

GÖRÜNTÜ DÖNÜŞÜMÜNDE YAPAY SİNİR AĞLARI YAKLAŞIMI

Esra HASALTIN¹

Erkan BEŞDOK²

¹ Bilgisayar Mühendisliği Bölümü, Mühendislik Fakültesi,

Erciyes Üniversitesi, 38039, Talas, Kayseri

² Bilgisayar Anabilim Dalı, Fen Bilimleri Enstitüsü,

Erciyes Üniversitesi, 38039, Talas, Kayseri

¹e-posta: esra@erciyes.edu.tr

²e-posta: ebesdok@erciyes.edu.tr

Anahtar sözcükler: Yapay sinir ağları, görüntü işleme, ayrık fourier dönüşümü

ABSTRACT

This study presents the comparison of the classical discrete fourier transform and the artificial neural network (ANN) based image transform. A new ANN approach is proposed instead of complex discrete fourier transform operations. The extended delta-bar-delta algorithm is used to train the neural network and the intensity values of the pixels and fourier coefficients of the image are used as inputs and targets. Use of ANNs reduced computational burden of discrete fourier transform and reduced response time. Image transform using ANNs gives accurate results that high quality output image is quantitatively and also qualitatively acceptable. According to the results of this study the new ANN approach is useful in image processing especially in image compression subjects because of high amount of pixel operations.

1. GİRİŞ

Ayrık veya sürekli görüntü dönüşüm tabanlı görüntü işleme ve analizleri klasik yaklaşımlardır. Dönüşümler yaygın olarak görüntü filtreleme, görüntü sıkıştırma, görüntü analizi gibi uygulamalarda kullanılırlar. Görüntü işlemede kullanılan başlıca dönüşümler; Fourier, Walsh, Hadamard, Wavelet Ayrık Kosinüs, Haar ve Slant dönüşümleridir [1, 2].

Bu çalışmada görüntü işleme problemlerinde yaygın olarak kullanılan ve klasik bir yaklaşım olan iki boyutlu ayrık fourier dönüşümü (FFT) ile yapay zeka tekniklerinden biri olan yapay sinir ağları kullanılarak gerçekleştirilen dönüşümlerin performansları karşılaştırılmıştır.

2. AYRIK FOURIER DÖNÜŞÜMÜ

Fourier dönüşümü görüntüyü sinüs ve kosinüs bileşenlerine ayıran önemli bir görüntü işleme tekniğidir. Dönüşüm çıktısı frekans domenindedir. Çıktıdaki her bir nokta görüntü üzerinde belirli bir frekansı ifade eder [3].

Ayrık fourier dönüşümünde ise görüntünün uzaysal olarak ifade edilmesine yetecek sıklıkta örneklenmiş frekans aralıkları üzerinde çalışılır. Bütün frekanslar kullanılmaz [3].

$M \times N$ boyutunda bir matrisin ayrık fourier dönüşümü hesaplanırken 1 nolu eşitlik kullanılır [1].

$$F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \exp[-j2\pi(ux/M + vy/N)] \quad (1)$$

Ters dönüşüm matrisini elde etmek için de 2 nolu eşitlik kullanılır [1].

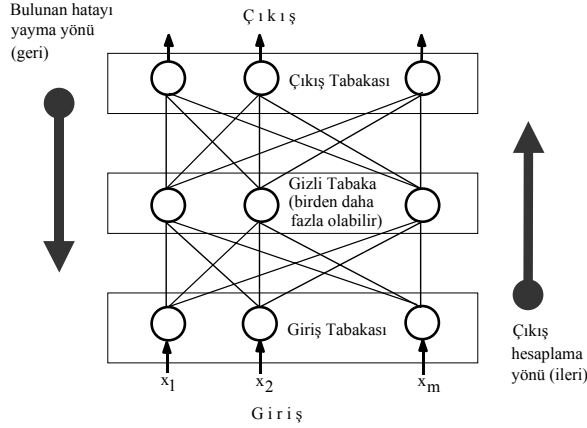
$$f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) \exp[j2\pi(ux/M + vy/N)] \quad (2)$$

3. YAPAY SİNİR AĞLARI

İnsan beyni bilinen en karmaşık hesaplayıcıdır ve Yapay sinir ağları (YSA) çalışma metodu açısından insan beyninin işleyişini taklit etmektedir.

