06570, Maltepe-Ankara
matalar@yahoo.com, bilge@gazi.edu.tr

(Geliş/Received: 25.06.2008 ; Kabul/Accepted: 01.12.2008)

ÖZET

Hareket tahmini, literatürdeki çalışmalarda daha çok uzaysal bölgede yapılmaktadır. Bu çalışmada ise frekans bölgesinde sözde faz hesaplaması ile hareket tahmini konusu incelenmiştir. Sözde faz hesaplaması, gerçekten faz bilgisi içermeyen frekans bölgesine (Ayrık Kosinüs Dönüşümü ve Ayrık Sinüs Dönüşümü) ait katsayılar üzerinden faz bilgisinin çıkartılmasıdır. Sözde faz ile hareket tahmini yöntemi temiz ve sentetik görüntülerde başarılı bir şekilde çalışmaktadır. Bu çalışmada bu yöntemin başarısız olduğu durumlar analiz edilmiş ve bu durumlarda da başarılı bir şekilde çalışması için çeşitli iyileştirmeler önerilmiştir. Yapılan deneysel incelemelerde, sözde faz hesaplamasının, görüntü üzerindeki gürültülerden olumsuz etkilendiği görülmüştür. Bu durumu ortadan kaldırmak için, medyan süzgeci ile gürültü temizlendikten sonra sözde faz hesaplaması yapılması önerilmekte ve bu şekilde daha başarılı sonuçlar elde edilmektedir. Ayrıca hareket eden nesnenin arka plana göre ayırt edilebilirliğinin düşük olması durumunda da yöntemin başarısının olumsuz etkilendiği görülmüştür. Görüntünün kontrastının (üssel dönüşüm ile) iyileştirildiğinde sözde faz hesaplamasının daha başarılı sonuçlar verdiği gözlenmiştir. Bunun dışında uyarlanabilir bir eşikleme yöntemi (Otsu yöntemi) uygulandıktan sonra da, başarılı sonuçlar elde edildiği görülmüştür. Sonuç olarak sözde faz kullanarak önerilen yöntemle zor şartlarda bile başarılı bir şekilde çalışan dayanıklı bir hareket tahmini yapılmıştır.

Anahtar Kelimeler: Hareket tahmini, sözde faz, ayrık kosinüs dönüşümü, ayrık sinüs dönüşümü.

A ROBUST MOTION ESTIMATION BY USING PSEUDO PHASE

ABSTRACT

Motion estimation is usually performed in the spatial domain at the studies in the literature. In this study, motion estimation in frequency domain by using pseudo phase calculation is investigated. Pseudo phase calculation is defined as the extraction of the phase information by using coefficients belonging to frequency domain (Discrete Cosine Transform and Discrete Sine Transform) which don't really include any phase information. The method of motion estimation by using pseudo phase is successful in clean and synthetic images. In this study, the unsuccessful conditions are analyzed and some enhancements are proposed in order to have success even in these conditions. In experimental investigations, it is seen that the calculation of the pseudo phase is negatively affected by the noise on images. To avoid this drawback, it is proposed that the pseudo phase is calculated after the noise reduction by a median filter and hence more successful results are obtained. When the gray level difference between the object and its background is low, then the result of the method is also wrongly affected. After contrast enhancement (e.g., exponential transformation), it is seen that the success of the pseudo phase calculation is increased. Furthermore, after adaptive thresholding (e.g. Otsu's method) successful results are again obtained. As a result, robust motion estimation is successfully applied by pseudo phase calculation even under severe conditions.

Keywords: Motion estimation, pseudo phase, discrete cosine transform, discrete sine transform.

1. GİRİŞ (INTRODUCTION)

Görüntü işleme uygulamaları içerisinde hareket tah-

mini üzerine yapılan çalışmalar daha çok uzaysal bölge verileri üzerinde yoğunlaşmaktadır. Uzaysal bölgede çalışan ve yaygın kullanılan hareket tahmini

yöntemlerinden biri Blok Eşleştirme Algoritmasıdır (BEA). BEA'nın farklı uygulama yaklaşımları bulunmaktadır. Bunlardan ikisi tüm arama ve logaritmik arama yaklaşımlarıdır. BEA'da, hareket unsuru içeren iki görüntü arasında bloklar tek tek karşılaştırılarak, benzer blok bulunmaya çalışılmaktadır. Yöntemleri farklılaştıran ise blokları tarama algoritması ve eşleştirme değerinin hesaplanmasıdır. İki görüntü arasındaki benzer blokların bulunması için belirlenen blokların diğer görüntü üzerindeki muhtemel tüm bloklar ile karşılaştırılması beklenir, ancak bu çok sayıda işlem yapılmasını gerektirmektedir. Bu yüzden farklı yöntemler ile daha az sayıda işlem yapılarak en uygun bloğun tespit edilmesi amaçlanmaktadır.

Roma ve Sousa (2005), doğrudan Ayrık Kosinüs Dönüşümü (AKD) katsayılarını kullanarak frekans bölgesinde hareket tahmini yapmışlardır [1]. Burada MPEG-x ve H.26x kodlamalarında hareket vektörlerinin belirlenmesinin daha yüksek performansla yapılabildiğini göstermişlerdir.

Bozinovic ve Konrad (2005), hareket analizi üzerine yaptıkları çalışmada, hareketin 3B AKD bölgesindeki özelliklerini incelemişlerdir [2]. Sonuç olarak sabit hızda, kendi etrafında dönen hareketin (translational), 3B AKD bölgesinde katlanmış bir düzlem üzerinde enerjisinin yoğunlaştığını görmüşlerdir. Bu sonuçtan yola çıkarak bir hareket tahmini yöntemi önermişler ve bu yöntemin blok eşleştirme yöntemlerine göre daha iyi performansla sonuç verdiğini göstermişlerdir. Aynı zamanda bu hareket analizini, AKD nicemeleme ve katsayıların sıralanmasında da kullanmışlardır.

Yip ve Rao (1987), çalışmalarında normal görüntü dizileri ve onların ötelenmiş biçimlerinin AKD ve Ayrık Sinüs Dönüşümü (ASD) bölgesindeki durumlarını karşılaştırarak frekans bölgesinde faz hesaplamasını incelemişlerdir [3].

Başka bir çalışmada, gürültülü görüntü dizilerinde düzlemsel hareketi tahmin eden bir yöntem önerilmiştir [4]. Burada bir grup çerçevenin ortalaması karşılaştırılarak Fourier faz bilgisi kullanılmıştır. Önerilen bu yöntemin Gauss tipi gürültülerde kalite ve işlem hızı açısından diğer yöntemlere göre daha iyi çalıştığı bildirilmiştir.

Hem hareket tahmininin başarısı, hem de JPEG, MPEG gibi sıkıştırma uygulamalarında performansın artırılması için frekans bölgesinde hareket tahmini yapılması önem kazanmaktadır. Hareket, görüntü sinyali üzerindeki öteleme anlamına gelmektedir. Bu öteleme sonucunda hareket içeren iki görüntüyü oluşturan sinyaller arasında faz farkı oluşmaktadır. İki sinyal arasındaki faz farkının hesaplanması ile hareket bulunabilmektedir. Faz farkı hesaplanması, Fourier dönüşümü üzerinden yapılabilmektedir. Fourier dönüşümü katsayıları karmaşık sayı olduklarından dolayı faz bilgisini taşımaktadır. Ancak AKD ve ASD

katsayıları ise gerçel sayı olduklarından dolayı tek başlarına faz bilgisi taşımamaktadırlar. Ancak AKD ve ASD katsayıları kullanılarak sözde faz hesaplaması denilen bir yöntem ile faz bilgisi çıkarılabilmektedir [5]. Koc ve Liu (1998), önerdikleri sözde faz hesaplaması yönteminin üzerinde farklı çalışmalarda bulunmuşlardır [5,6,7]. Yaptıkları çalışmalarda, AKD bölgesinde sözde faz hesaplaması ile alt piksel hareketlerinin tanınması için önerdikleri yöntemin, pikseller üzerinde aradeğerleme yapmaksızın, blok tanıma yöntemlerine göre daha yüksek performansla çalıştığını göstermişlerdir [6,7]. Aynı zamanda AKD bölgesinde hareket tahmininin yanı sıra, tüm AKD tabanlı kodlayıcılarda kullanılabilmesi amacıyla bir hareket dengeleme (motion compensation) yöntemi önermişlerdir [8].

İşlem karmaşıklığı açısından AKD tabanlı sözde faz hesaplamasına dayalı hareket tahmini yöntemi, blok eşleştirme algoritmasına göre daha hızlı çalışmaktadır. Blok eşleştirme algoritmasının karmaşıklığı $M \times M$ lik bir arama alanı ve $N \times N$ lik bir blok için asimptotik olarak $O(N^2 \times M^2)$ 'dir. AKD tabanlı hareket tahmini yönteminin ise işlem karmaşıklığı asimptotik olarak $O(N^2)$ 'dir ve sadece blok boyutuna bağlıdır. Ayrıca AKD tabanlı algoritma sıkıştırma ve hareket tahmini işlemlerini birleştirdiği için video kodlama karmaşıklığını da azaltmaktadır [9].

