

# ÖRÜNTÜ TANIMADA HOPFIELD AĞININ KULLANILMASI

Necmettin Sezgin<sup>1,\*</sup>, Ramazan Tekin<sup>2</sup>, Abidin Çalışkan<sup>2</sup>

<sup>1</sup> Batman Üniversitesi, Müh.Mim. Fakültesi, Elektrik-Elektronik Müh. Bölümü, 72100,  
Batman

<sup>2</sup> Batman Üniversitesi, Müh.Mim. Fakültesi, Bilgisayar Müh. Bölümü, 72100, Batman  
\*necmettin.sezgin@batman.edu.tr

**Özet:** Bilgisayar teknolojisinin hızlı bir şekilde gelişmesi akıllı sistemlerin insan yaşamının birçok alanında kullanılmasını artırmıştır. Bu alanlardan birisi de alfa nümerik karakterlerin otomatik olarak doğru bir şekilde tanınması, istenen bir objenin tespit edilmesi ve seçilmesidir. Hopfield ağı, gürültülü veya bozuk olan desenin kısmi ipuçlarından ve önceden depolanmış desenlerden yararlanarak bu deseni düzeltebilen karakteristik bir yapıya sahiptir. Bu süreçte ağ, girdi örüntüsünde yapılan her ufak değişimin ardından örüntü enerjisini yeniden hesaplayarak morfolojik dönüşümünün kontrolünü sağlar ve bu örüntünün daha önce öğrendiği başka bir örüntüye yakınsamasını zorlar. Bu benzetişim işlemi, enerjideki değişkenlik durağan olana dek sürer. Nesnelerin otomatik olarak tanınması, seçilmesi ve işlenmesi gibi işlemlerden sorumlu bir ağına kullanıldığı akıllı sistemler özellikle robotik alanında önemli bir yere sahiptir. Bu çalışmada Hopfield ağ yapısını kullanarak örüntü tanıyan bir sistem geliştirilmeye çalışılmıştır.

**Anahtar Kelimeler:** Hopfield ağı, Karakter tanıma, Örüntü tanıma.

## Using Hopfield Network in Pattern Recognition

**Abstract:** The development of computer technology has increased the use of intelligent systems in many areas of human life. One of these areas is automatically and correctly identification of the alphanumeric characters and patterns as well as selection of the desired objects. Benefiting from the partial characteristics of patterns and previously stored characters or patterns, Hopfield Network has a characteristic feature to not only identify but also correct/reconstruct the noisy and destroyed characters or patterns. In the process the network re-calculates the energy of the character or pattern after even a small change to the pattern given to the input of the network, and hence, controls the morphologic transform and forces the noisy pattern to converge to a previously learned and stored nearest pattern. This simulation continues until the variation in the energy becomes stable. Intelligent systems, which are built in neural networks responsible for identifying choosing and processing, have important roles particularly in robotics. In this study a system consisting of Hopfield network model was developed for identifying the patterns of alphanumeric characters.

**Keywords:** Hopfield network, character identification, pattern recognition.

## 1. GİRİŞ

Yazılı dokümanların bilgisayarlar tarafından kullanılabilir formatlara çevirme gereksinimi gün geçtikçe artmaktadır. Bir şeklin veya yazının bilgisayar ortamına el ile giriş maliyetinin yüksek oluşundan bu bilgilerin otomatik olarak tanınup dijital ortama aktarılması gerekmektedir. Bu bağlamda alfa nümerik karakterlerin otomatik olarak doğru bir şekilde tanınması, istenen bir objenin doğru bir şekilde tespit edilmesi ve seçilmesi gibi görevleri yerine getiren akıllı sistemlerin geliştirilmesi önem taşımaktadır. Temelde biyolojik sinir hücre yapısından esinlenerek geliştirilen Yapay Sinir Ağı (YSA) bu akıllı sistemlerinin en önemli üyesidir. Günümüzde uygulama alanına göre tasarlanmış farklı YSA modellerine rastlamak mümkündür (Ufer, 1970). YSA'nın en önemli uygulamalarından biri örüntü tanımadır, ki bu alanda ileri yönlü sinir ağı geliştirilerek önemli uygulamalar yapılmıştır. YSA, girişe verilen örüntü işaretlerinden istenilen çıkış örüntüsü elde edilinceye kadar eğitilir ve daha sonra da örüntü tanıma sisteminde yerini alır. YSA modellerinden biri olan Hopfield ağı, gürültülü veya bozuk olan desenin kısmi ipuçlarından ve önceden depolanmış desenlerden yararlanarak bu deseni düzeltebilen karakteristik bir yapıya sahiptir. Bu süreçte ağ girdi örüntüsünde yapılan her ufak değişimin ardından örüntü enerjisini yeniden hesaplayarak morfolojik dönüşümünün kontrolünü sağlar ve bu örüntünün daha önce öğrendiği başka bir örüntüye yakınsamasını zorlar. Bu benzetişim işlemi, enerjideki değişkenlik durağan olana dek sürer. Nesnelere otomatik olarak tanınması, seçilmesi ve işlenmesi gibi işlemde sorumlu bir ağın kullanıldığı akıllı sistemler özellikle robotik alanında önemli bir yere sahiptir. Bu çalışmada Hopfield ağı kullanılarak karakter tanıma işlemi yapılmıştır. Geliştirilen bu YSA modelinin yapay zeka alanında, özellikle robotik alanda uygulanabileceği düşünülmektedir.