Literatürde birçok YSA yapısı mevcuttur. Bu ağlar farklı şekillerde sınıflandırılabilir. Bazı kaynaklarda ileri veya geri beslemeli olarak ikiye ayrılırken bazı kaynaklarda dinamik veya statik, veya danışmanlı veya danışmansız olmak üzere farklı şekilde sınıflandırılabilirler. MLP, RBFN,

LVQ, PNN, GNN ağları ileri beslemeli danışmanlı ağlar iken BAM, Backpropagation through time, Elman, Jordan, Real-time recurrent network, Recurrent backpropagation, TDNN ise geri beslemeli ağlardır. Bir çok katmanlı perseptron sinir ağı modeli, Şekil 1’de gösterilmiştir. Bu ağ modeli özellikle mühendislik uygulamalarında en çok kullanılan sinir ağı modeli olmuştur [4].



Şekil-1 Geri yayılım MLP yapısı

Bu çalışmada öğrenme algoritması olarak Genişletilmiş Delta-Bar-Delta (EDBD) kullanılmıştır.

Genişletilmiş Delta-Bar-Delta: Genişletilmiş Delta-Bar-Delta (EDBD) öğrenme algoritması eğitimde bulunan epokun genel hatasının hesaplanmasına dayanan ve az kullanılan bir hata düzeltme metodu uygular [5]. Bulunulan epok sırasında ölçülen hata bir önceki dönemin hatasından büyük ise ağırlıklar bir önceki ağırlık değerlerini geri yükler. Daha iyi sonuçlar elde etmek için hata düzeltme özelliğine dayanıklılık faktörü eklenmiştir. Her epoktaki hatayı test etmek yerine daha önce iyileştirildiği için hata n. epokta test edilir. EDBD’nin geliştirilmesinde aşağıdaki durumlar göz önüne alınmıştır.

- EDBD algoritması sezgisel momentum kullanmaz.
- Bu algoritmada öğrenme hızını yeterince artırmak için lineer, küçük bir artış yapılması ağırlık uzayında büyük atlamalara sebep olabilir.
- Geometrik azaltma bazı durumlarda amacı aşan derecede büyük atlamaları engellemek için yeterince hızlı değildir. EDBD’de, her bir bağlantıdaki momentum hızı $\mu(k)$ ve öğrenme hızı $\alpha(k)$ değerleri zamanla değişmektedir. Karşılaştırma yapmak için standart delta kuralı ile verilen momentum için bağlantı ağırlık güncelleştirilmesi,

$$\Delta w(k+1) = \alpha \delta(k) + \mu \Delta w(k) \quad (3)$$

ile tanımlanır ve buradan da,

$$w(k+1) = w(k) + \Delta w(k+1) \quad (4)$$

elde edilebilir. Burada α ve μ sabittir. EDBD için değişken olan öğrenme ve momentum hızları şu şekilde ele alınmaktadır :

$$\Delta w(k+1) = \alpha(k) \delta(k) + \mu(k) \Delta w(k) \quad (5)$$

$$w(k+1) = w(k) + \Delta w(k+1) \quad (6)$$

Burada $\mu(k)$ momentum hızı ve $\alpha(k)$ öğrenme hızı aşağıdaki kurallara göre

hesaplanır. Öncelikle, $\bar{\delta}$ ’nin hesaplanması için DBD algoritmasından faydalanılarak :

$$\bar{g}(k) = (1 - \theta)(\delta(k) + \theta \delta(k-1)) \quad (7)$$

hesaplanır. EDBD için öğrenme hızı değişimi :

$$\Delta \alpha(k) = \begin{cases} \kappa_\alpha e^{-\gamma_\alpha |\bar{\delta}(k)|} & \bar{\delta}(k) g(k) > 0 \\ -\varphi_\alpha \alpha(k) & \bar{\delta}(k) g(k) < 0 \\ 0 & \text{Diğer Durumda} \end{cases} \quad (8)$$