Koc ve Liu'nun önerdiği AKD bölgesinde sözde faz hesaplamasına dayalı hareket tahmini yönteminin iki boyutlu görüntüler için donanımsal bir uygulaması da gerçekleştirilmiştir [9]. Burada kullanılan paralel yapı sayesinde AKD tabanlı sistem, blok eşleştirme algoritmasına göre daha az işlem karmaşıklığına sahiptir ve belirli sürede daha fazla çıkış üretebilmektedir.

İşlem karmaşıklığını azaltmak için AKD bölgesinde sıkıştırılmış video görüntülerinin kod çevirimi (transcoding) ile ilgili çeşitli çalışmalar yapılmıştır. Öte yandan birçok video sıkıştırma standardı uzaysal bölgede hareket dengeleme önermektedir. Hareket dengelemenin uzaysal bölgede ve veri sıkıştırmanın ise dönüşüm bölgesinde yapılması çeşitli sorunların ortaya çıkmasına neden olmaktadır [10]. Her iki işlemin de AKD bölgesinde yapılması daha avantajlı olacaktır.

Bu çalışmada, sözde faz bilgisi üzerinden hareket tahmini uygulanmış ve incelenmiştir. Sözde faz hesaplanmasını olumsuz etkileyen faktörler açık bir şekilde belirlenmiştir. Hareket tahmininin başarısız olduğu bu durumların giderilmesi için çeşitli iyileştirmeler önerilmiştir. Faz bilgisinin doğru hesaplanabilmesi için karşılaştırma yapılan görüntülerin içerisinde tek bir hareket unsuru bulunmalıdır. Bunun yanı sıra gürültü ve hareket eden nesnenin ayırt edilebilirliği de yöntemin başarısını etkilemektedir. Bu amaçla bu çalışmada, görüntüler üzerinde medyan süzgeci kullanılarak gürültü temizlenmiş, eşikleme ve

üsel gri seviyesi dönüşümü ile de dinamik aralık artırılarak ayırt edilebilirlik yükseltilmiştir. Bu şekilde iyileştirme yapılan görüntüler üzerinde sözde faz hesaplamasının daha başarılı sonuçlar verdiği görülmüştür.

2. HAREKET TAHMİNİ YÖNTEMLERİ (MOTION ESTIMATION METHODS)

2.1. Blok Eşleştirme Algoritması (Block Matching Algorithm)

Bu algoritmada, iki görüntü arasında piksel değerlerine göre, karşılaştırılan alanın aynı olup olmadığına bakılır. Arama yapılan alanın seçilmesine göre matematiksel işlemi azaltmaya ve performansı yükseltmeye yönelik olarak farklı yöntemler geliştirilmiştir. Arama yüzeyinin taramasında kullanılan farklı yaklaşımlar bu yöntemi çeşitlendirmiştir.

Tüm arama algoritması (Full Search Algorithm): Uzaysal bölgede, hareket tahmini yapan bu algoritmada, referans görüntüdeki $N \times N$ 'lik bir pencere içerisindeki alan, arama yapılan görüntü üzerinde $N \times N$ 'lik alanlarla, piksel değerlerine göre karşılaştırılarak aynı görüntü bloğu aranır (Şekil 1). Arama penceresi, arama yapılan görüntü üzerinde gezdirilir. Bu yüzden çok sayıda matematiksel işlem gerektirir. Bunun yanı sıra nesnenin kameraya göre yatay veya dikey düzlemde hareket etmesi durumunda iyi sonuçlar elde edilir. Nesnenin merkezi etrafında dönmesi durumunda veya kameraya yaklaşması durumunda, arama penceresine göre açısı değişmiş veya büyümüş olacağından, referans görüntü üzerindeki alan ile aynı sayıda piksel değerlerine sahip olmaz, bu da algoritmayı yarıltır. Bu yüzden arama uzayını, açıya bağlı olarak ve z eksenini boyunca da genişletmek gerekir. Bu da karşılaştırma sırasındaki matematiksel işlemleri daha da karmaşık hale getirmektedir. Dezavantajlarına rağmen Blok Eşleştirme Algoritması, en temel mantığa sahip ve yaygın kullanılan bir yöntemdir.

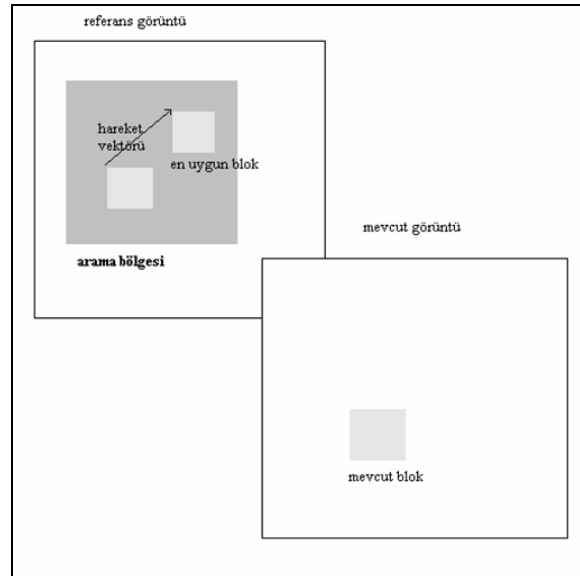
Logaritmik arama: Bu arama yönteminde, $N \times N$ 'lik blok $M \times M$ 'lik bir arama alanı içerisinde arama yaparken, ilk adımda arama alanının çevresi üzerinde 8 noktada arama yapılır (Şekil 2). Bu 8 noktanın merkezdeki referans görüntü bloğuna uzaklığı $d_1 = 2^{k-1}$, $k = \log_2(M+1)$ dir. İkinci adımda ise, birinci adımdaki en uygun blok, merkezde olacak şekilde aynı yöntem uygulanır, ancak arama blokları arasındaki uzaklık $d_2 = d_1/2$ dir [11].

Elmas arama: Logaritmik aramanın yanı sıra literatürde farklı yöntemler de bulunmaktadır. Zhu ve Ma, Elmas arama (Diamond Search) yöntemini önererek, bu yöntemin 3-adım arama, Yeni 3-adım arama ve 4-adım arama yöntemlerine göre daha az işlem gerektirerek yaklaşık performans sağladığını göstermişlerdir [12].

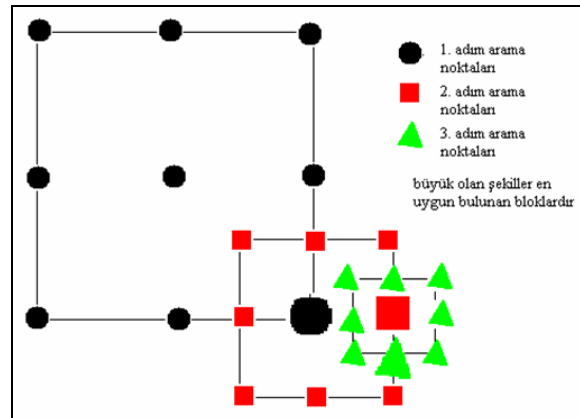
Diğer yaklaşımlar: Blok eşleştirme uygulayan

yöntemlerin genel bir problemi, mümkün olan en az sayıda karşılaştırma yaparken, arama alanı içerisindeki genel en benzer noktayı bulmayı garanti etmiyor olmalarıdır. Chen ve ark. (2001) bunun için "winner-update" yöntemini önermişlerdir [13]. Bu yöntemde referans bloğa göre hata oranı en az çıkan bölgelerde arama yaygınlaştırılmakta, bu sayede en çok benzer bloğun bulunması ihtimali yükselirken karşılaştırma sayısı azaltılmaktadır. Liu ve Feiq (1996), "gradient descent search" yöntemlerinde [14] arama yapılan görüntünün merkezinde bir alan seçerek öncelikle bu alan içerisinde arama yapmaktadır. Bu alan içerisinde aranan blokların hata değerlerine bakılarak, hangi yöne doğru azalma olduğu saptanmaktadır. Aynı şekilde hatadaki değişime bakılarak sonraki karşılaştırma bloğu tespit edilir.

Blok Eşleştirme Algoritmasında, arama alanlarının karşılaştırması için farklı matematiksel yöntemler kullanılmaktadır. En basit yöntem piksel farklarının toplamıdır. Arama alanındaki her bir piksel değeri, karşılaştırılan alandaki aynı pozisyondaki piksel değerinden çıkartılarak, tüm farkların toplamı hesap-



Şekil 1. Blok Eşleştirme Algoritmasında tüm arama uzayı (Full search space for Block Matching Algorithm)



Şekil 2. Logaritmik arama noktaları (Logarithmic search points)

lanmaktadır (Eş. 1). Sonucu 0'a en yakın olan nokta, en çok benzerlik taşımaktadır.

$$SAD = \sum_{x=1}^N \sum_{y=1}^N |f_{mevcut}(x, y) - f_{referans}(x, y)| \quad (1)$$

Burada, SAD (Sum of Absolute Differences) piksel farklarının toplamıdır, N blok boyutudur.

2.2. Fourier Bölgesinde Faz Hesaplama (Phase Calculation in Fourier Domain)

Uzaysal bölgedeki bir hareket, Fourier dönüşümünün karmaşık katsayılarında bir faz üretmektedir. Bu faz bilgisi çıkartılarak hareket tahmini yapılabilir. Eş. 2'de Fourier dönüşümü ve faz arasındaki ilişki görülebilmektedir.

$$\mathfrak{T}(x(t-T)) = \mathfrak{T}(x(t))e^{-j\omega T} \quad (2)$$

Burada T öteleme miktarına karşılık gelmektedir. Harekete karşılık gelen faz, Fourier dönüşüm katsayılarından Eş. 3'te gösterildiği şekilde hesaplanabilmektedir [15].

$$faz = \mathfrak{T}^{-1}(\mathfrak{T}(t)conj(\mathfrak{T}(t-T))) \quad (3)$$

Burada conj, karmaşık eşleniğe karşılık gelmektedir. Eş. 3 sonucunda ortaya çıkan sinyalin tepisi noktasına bağlı olarak öteleme miktarı belirlenmektedir.

Reddy ve Chatterji (1996), yaptıkları çalışmalarında Fourier hesaplamasının rotasyonel, çevrilme (translational) ve ölçeklendirme (scaling) özelliklerinden bahsetmektedirler. Buna dayalı olarak bir korelasyon hesaplaması önermişlerdir [16].

Frekans bölgesinde Fourier katsayılarının Eş. 3'teki gibi çarpımı, işlemsel olarak uzaysal bölgede iki sinyal arasındaki korelasyona karşılık gelmektedir. Korelasyon benzerliğin bulunmasında kullanılan bir yöntemdir. Korelasyon işlemi, öteleme uygulanmış iki görüntü üzerinde uygulanarak, öteleme miktarı benzerliğe bağlı olarak hesaplanabilmektedir.

3. SÖZDE FAZ HESAPLAMASI (PSEUDO PHASE CALCULATION)

Sözde faz, bir sinyal ve ötelenmiş halinin AKD ve ASD katsayıları kullanılarak hesaplanmaktadır. AKD ve ASD katsayıları tek başlarına faz bilgisi taşımamaktadır. Ancak her iki sinyal üzerinden yola çıkılarak faz bilgisine erişilebilmektedir. Koc ve Liu, ilk olarak 1994 yılında yayınladıkları bir çalışmada [17], Ayrık Kosinüs ve Sinüs Dönüşümü kullanarak hareket tahmini yöntemlerini açıklamışlardır. 1998 yılında yayınladıkları çalışmada, yöntemlerini genişleterek yeni bir makaleyle anlatmışlardır [5].

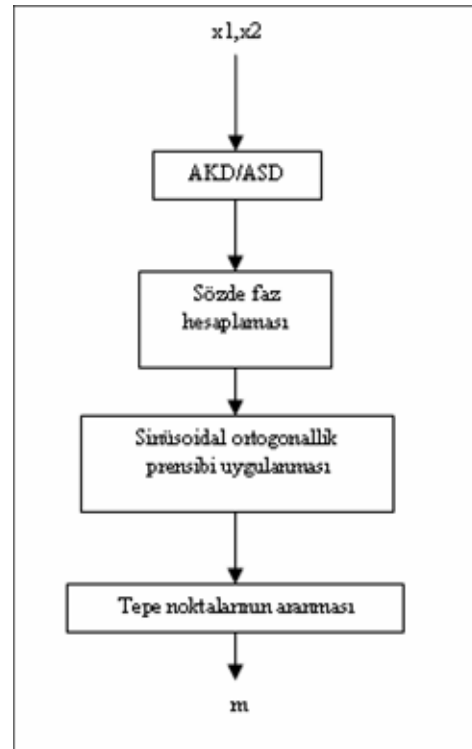
3.1. 1B Sinyaller İçin Sözde Faz Hesaplaması (Pseudo Phase Calculation for 1-D Signals)

AKD/ASD tabanlı hareket tahmini yöntemi, AXD-HT (AKD/ASD tabanlı Hareket Tahmini), öncelikle 1B sinyaller üzerine uygulanmıştır. Daha sonra 1 boyutlu üzerinde yapılan çalışma 2 boyutlu düzleme aktarılmıştır. Şekil 3'te 1 boyutlu sinyaller için sözde faz ile hareket tahminine ait akış şeması görülmektedir. x_1 ; orijinal sinyal, x_2 ise x_1 'in m kadar ötelenmiş halidir. Sözde faz hesaplaması Eş. 4'teki gibidir.

$$g_m^s(k) = \begin{cases} \frac{Z_1^C(k)X_2^S(k) - Z_1^S(k)X_2^C(k)}{[Z_1^C(k)]^2 + [Z_1^S(k)]^2}, & k \neq N \\ 1, & k = N \end{cases} \quad (4)$$

g_m^s ; sözde faz fonksiyonu, Z_1^C ; x_1 'in 1. tip AKD'si, Z_1^S ; x_1 'in 1. tip ASD'si, X_2^C ; x_2 'nin AKD'si, X_2^S ; x_2 'nin ASD'sidir.

Sözde faz fonksiyonu $\sin[(k\pi/N)(m+0,5)]$ 'e eşittir. Sonuç olarak sözde faz fonksiyonunun ters ASD'si hesaplanarak, gürültüsüz sinyaller için tek bir frekansta tanımlı sinüs fonksiyonunun ters dönüşümü olarak bir tepisi fonksiyonu elde edilir. Gürültülü sinyaller üzerinde ise aynı tepisi fonksiyonu gürültülü olarak elde edilir. Koc ve Liu çalışmalarında orijinal ve ötelenmiş sinyaller arasında PSNR=10dB gürültüde dahi ayırt edilebilir sonuç elde edildiğini göstermişlerdir [5]. Ters dönüşüm sonucu elde edilen fonksiyonda tepiyi oluşturan noktanın indisi öteleme miktarını vermektedir.



Şekil 3. Sözde faz ile hareket tahmini akış şeması (Block diagram of motion estimation by pseudo phase)

Eş. 5'e göre, $d(i)$ (tepinin değeri), pozitif yönde ise öteleme miktarı $+x$ yönünde i kadar, negatif yönde ise öteleme miktarı $-x$ yönünde $(i+1)$ kadardır.

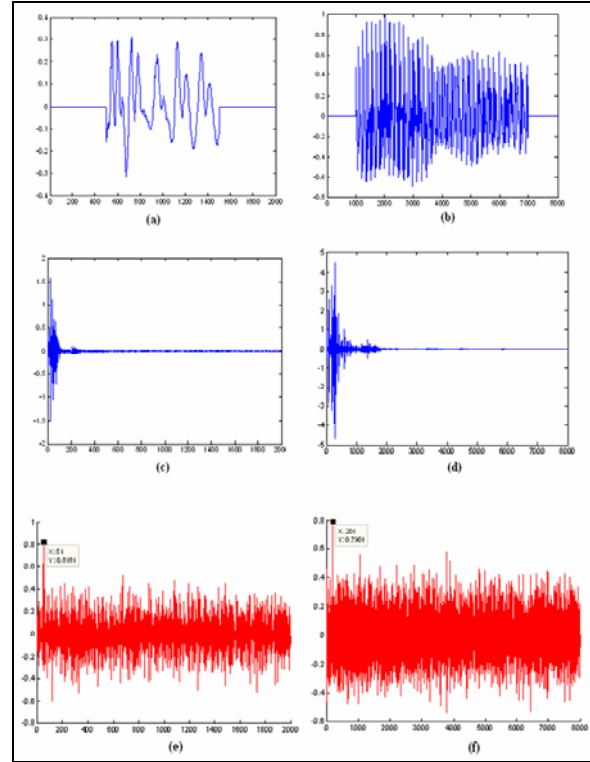
$$m = \begin{cases} i & , d(i) > 0 \\ -(i+1) & , d(i) < 0 \end{cases} \quad (5)$$

Simülasyonda sözde faz yöntemi uygulanarak, doğrulaması ve geliştirilmesi yapılmıştır. Gerçek bir ses dosyasından alınan örnek sinyaller ötelenerek ve üzerine gürültü eklenerek, AXD-HT yöntemi ile sinyaldeki öteleme bulunmaya çalışılmıştır. Orijinal sinyalin üzerine eklenen gürültü Eş. 6'daki formülle hesaplanmıştır.