ile sağlanır. Burada expotansiyel ($e^{-\gamma_\alpha |\bar{\delta}(k)|}$) üstel fonksiyonunu, κ_α sabit öğrenme hızı ölçek faktörünü ve γ_α sabit öğrenme hızı üstel faktörünü göstermektedir. Benzer şekilde momentum hızındaki değişim,

$$\Delta \mu(k) = \begin{cases} \kappa_\mu \exp(-\gamma_\mu |\bar{\delta}(k)|) & \bar{\delta}(k) g(k) > 0 \\ -\varphi_\mu \mu(k) & \bar{\delta}(k) g(k) < 0 \\ 0 & \text{Diğer Durumda} \end{cases} \quad (9)$$

ile hesaplanır. Burada κ_μ sabit momentum hızı skala faktörünü ve γ_μ sabit momentum hızı üstel faktörünü gösterir. Orijinal DBD algoritmasında yapılan değişiklikler aşağıda verilmiştir.

$\mu(k)$: k zamanındaki bağlantı momentum hızı

$\Delta \mu(k)$: k zamanındaki bağlantı momentum hızı değişimi

φ_α : Sabit öğrenme hızı azaltma faktörü

φ_μ : Sabit momentum hızı azaltma faktörü

α_{max} : Öğrenme hızının üst sınırı

μ_{max} : Momentum hızının üst sınırı

λ : Düzeltme tolerans parametresi

Öğrenme ve momentum hızları onların azalması veya artmasını kontrol eden farklı sabitlere sahiptir. $\bar{g}(k) g(k)$,’nin işareti ilgili hızlarda sezgisel olarak artma mı yoksa azalma mı yapılması gerektiğini tesbit etmek için kullanılır. Azalmanın miktarı EDBD’nin kurallarına göre belirlenir.

Öğrenme ve momentum oranlarının artırılması durumunda $|\bar{g}(k)|$,’nin üstel azalan fonksiyonu değiştirilir. Böylece küçük gradiyentli bölgelerde büyük gradiyentli bölgelere nazaran daha büyük artırımlar söz konusu olur. Bu çözüm atlama

problemi için kısmi bir çözümdür. Ağırlık uzayında osilasyonları ve aşırı salınımları engellemek gerekir. Bu amaçla her bir yapay nöronun öğrenme hızları ve momentum hızları üzerine üst sınırlar yerleştirilir. Bu yaklaşımın matematiksel ifadesi tüm bağlantılar için aşağıdaki şekilde tanımlanır :

$$\alpha(k) \leq \alpha_{max} \quad (10)$$

$$\mu(k) \leq \mu_{max} \quad (11)$$

Bu eşitliklerde α_{max} öğrenme katsayısının üst sınırını ve μ_{max} ise momentumun üst sınırını ifade eder. Sonuçta EDBD algoritmasına bir hafıza özelliği kazandırılmış olur. Bu hafıza kullanılarak her epoktan sonra birikmiş hata değerlendirilir.

Hata değeri $E(k)$ bir önceki minimum hatadan küçük ise ağırlıklar o anki en iyiler olarak hafızada depolanır. Tolerans parametresi λ düzeltme sürecini kontrol etmektedir. O andaki hata önceki minimum hatayı aşarsa,

$$E(k) > E_{min} \lambda \quad (12)$$

olur. Bu durumda tüm ağırlıklar hafızada saklı olan en iyi değerlere geri dönerler. Bu durumda öğrenme ve momentum hızlarını düzeltmeyi tekrar başlatmak için bu değerleri azaltmak gerekir [5].