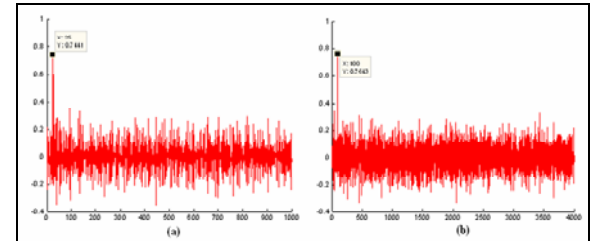
$$PSNR_{hata} = 10 \log_{10} \left(\frac{\sum (orijinal.*orijinal)}{\sum (gürültü.*gürültü)} \right) \quad (6)$$

Şekil 4'te (a) ve (b) grafikleri sırasıyla x_1 ve x_2 sinyallerini göstermektedirler. (c) ve (d) görüntüleri bunların AKD'lerini göstermektedir. Bu grafiklerden x_2 sinyalinin yüksek frekanslarda x_1 sinyaline göre daha fazla bileşen içerdiği görülmektedir. x_1 sinyaline $+50$ öteleme uygulanmış ve eklenen gürültü sonucunda $13,8513$ dB hata hesaplanmıştır. x_2 sinyaline ise $+200$ öteleme uygulanmış ve eklenen gürültü sonucunda $11,4971$ dB hata hesaplanmıştır. (e) grafiğinde x_1 ve ötelenmiş sinyaline AXD-HT uygulanmıştır. 50 indisinde en yüksek genliğe sahip tepiyi göstermektedir ve gürültüden ayırt edilebilmektedir. (f) grafiğinde x_2 ve ötelenmiş sinyaline AXD-HT uygulanmış ve 200 indisinde ortaya çıkan tepinin genliği en yüksek tepiyi olduğu görülmüştür. x_2 sinyaline x_1 sinyalinden daha fazla gürültü eklenmiş olmasına rağmen AXD-HT, x_2 sinyali üzerinde x_1 sinyaliyle benzer ayırdedilebilirlikte başarılı bir sonuç göstermektedir. x_2 sinyalinin örnekleme sayısı daha fazla, ve daha yüksek frekansta bileşenler içermekte, yani sinyal daha fazla farklı bileşen içermektedir. AXD-HT yöntemi, iki sinyal arasındaki benzerliği frekans alanında incelemektedir. Benzerliklerin belirlenebileceği daha çok bilgi içermesinden dolayı x_2 sinyali üzerinde daha başarılı sonuç vermektedir.

AXD-HT yönteminde, farklı bir yaklaşım olarak sözde faz fonksiyonu hesaplamasında kullanılan x_1 ve x_2 sinyallerinin AKD ve ASD katsayılarının, ilk yarısı hesaplama işlemine katılarak sonuçları araştırılmıştır. Frekans bölgesinde çıkan katsayıların ilk yarısı alınarak, uzaysal bölgede sinyalin örnekleme sayısı yarıya düşürülmüş, genliği artırılmış olur. x_1 sinyali, $+50$ ötelenmiş ve üzerine eklenen gürültü ile PSNR hata oranı $13,7478$ olan sinyal ile bu yöntem uygulanmıştır. Şekil 5 (a) grafiğinde sonucu görülmektedir. x_2 sinyali ise $+200$ ötelenmiş ve üzerine eklenen gürültü ile PSNR hata oranı $11,4867$ olmuştur. Sonuç grafiği Şekil 5 (b) grafiğinde görülmektedir. Sinyallerin uzunluğundaki azalmadan



Şekil 4. Yüksek frekantaki bileşenlerin AXD-HT'ye etkisi (Impact of high frequency components on DXT-ME)

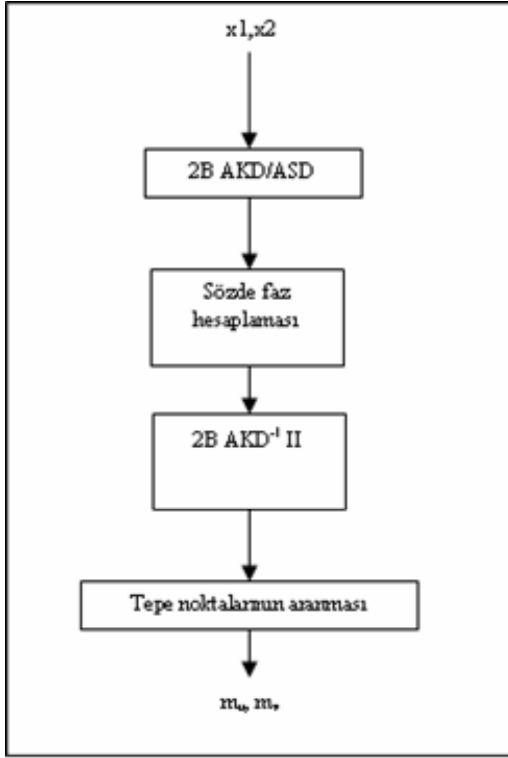


Şekil 5. AXD-HT sonucu (DXT-ME result)

dolayı öteleme miktarı yarıya indirilmiş olarak görülmektedir. x_1 için 25 . indiste, x_2 için 100 . indiste tepiyi oluştuğu görülmektedir. Bu da öteleme miktarındaki çözünürlüğü düşürmektedir. Örneğin $+200$ ve $+201$ öteleme miktarı için yöntem aynı sonuçları verecektir. Ancak tepiyi oluşan noktanın daha rahat okunabilir olduğu görülmektedir. Bu yüzden daha fazla gürültüde daha iyi sonuçlar elde etmek için kullanılabilir.

3.2. 2B Sinyaller İçin Sözde Faz Hesaplaması (Pseudo Phase Calculation for 2-D Signals)

Görüntü iki boyutlu sinyal olarak düşünüldüğünde, tek boyutlu sinyallerdeki öteleme, 2 boyutlu görüntüdeki hareket unsuruna karşılık gelmektedir. Ötelenmiş bir sinyalin Fourier dönüşümü içerisinde öteleme miktarı bilgisi faz bileşenine bağlı olarak taşınmaktadır. Ancak AKD veya ASD içerisinde faz bileşeni yoktur. AKD ve ASD katsayılarından sözde faz bilgisi çıkartılmakta ve bu sayede öteleme miktarı hesaplanabilmektedir. Sözde faz fonksiyonunu elde etmek için, sinüzoidal ortogonal prensibi kullanılmaktadır.



Şekil 6. İki boyutlu sinyallerde sözde faz hesaplanması (Block diagram of motion estimation on two dimensional signals)

Şekil 6'da iki boyutlu sinyaller üzerinde sözde faz hesaplaması için akış şeması görülmektedir. Buna göre x_1 ve onun ötelenmiş hali olan x_2 sinyali için AKD ve ASD katsayıları hesaplanır. x_2 sinyali için Eş.7-10'da gösterilen 2. tip AKD, AKSD (Ayrık Kosinüs/Sinüs Dönüşümü), ASKD (Ayrık Sinüs/Kosinüs Dönüşümü), ASD katsayıları hesaplanmaktadır.

$$X_t^{CC}(k, l) = \frac{4}{N^2} C(k)C(l) \sum_{m,n=0}^{N-1} x_t(m, n) \cos\left[k\frac{\pi}{N}(m+0,5)\right] \cos\left[l\frac{\pi}{N}(n+0,5)\right] \quad (7)$$

$$k, l \in \{0, \dots, N-1\}$$

$$X_t^{CS}(k, l) = \frac{4}{N^2} C(k)C(l) \sum_{m,n=0}^{N-1} x_t(m, n) \cos\left[k\frac{\pi}{N}(m+0,5)\right] \sin\left[l\frac{\pi}{N}(n+0,5)\right] \quad (8)$$

$$k \in \{0, \dots, N-1\}; l \in \{0, \dots, N-1\}$$

$$X_t^{SC}(k, l) = \frac{4}{N^2} C(k)C(l) \sum_{m,n=0}^{N-1} x_t(m, n) \sin\left[k\frac{\pi}{N}(m+0,5)\right] \cos\left[l\frac{\pi}{N}(n+0,5)\right] \quad (9)$$

$$k \in \{1, \dots, N\}; l \in \{0, \dots, N-1\}$$

$$X_t^{SS}(k, l) = \frac{4}{N^2} C(k)C(l) \sum_{m,n=0}^{N-1} x_t(m, n) \sin\left[k\frac{\pi}{N}(m+0,5)\right] \sin\left[l\frac{\pi}{N}(n+0,5)\right] \quad (10)$$

$$k \in \{1, \dots, N\}; l \in \{1, \dots, N\}$$

x_1 sinyali için ise, Eş. 11-14'de gösterilen 1. tip AKD, AKSD, ASKD ve ASD katsayıları hesaplanmaktadır.

$$Z_{t-1}^{CC}(k, l) = \frac{4}{N^2} C(k)C(l) \sum_{m,n=0}^{N-1} x_{t-1}(m, n) \cos\left[k\frac{\pi}{N}\right] \cos\left[l\frac{\pi}{N}\right] \quad (11)$$

$$k, l \in \{0, \dots, N\}$$

$$Z_{t-1}^{CS}(k, l) = \frac{4}{N^2} C(k)C(l) \sum_{m,n=0}^{N-1} x_{t-1}(m, n) \cos\left[k\frac{\pi}{N}\right] \sin\left[l\frac{\pi}{N}\right] \quad (12)$$

$$k \in \{0, \dots, N\}; l \in \{1, \dots, N-1\}$$

$$Z_{t-1}^{SC}(k, l) = \frac{4}{N^2} C(k)C(l) \sum_{m,n=0}^{N-1} x_{t-1}(m, n) \sin\left[k\frac{\pi}{N}\right] \cos\left[l\frac{\pi}{N}\right] \quad (13)$$

$$k \in \{1, \dots, N-1\}; l \in \{0, \dots, N\}$$

$$Z_{t-1}^{SS}(k, l) = \frac{4}{N^2} C(k)C(l) \sum_{m,n=0}^{N-1} x_{t-1}(m, n) \sin\left[k\frac{\pi}{N}\right] \sin\left[l\frac{\pi}{N}\right] \quad (14)$$

$$k, l \in \{1, \dots, N-1\}$$

x_1 ve x_2 'ye ait dönüşüm hesaplamaları kullanılarak Eş. 15 ve Eş. 16'daki sözde faz fonksiyonları hesaplanmaktadır.

$$f_{m_u, m_v}(k, l) = \begin{cases} \frac{\hat{g}_{m_u, m_v}^{CS}(k, l)}{Z_{t-1}^{CC}(k, l)X_t^{CS}(k, l) + Z_{t-1}^{CS}(k, l)X_t^{CC}(k, l)} & k, l \in \{1, \dots, N-1\} \\ \frac{Z_{t-1}^{CC}(k, l)^2 + Z_{t-1}^{CS}(k, l)^2}{Z_{t-1}^{CC}(k, l)X_t^{CS}(k, l) + Z_{t-1}^{CS}(k, l)X_t^{CC}(k, l)} & k=0, l \in \{1, \dots, N-1\} \\ \frac{Z_{t-1}^{CC}(k, l)^2 + Z_{t-1}^{SC}(k, l)^2}{Z_{t-1}^{CC}(k, l)X_t^{CS}(k, l) + Z_{t-1}^{SC}(k, l)X_t^{SS}(k, l)} & l=N, k \in \{1, \dots, N-1\} \\ \frac{X_t^{CS}(k, l)}{Z_{t-1}^{CC}(k, l)} & k=0, l=N \end{cases} \quad (15)$$

$$g_{m_u, m_v}(k, l) = \begin{cases} \frac{\hat{g}_{m_u, m_v}^{SC}(k, l)}{Z_{t-1}^{CC}(k, l)X_t^{SC}(k, l) - Z_{t-1}^{SC}(k, l)X_t^{CC}(k, l)} & k, l \in \{1, \dots, N-1\} \\ \frac{Z_{t-1}^{CC}(k, l)^2 + Z_{t-1}^{SC}(k, l)^2}{Z_{t-1}^{CC}(k, l)X_t^{SC}(k, l) - Z_{t-1}^{SC}(k, l)X_t^{CC}(k, l)} & l=0, k \in \{1, \dots, N-1\} \\ \frac{Z_{t-1}^{CC}(k, l)X_t^{SC}(k, l) + Z_{t-1}^{CS}(k, l)X_t^{SS}(k, l)}{Z_{t-1}^{CC}(k, l)^2 + Z_{t-1}^{CS}(k, l)^2} & k=N, l \in \{1, \dots, N-1\} \\ \frac{X_t^{SC}(k, l)}{Z_{t-1}^{CC}(k, l)} & k=N, l=0 \end{cases} \quad (16)$$

Eş. 15 ve Eş. 16'da gösterilen sözde faz fonksiyonları matrisin kenar satır ve sütunlarının hesaplanmasında

kullanılmaktadır. Buradaki \hat{g} ve \hat{f} fonksiyonları ise Eş. 17'de verilen eşitliğe göre hesaplanmaktadır [17].

$$\begin{bmatrix} Z_{t-1}^{CC} & -Z_{t-1}^{CS} & -Z_{t-1}^{SC} & Z_{t-1}^{SS} \\ Z_{t-1}^{CS} & Z_{t-1}^{CC} & -Z_{t-1}^{SS} & -Z_{t-1}^{SC} \\ Z_{t-1}^{SC} & -Z_{t-1}^{SS} & Z_{t-1}^{CC} & -Z_{t-1}^{CS} \\ Z_{t-1}^{SS} & Z_{t-1}^{SC} & Z_{t-1}^{CS} & Z_{t-1}^{CC} \end{bmatrix} \begin{bmatrix} \hat{g}^{CC}(k,l) \\ \hat{g}^{CS}(k,l) \\ \hat{g}^{SC}(k,l) \\ \hat{g}^{SS}(k,l) \end{bmatrix} = \begin{bmatrix} X_t^{CC} \\ X_t^{CS} \\ X_t^{SC} \\ X_t^{SS} \end{bmatrix} \quad (17)$$

Elde edilen \hat{f} ve \hat{g} fonksiyonlarının ters Ayrık Kosinüs/Sinüs Dönüşümü (AKSD) ve Ayrık Sinüs/Kosinüs Dönüşümü (ASKD) katsayıları Eş. 18 ve Eş. 19'daki gibi hesaplanarak elde edilen grafiğin tepsi noktasının işaret ve koordinat bilgisi Tablo 1'e göre yorumlanarak öteleme miktarı ve yönü belirlenmektedir. Buradaki görüntülerde orijin noktası sol üst köşe olduğu, x eksenini değerlerinin sağa doğru büyümekte ve y eksenini değerlerinin aşağı doğru büyümekte olduğu kabul edilmektedir.

$$AKSD^{-1} = \frac{4}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} C^2(k)C^2(l)f_{mu,mv}(k,l) \quad (18)$$

$$\cos(k\pi/N(m+0,5))\sin(l\pi/N(n+0,5))$$

$$ASKD^{-1} = \frac{4}{N^2} \sum_{k=1}^N \sum_{l=0}^{N-1} C^2(k)C^2(l)g_{mu,mv}(k,l) \quad (19)$$

$$\sin(k\pi/N(m+0,5))\cos(l\pi/N(n+0,5))$$

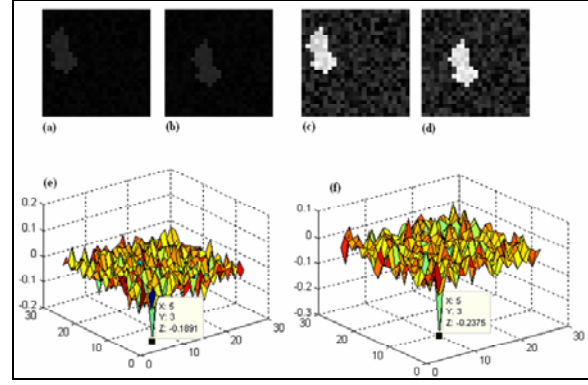
Tablo 1. Sözde faz ile hareket belirleme çizelgesi (Table for determining motion by pseudo phase)

ASKD ⁻¹ işareti	AKSD ⁻¹ işareti	Öteleme (x,y)
+	-	(m _u ,-(m _v -1))
-	-	(m _u ,m _v)
+	+	(-(m _u -1),-(m _v -1))
-	+	(-(m _u -1),m _v)

Anlatılan yöntem, başarısının doğrulanması amacıyla oluşturulan örnek veriler üzerinde uygulanmıştır. Arka plana ve nesneye gürültü eklenerek Eş. 20'da belirtilen hesaplama ile hata oranı hesaplanarak belirli bir hata oranı altında da sözde faz hesaplaması ile hareket tahmini yapıp yapılamayacağı incelenmiştir.

$$MSE = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N (x_2(i,j) - x_1(i,j))^2 \quad (20)$$

Arka plandaki hata değeri 17,1488, nesne üzerindeki hata değeri 12,9 olarak x₂ görüntüsünde x₁ görüntüsüne göre, x ekseninde +5 ve y ekseninde +3 birim öteleme uygulanarak, x₁ ve x₂ görüntüleri üzerinde sözde faz hesaplaması uygulanarak öteleme hesaplanmıştır. Şekil 7'de sonuçları gösterilmektedir; başarılı bir şekilde x ekseninde +5 ve y ekseninde +3



Şekil 7. AXD-HT sonucu (DXT-ME result)

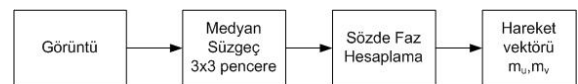
birim öteleme miktarı hesaplanabilmiştir.