4. BİKÜBİK İNTERPOLASYON

Görüntüleri büyütmek için çeşitli yöntemler mevcuttur. Ancak en etkili yöntem pikseller arasındaki renklerin interpolasyonunu gerektirir. En yakın komşu, lineer ve bikübik olmak üzere yaygın olarak kullanılan üç interpolasyon yöntemi vardır. Bu üç yöntem arasında en basit olanı en yakın komşu yöntemi olup çıktı resimde mozaikleşme görülür. Lineer interpolasyonda bu keskin köşeler elimine edilmiş, daha düzgün bir görüntü elde edilmiş olsa da bulanıklık mevcuttur. Genellikle bu üç yöntem içinde bikübik interpolasyon en iyi sonucu verdiği için tercih edilir. En yakın komşu yöntemindeki köşeler ve lineer interpolasyondaki bulanıklık dezavantajlarını elimine eder en iyi çıktı resmi üretir [6, 7].

Bikübik interpolasyon, ilgili pikselin 4×4 komşuluğundaki 16 pikseli kullanarak yeni pikselin renk bilgisini oluşturur. Yeni değerin hesaplandığı Eşitlik 13'de m ve n ilgili pikselin 4×4 komşuluğundaki pikselleri ifade eder [7].

$$F(i', j') = \sum_{m=-1}^2 \sum_{n=-1}^2 F(i+m, j+n) R(m-dx) R(dy-n) \quad (13)$$

Eşitlik 14 kübik ağırlık fonksiyonudur.


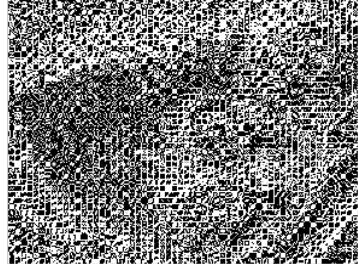


$$R(x) = \frac{1}{6} [P(x+2)^3 - 4P(x+1)^3 + 6P(x)^3 - 4P(x-1)^3] \quad (14)$$

$$P(x) = \begin{cases} x & x > 0 \\ 0 & x \leq 0 \end{cases} \quad (15)$$



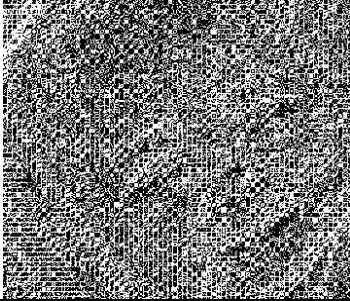

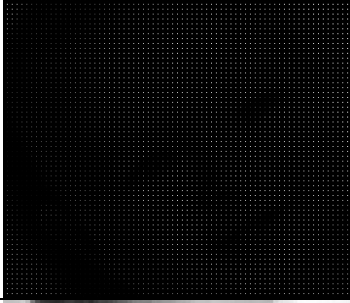


5. YAPAY SİNİR AĞLARI İLE GÖRÜNTÜ DÖNÜŞÜMÜ

Bu çalışmada kullanılan yapay sinir ağında 25 giriş, 1 çıkış ve arakatta 10 nöron kullanılmıştır. Sigmoid transfer fonksiyonu arakat ve çıkış nöronlarında kullanılmıştır. YSA eğitimi 20 epok ile sınırlandırılmıştır.

Eğitim kümesi olarak Şekil 2(a)'daki 290×220 boyutlu görüntü kullanılmıştır. Orijinal görüntünün $[5 \ 5]$ 'lik blokları YSA'ya giriş olarak verilmiştir. YSA çıkışında ise her bir blok için hesaplanan maksimum FFT katsayısı kullanılmıştır. Eğitim sonunda RMS hata değeri 0.0156 ve korelasyon 0.9907 olarak hesaplanmıştır.

	(a) Orijinal eğitim seti görüntüsü
	(b) Orijinal görüntünün $[5 \ 5]$ boyutlu FFT'si
	(c) Her bir $[5 \ 5]$ bloktaki max. elemanların konumları. Toplam max. eleman sayısı $[44 \ 58]$ olarak hesaplanmıştır
	(d) a'nın $[5 \ 5]$ blok FFT'sinin sadece max. elemanı kullanılarak elde edilen ters FFT görüntüsü.