Sözde faz hesaplaması ve diğer faz hesaplama yöntemleri frekans bölgesinde iki sinyal arasındaki faz farkını ortaya çıkarmaktadırlar. Bu yüzden görüntü içerisinde iki farklı hareket bulunuyorsa, tek bir sinyalin ötelenmesinden farklı bir durum bulunmaktadır. Dolayısıyla faz farkından bahsedilememektedir. Sözde faz hesaplaması tek bir hareket bulunan görüntü üzerinde uygulanabilmektedir. Bununla birlikte bahsedilen gürültü ve nesnenin ayırt edilebilirliğinde yöntemin başarısız sonuç vermesinden dolayı gerçek video görüntüleri üzerinde verimli olamamaktadır. Ancak sentetik görüntülerde ve belirlenen şartların sağlanabildiği koşullar altında veya burada önerilen iyileştirmeler yapılarak sözde faz hesaplaması başarılı sonuçlar vermektedir.

4. ÖNERİLEN YÖNTEM (PROPOSED METHOD)

Koc ve Liu'nun önerdikleri yöntem sıfır gri seviyeli bir arkaplana sahip yüksek gri seviyesi değerli piksellerin hareketini yakalamada başarılı olmaktadır. Ancak gerçek görüntülerde bu özel durumun dışında pek çok görüntü çerçevesi bulunmaktadır. Yapılan incelemelerde, sözde faz hesaplamasının görüntüler üzerindeki gürültüden olumsuz olarak etkilediği görülmüştür. İki görüntü arasındaki yüksek miktardaki gürültü oranı, sonuç grafiklerinde de gürültüye neden olmaktadır. Böylelikle tepsi noktası doğru bir şekilde belirlenememektedir. Gürültüyü temizleyerek doğru sonuca yaklaşabilmek için görüntüler üzerinde medyan süzgeci kullanılması önerilmiştir (Şekil 8).

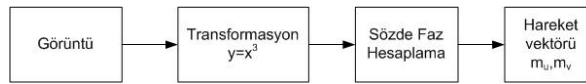
Medyan süzgeci, işlem yaptığı pencere içerisinde ortanca değeri hesaplayarak gürültüyü azaltırken, nesnelerin kenar çizgilerinin yumuşamasını engellemektedir. Kenar çizgilerin belirgin olması, hareket tahmininde önemli bir veri oluşturmaktadır. Bu amaçla medyan süzgeçleme başarılı sonuçlar vermiştir.



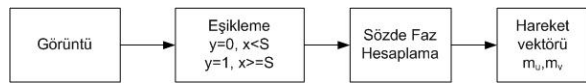
Şekil 8. Medyan süzgeci ile iyileştirme (Improvement by median filter)

Sözde faz hesaplamasının başarısını olumsuz etkileyen diğer bir durum ise nesne ile arka plan arasındaki ayırt edilebilirliğin düşük olmasıdır. Bu durumda görüntünün frekans bölgesinde yüksek frekans bileşenleri bulunmadığından dolayı, sözde faz hesaplamasının faz farkını hesaplayabilmesi için yeterince bilgi bulunamamaktadır. Dolayısıyla hareket algılanamadığından sonuç grafikleri üzerinde (0,0) noktasında tepik görülmektedir ve x , y eksenleri boyunca yanılıcı tepik noktaları görülmektedir. Bu duruma karşı olarak, görüntünün uzaysal bölgedeki gri seviyelerine üssel dönüşüm uygulanarak nesne ile arka plan arasındaki ayırt edilebilirliğin artırılması önerilmiştir (Şekil 9).

Üçüncü bir yöntem olarak da, eşikleme kullanılarak görüntülerin ikili hale dönüştürülmesi önerilmiştir (Şekil 10).



Şekil 9. Gri seviyesi dönüşümü ile iyileştirme (Improvement by gray-level transformation)



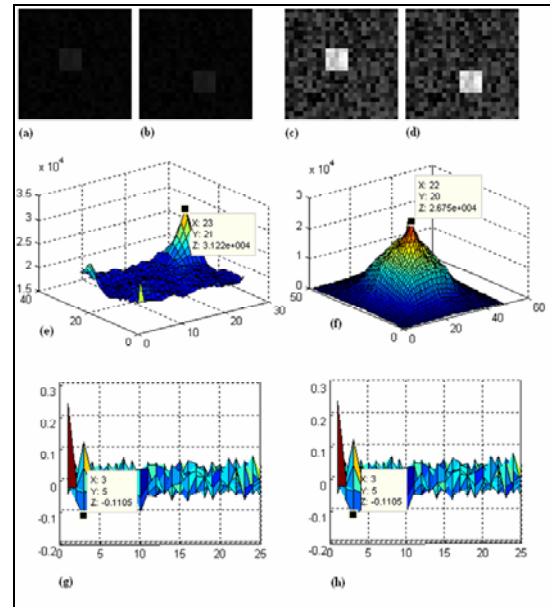
Şekil 10. Eşikleme ile iyileştirme (Improvement by thresholding)

Bu sayede nesne ve arka plan kesin olarak birbirlerinden ayrılarak, hareket tahmini kesin sonuçlar ile belirlenebilmektedir. Ancak arka plan ve nesne arasındaki gri seviyeleri farkı yüksek olmadığından dolayı eşik noktası sabit belirlenmemektedir. Bundan dolayı Otsu'nun uyarlanı eşikleme yöntemi uygulanmaktadır.

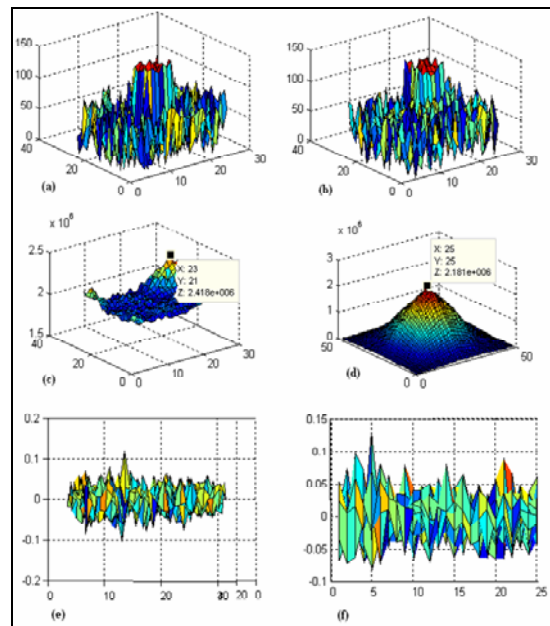
5. DENEYSEL ÇALIŞMALAR (EXPERIMENTAL RESULTS)

Şekil 11'de ayırdedilebilirliği düşük bir nesnenin görüntüsü üzerinde Fourier faz hesaplama ve sözde faz hesaplama uygulanmıştır. (a) x_1 görüntüsü, arka plan 0-10 arası, nesne 20-30 arası rastgele değerlerden üretilmiştir, (b) x_2 görüntüsü, x_1 görüntüsündeki nesnenin ötelenmiş hali, (c) x_1 'in histogram eşitleme yapılmış hali, (d) x_2 'nin histogram eşitleme yapılmış hali, (e) Fourier faz hesaplama yöntemi sonucu, (f) korelasyon sonucu, (g) sözde faz hesaplama yöntemi AKSD⁻¹ sonucu, (h) sözde faz hesaplama yöntemi ASKD⁻¹ sonucunu göstermektedir. Fourier faz hesaplaması başarılı sonuç verirken, sözde faz için sonuç grafiklerine bakıldığında (0,0) noktasında en yüksek tepik görülmüştür. Ancak $x=0$ ve $y=0$ eksenleri haricinde, iç bölgede en yüksek tepinin bulunduğu nokta, hareket vektörünü başarıyla vermektedir. Sonuç olarak nesne ile arkaplan arasındaki ayırdedilebilirliğin düşük olduğu durumda yöntemin yanılıcı sonuçlar verebildiği görülmüştür.

Şekil 12'de üzerinde hareket tahmini yapılacak görüntünün arka planına ve nesneye gürültü eklenmiş, ve nesne ötelenmiştir. Sonuçta ortaya çıkan x_2 görüntüsü ile x_1 görüntüsünün arka planları arasında MSE ile 1610,1 hata hesaplanmıştır. Nesne üzerinde ise MSE ile 132,1837 hata hesaplanmıştır. (c) grafiği Fourier faz hesaplama yöntemi sonucudur ve bu grafiğe göre başarıyla öteleme miktarı bulunmaktadır. (e) ve (f) grafikleri ise sözde faz hesaplama yöntemi sonuçlarıdır ve görüntü üzerindeki yüksek gürültü, sonuçları üzerinde de gürültü oluşturarak doğru sonucun bulunmasını engellemektedir.