Şekil-2 Eğitim seti üzerindeki işlemler

	(a) Orijinal test seti görüntüsü		(e) YSA test sonucu, görüntünün [0 – 255] gritonları arasında sayısallaştırılmış şekli.
	(b) (a) görüntüsünün [5 5] boyutlu FFT'si		(f) (e) görüntüsünün biküçük interpolasyon yöntemi ile orijinal boyutlara ölçeklendirilmiş şekli.
	(c) Herbir [5 5] bloktaki max. elemanların konumları. Toplam max. eleman sayısı [62 72] olarak hesaplanmıştır		(g) (e) görüntüsünün ters FFT dönüşümü uygulanmış şekli
	(d) a'nın [5 5] blok FFT'sinin sadece max. elemanı kullanılarak elde edilen ters FFT görüntüsü.		

Şekil-3 Test seti üzerindeki işlemler

6. SONUÇLAR

Şekil-2'de (a) orijinal görüntü ile maksimum FFT katsayıları kullanılarak ters FFT dönüşümü uygulanarak elde edilmiş (d) görüntüsü arasındaki korelasyon 0.9829 olarak hesaplanmıştır.

Şekil-3'de test seti görülmektedir. (a) orijinal görüntü ile maksimum FFT katsayıları kullanılarak ters FFT dönüşümü uygulanarak elde edilmiş (d) görüntüsü arasındaki korelasyon 0.9919 olarak hesaplanmıştır.

Şekil-3'de (a) orijinal görüntü ile YSA sonucu elde edilen FFT katsayılarının biküçük interpolasyon ile orijinal boyuta ölçeklendirilmiş (f) görüntüsü arasındaki korelasyon 0.9920 olarak hesaplanmıştır.

Şekil-3'de (a) orijinal görüntü ile YSA sonucu elde edilen FFT katsayıları kullanılarak ters FFT dönüşümü uygulanan (g) görüntüsü arasındaki korelasyon 0.9913 olarak hesaplanmıştır.

Çalışma sonucunda;

- Hem kantitatif hem de kalitatif olarak kabul edilebilir düzeyde yüksek kalitede bir yaklaşım önerilmiştir.
- FFT kullanılarak gerçekleştirilen dönüşüm sonuçları ile YSA kullanılan görüntüler karşılaştırıldığında performans bakımından fark görülmemektedir.

- FFT işlemlerindeki hesaplama yükü, YSA kullanımı ile hafifletilmiştir. Hızlı cevap verebilen bir sistem tasarımı için bir defa eğitim yapıldıktan sonra çok daha hızlı cevap verebilen YSA kullanımı avantajlıdır. Özellikle görüntü işleme alanında büyük boyutlu matrisler ile işlem yapıldığı göz önünde bulundurulursa klasik yöntemlere göre YSA'nın avantajları ön plana çıkmaktadır.
- Lurawave [8] teknolojisinde olduğu gibi Görüntü sıkıştırma işlemlerinde kullanılacak düzeyde kaliteli ve hızlı cevap verebilen bir yapıya sahiptir.

5. KAYNAKLAR

- [1] Gonzales, Rafael, C., Woods, Richard, E., Digital Image Processing, s. 81-159, Addison Wesley Publishing Company, USA, 1993.
- [2] <http://www.icaen.uiowa.edu/~dip/LECTURE/>

- LinTransforms.html
- [3] <http://homepages.inf.ed.ac.uk/rbf/HIPR2/fourier.htm>
- [4] Sağiroğlu, Ş., Beşdok, E., Erler, M., Mühendislikte Yapay Zeka Uygulamaları – I: Yapay Sinir Ağları, s. 23 -103, Ufuk Kitap Kirtasiye – Yayıncılık Ltd. Şti, Kayseri, 2003.
- [5] Beşdok, E., Raster Veritabanlarının Yapay Sinir Ağları ile Yorumlanması, Doktora Tezi, İstanbul Teknik Üniversitesi, İstanbul, 2000.
- [6] Sonka, M., Hlavac, V., Boyle, R., Image Processing, Analysis and Machine Vision, s. 65-68, BROOKS/ COLE Publishing Company, USA, 1999.
- [7] <http://astronomy.swin.edu.au/~pbourke/colour/bicubic/>
- [8] <http://www.acdsystems.com/English/Products/LuraWave/overview.htm?LAN=englishX10>