Şekil 11. Fourier faz hesaplama, korelasyon ve AXD-ME sonuçları (Fourier phase calculation, correlation and DXTE results)



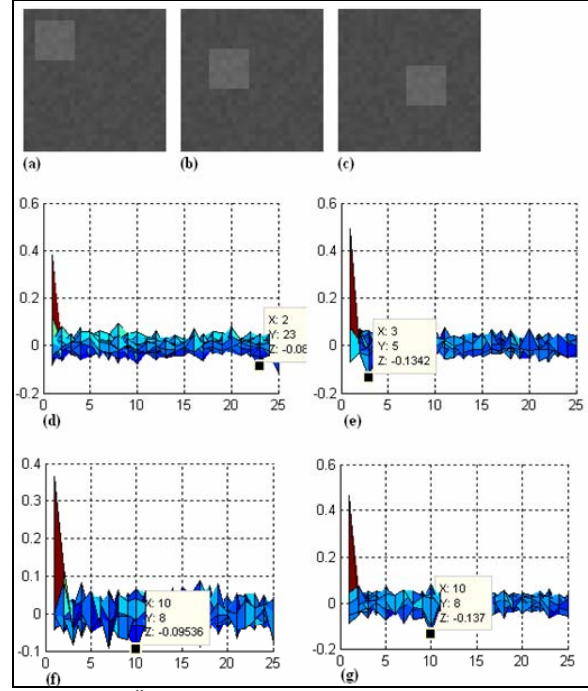
Şekil 12. Fourier faz hesaplama, korelasyon ve AXD-ME sonuçları (Fourier phase calculation, correlation and DXTE results)

Şekil 13 (a)'da gösterilen x_1 görüntüsünün nesne ile arka planı arasındaki ayırt edilebilirlik düşüktür. (b)'deki x_2 görüntüsü, x_1 'in +3,+5 ötelenmiş halidir. Bu iki görüntü üzerinde sözde faz hesaplaması yapıldığında (d) ve (e) grafiklerinde görülen sonuçlar elde edilmiştir. Buna göre başarılı sonuç elde edilemediği görülmektedir. (c)'deki x_3 görüntüsü ise x_1 'in +10,+8 ötelenmiş halidir. x_1 ve x_3 üzerinde sözde faz hesaplaması yapıldığında (f) ve (g) grafiklerindeki sonuçlar elde edilmiştir ve başarılı olduğu görülmüştür. Buna göre, sözde faz hesaplamasının başarısı görüntü üzerindeki öteleme miktarına bağlı olduğu görülmektedir. Sözde faz hesaplaması görüntünün tamamına ait AKD ve ASD katsayıları ile yapılmaktadır. Bu yüzden gürültülü görüntüler ve nesnenin ayırt edilebilirliğinin düşük olduğu görüntülerde, sonuç üzerinde ortaya çıkan gürültü küçük değişimlerin hesaplanabilmesine engel olmaktadır. Bu problemi çözmek için bu çalışmada eşikleme yönteminin kullanılması önerilmiştir.

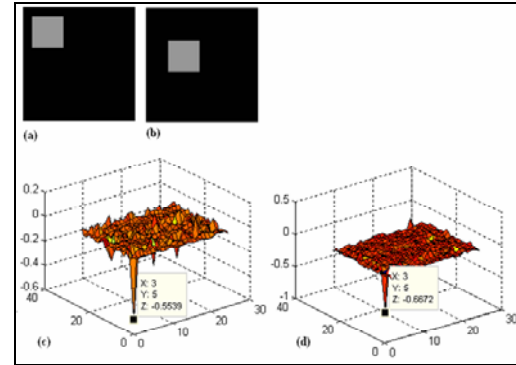
Şekil 13'teki x_1 ve x_2 görüntülerine Otsu'nun eşikleme yöntemi ile eşikleme uygulanarak nesne ve arka plan arasındaki ayırt edilebilirlik artırılmış ve Şekil 14'teki (a) ve (b) görüntüleri elde edilmiştir. Bu görüntüler üzerinde sözde faz hesaplaması yapıldığında, Şekil 14 (c) ve (d) grafiklerinde görülen $AKSD^{-1}$ ve $ASKD^{-1}$ sonuçları elde edilmiştir. Böylelikle nesnenin ayırt edilebilirliği düşük olan görüntülerde eşikleme uygulanarak sözde faz hesaplamasının başarısı artırılmıştır.

Şekil 15'de (a) ve (b) görüntülerinde x_1 ve x_2 'nin grafiklerinde nesnenin ayırt edilebilirliğinin arka plana göre düşük olduğu görülmektedir. (c) şeklinde x_1 ve x_2 için gri seviyesi aralığı görülmektedir. Buna göre tüm görüntü 25-40 arasında değerlerden oluşmaktadır. x_1 ve x_2 üzerinde sözde faz hesaplaması yapıldığında (d) ve (e) grafiklerindeki sonuçlar elde edilmektedir. $AKDS^{-1}$ sonucu başarılı olmasına rağmen $ASKD^{-1}$ sonucu başarılı olamadığından dolayı kesin bir sonuca varılamaz. x_1 ve x_2 görüntülerine üssel fonksiyon kullanılarak gri seviyesi dönüşümü uygulandığında, (f) ve (g) şekillerinde görülen y_1 ve y_2 görüntüleri elde edilmiştir. (h) şeklinde ise y_1 ve y_2 'nin gri seviyesi aralığı verilmektedir. Buna göre y_1 ve y_2 görüntülerinin gri seviyeleri 50-250 arasında değerlerden oluşmaktadır. Böylece nesne ve arka plan arasındaki ayırt edilebilirlik artırılmıştır. y_1 ve y_2 görüntüleri üzerinde sözde faz hesaplaması yapılarak (i) ve (i) sonuçları elde edilmiştir. Buna göre $AKSD^{-1}$ ve $ASKD^{-1}$ sonuçları $x=0$ ve $y=0$ doğruları göz ardı edildiğinde doğru sonucu vermektedir. Sonuç olarak nesnenin ayırt edilebilirliğinin düşük olduğu görüntüler üzerinde üssel fonksiyon ile gri seviyesi dönüşümü uygulanarak sözde faz hesaplamasının başarımı artırılabilir.

Şekil 16'da (b) x_2 görüntüsü, (a) x_1 görüntüsünde arka plana MSE hesaplamasına göre 16,3392, nesneye ise



Şekil 13. Öteleme miktarının AXD-HT üzerindeki etkisi (Impact of motion magnitude on DXT-ME)

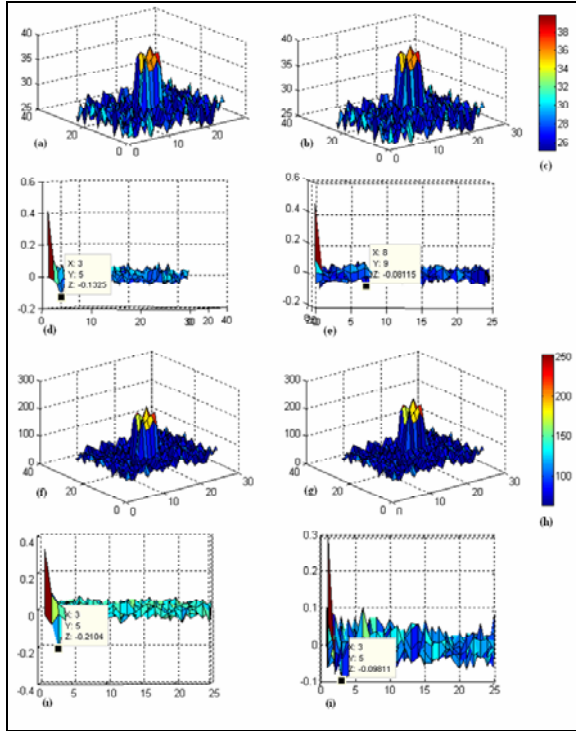


Şekil 14. Eşiklemenin AXD-HT üzerinde etkisi (Impact of thresholding on DXT-ME)

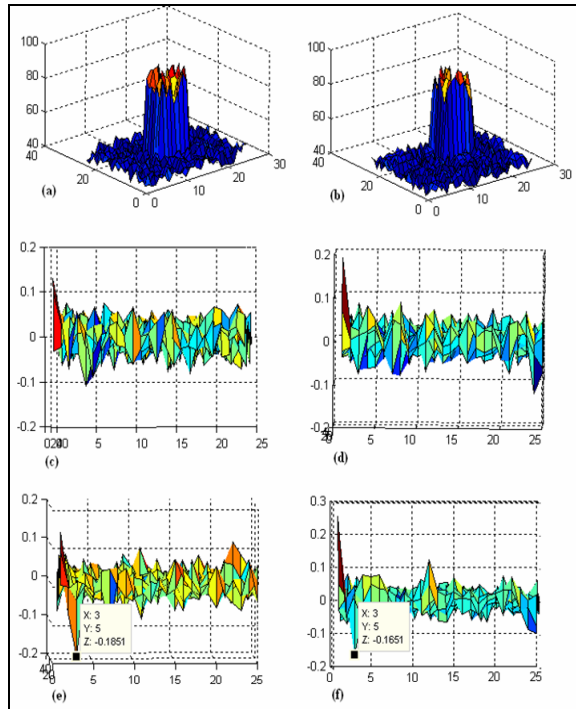
85,5102 hata eklenmiş ve x ekseninde 3 birim y ekseninde 5 birim ötelenmiş halidir. Sözde faz hesaplaması doğrudan x_1 ve x_2 görüntüsü üzerine uygulandığında, sonuç grafikleri (c) ve (d) şekillerinde görüldüğü üzere başarılı olamamaktadır. x_1 ve x_2 üzerinde 3x3 lük pencere kullanılarak medyan süzgeç uygulanmak suretiyle gürültü temizlenmiş ve elde edilen görüntüler üzerinde sözde faz hesaplaması yapıldığında (e) ve (f) şekillerinde görülen sonuç grafikleri elde edilmiştir. Bu sonuçlara göre gürültü temizlendikten sonra sözde faz hesaplaması başarılı sonuç vermektedir.

6. SONUÇLAR (RESULTS)

Sözde faz hesaplaması, gerçek faz bilgisi içermeyen AKD ve ASD katsayıları üzerinden faz bilgisinin çıkartılmasıdır. AKD tabanlı sözde faz hesaplamasına dayalı hareket tahmini yönteminin işlem karmaşıklığı, blok eşleştirme algoritmasına göre daha azdır. Ayrıca AKD tabanlı algoritma, sıkıştırma ve hareket tahmini



Şekil 15. Gri seviyesi dönüşümünün AXD-HT üzerinde etkisi (Impact of gray-level transformation on DXT-ME)



Şekil 16. Medyan süzgeçlemenin AXD-HT üzerinde etkisi (Impact of median filtering on DXT-ME)

işlemlerini birleştirdiği için video kodlama karmaşıklığını da azaltmaktadır.

Görüntü dizisi içerisinde hareket tahmini yapılacak görüntülerdeki gürültüler ve nesne ile arka plan arasındaki ayırdedilebilirliğin düşük olması sözde faz hesaplamasının başarısını olumsuz etkilemektedir. Bu

çalışmada, sözde faz ile hareket tahmini yönteminin her durumda başarılı bir şekilde çalışabilmesi için bazı görüntü iyileştirme işlemlerinin kullanılması önerilmiştir.

Başarıyı artırabilmek amacıyla, uzaysal bölgede görüntüye medyan süzgeci uygulanarak gürültünün temizlenmesi amaçlanmıştır. Medyan süzgeci, görüntü içerisindeki şekillerin kenarlarını korumaktadır. Böylece hareket eden nesnenin hareket bilgisi kaybolmamakta ve gürültü temizlenerek sözde faz hesaplaması başarılı bir şekilde yapılabilmektedir. Yapılan deneylerde görüntüde arka plana MSE hesaplamasına göre 16,3392, nesneye ise 85,5102 hata eklendiğinde klasik sözde faz hesaplamasının doğru sonucu veremediği görülmüştür. Ancak medyan süzgeci ile gürültü temizlendikten sonra tekrar sözde faz hesaplaması yapıldığında, yöntemin başarılı sonuç verdiği görülmüştür. Nesne ile arka plan arasındaki ayırt edilebilirliğin düşük olmasının, sözde faz hesaplamasını olumsuz etkilediği görülmüştür. Bu sorunun çözümü olarak da iki yöntem uygulanmış ve her ikisinde de başarılı sonuç alınmıştır. Bu yöntemlerden ilki, uzaysal bölgede üssel fonksiyon ile gri seviyesi dönüşümü yapılmasıdır, bu sayede nesnenin daha belirgin hale gelmesi sağlanmıştır. İkinci olarak da eşikleme uygulanarak nesne ile arka planın ayrılması sağlanmıştır. Sonuçta iki görüntü elde edilerek nesne ile arkaplan kesin bir şekilde birbirinden ayrılabilir hale gelmiştir. Ancak nesne ile arka plan gri seviyesi aralıkları birbirine yakın olmasından ve her görüntü için aynı olamayacağından dolayı eşikleme Otsu'nun yöntemi ile uyarlanır bir şekilde yapılmıştır. Sonuç olarak önerilen iyileştirmeler ile sözde faz hesaplamasına dayalı hareket tahmini yönteminin, zor şartlarda bile başarılı bir şekilde çalıştığı görülmüştür.

KAYNAKLAR (REFERENCES)

1. Roma, N. ve Sousa, L., "Least squares motion estimation algorithm in the compressed DCT domain for H.26x/MPEG-x video sequences", **IEEE Conference on Advanced Video and Signal Based Surveillance**, Lisboa, 576-581, 2005.
2. Božinović, N. ve Konrad, J., "Motion analysis in 3D DCT domain and its application to video coding", **Signal Processing: Image Communication**, Cilt 20, No 6: 510-528, 2005.
3. Yip, P. Rao, K., "On the shift property of DCT's and DST's", **IEEE Transactions on Acoustics, Speech, and Signal Processing**, Cilt 35, No 3: 404-406, 1987.
4. Bruni, V., De Canditiis, D., Vitulano, D., "Phase-based Motion Estimation for Noisy Sequences", **14th International Workshop on Systems, Signals and Image Processing, 2007 and 6th EURASIP Conference focused on Speech and Image Processing, Multimedia Communications**

- and Services, Rome, 381-384, 2007.
5. Koc, U. ve Liu, R., "DCT Based Motion Estimation", **IEEE Transactions on Image Processing**, Cilt 7, No 7: 948-965, 1998.
 6. Koc, U. Ve Liu, R., "Interpolation-free subpixel motion estimation techniques in DCT domain", **IEEE Transactions on Circuits and Systems for Video Technology**, Cilt 8, No 4: 460-487, 1998.
 7. Koc, U. Ve Liu, R., "DCT-based subpixel motion estimation", **IEEE International Conference on Acoustics, Speech, and Signal Processing**, Atlanta, 1930-1933, 1996.
 8. Koc, U. Ve Liu, R., "Motion Compensation on DCT Domain", **EURASIP Journal on Applied Signal Processing**, Cilt 2001, No 3: 147-162, 2001.
 9. Viitanen, M., Kolinummi, P., Hämäläinen, T., Saarinen, J., "Scalable DSP implementation of DCT-based motion estimation algorithm", **European Signal Processing Conference (EUSIPCO 2000)**, Tampere, Finlande, 251-254, 2000.
 10. Oh, S.-K., Park, H.W., "Analysis of IDCT and motion-compensation mismatches between spatial-domain and transform-domain motion-compensated coders", **IEEE Transactions on Circuits and Systems for Video Technology**, Cilt: 15, No: 7, 835- 843, 2005.
 11. Yalçın, S., **H.264 Motion Estimator Design**, Yüksek Lisans Tezi, Sabancı Üniversitesi, Fen Bilimleri Enstitüsü, 2005.
 12. Zhu, s. ve Ma, K. K., "A new diamond search algorithm for fast block-matching motion estimation", **IEEE Transactions on Image Processing**, Cilt 9, No 2: 287-290, 2000.
 13. Chen, Y. S., Hung, Y. P. ve Fuh, C. S., "Fast block matching algorithm based on the winner-update strategy", **IEEE Transactions on Image Processing**, Cilt 10, No 8: 1212-1222, 2001.
 14. Liu, L. K. ve Feiq, E., "A block-based gradient descent search algorithm for block motion estimation in video coding", **IEEE Transactions on Circuits and Systems for Video Technology**, Cilt 6, No 4: 419-422, 1996.
 15. Çontar, E., **Mikrofon Dizilerinde Ses Kaynağının Genetik Algoritma Kullanılarak Bulunması**, Yüksek Lisans Tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, 2007.
 16. Reddy, B.S. ve Chatterji, B.N., "An FFT-based technique for translation, rotation, and scale-invariant image registration", **IEEE Transactions on Image Signal Processing**, Cilt 5, No 8: 1266-1271, 1996.
 17. Koc, U. Ve Liu, R., "Discrete-Cosine/Sine-Transform Based Motion Estimation", **IEEE International Conference on Image Processing**, Austin, 771-775, 1994.