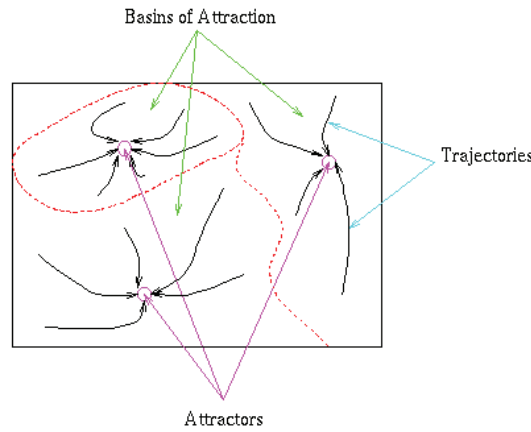
## 2. YÖNTEM

### 2.1. Verilerin oluşturulması

Çalışma için arial A-Z harflerinden oluşan bir veri kümesi oluşturuldu. Her bir karakter 48x48 pikselden oluşan BMP formatında resim olarak kayıtlı edildi. Her bir karakter için 10'ar defa rastgele üretilen 48x48 boyutunda gürültü üretildi ve A-Z alfabetik karakterlere eklenip çıkartılarak test veri kümesi oluşturuldu. Her bir küme 260 adet karakterden olmak üzere, %25 gürültülü, %50 gürültülü ve %75 gürültülü karakterlerden üç adet veri kümesi oluşturuldu. Bu kümelerdeki test verilerin tamamı Hopfield ağına uygun olması açısından -1 ile 1 aralığında normalize edildi. Normalizasyon fonksiyonu olarak da hiperbolik tanjant kullanıldı.

## 2.2. Hopfield Ağı

Hopfield ağı çağrışımsal (associative) bir ağ modelidir. Diğer yapay sinir ağ modellerinden farklı olarak tek bir katmana sahip olup giriş ve çıkış katmanı aynıdır. Katmandaki hücre sayısına bağlı olarak sınırlı bir kapasitesi vardır. Hopfield ağındaki her bir hücre ya açık (+1) ya da kapalı (-1) olarak iki duruma sahip olup, ağıdaki  $i$  ve  $j$  hücre çifti  $a_{ij}$  ağırlığıyla birbirine bağlıdır.  $a_{ij}$  ağırlığı  $i$  hücresinin çıkış işaretinin  $j$  hücresi üzerindeki etkisini göstermektedir. Hopfield ağı işlemini depolama aşaması ve geri çağırma aşaması olmak üzere iki aşamada gerçekleşmektedir. Depolama aşamasında  $n$  boyutlu karakterler/desenler hafızada depolanırlar (ağın eğitim aşaması). Ağın örüntü tanımlama işlemi için bu karakterler ağın hafızasından geri çağırılıp (test aşaması) girişe verilen gürültülü karakter/desenler üzerinde bir takım kısmi ip uçlardan yola çıkarak karakterleri/desenleri eşleştirir ve gerekirse bu gürültülü örüntüleri düzeltir. Ağ bu işlemi her deseni temsil eden attractorlara (enerji açısından çukur bölgeler) sahip bir enerji yüzeyi oluşturarak yapar. Şöyle ki: gürültülü ve kısmi ipuçları verilen örüntüler sistem tarafından kontrol edilerek sahip olduğu enerji açısından hangi attractora yakınsadığını tespit ederek gürültülü örüntüyü bu yöne doğru kaydırır. Bu yakınsama işlemini tekrarlayarak (iterasyon yoluyla) sonuçta verilen örüntünün en fazla hangi attractora yakınsadığını tespit eder (Şekil 1). En ilgi çekici görsel uygulamalardan birisi bir grup resmin depolandığı bir sistemde ağa ya resimlerden birinin bir parçası (kısmi ipucu) ya da gürültülü bir resim (gürültülü ipucu) verildiğinde iterasyon yoluyla verilen ipucundan bu resmi daha önce depoladığı resimlerden bulup çıkarma işlemidir.



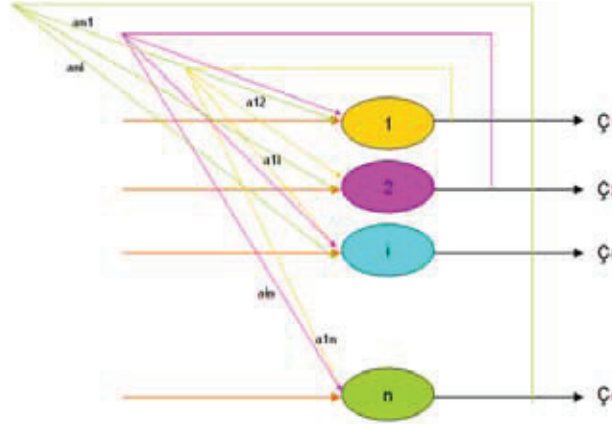
Şekil 1. Hopfield ağında enerjinin yerel minimuma yakınsaması

### 2.3. Ayrık Hopfield Ağı

Şekil 2’de mimari yapısı gösterilen Hopfield ağ modelinde  $i$ . hücreni ele alalım.  $a_{ji}$   $i$  hücresinden  $j$  hücresine bağlantının ağırlığını göstermek üzere,  $t$  zamanında ağın bu hücresinin girdisi,

$$g_i(t) = \sum_{j \neq i} a_{ji} \zeta_j(t-1) - \theta_i \dots\dots\dots (1)$$

Burada  $\zeta_j(t-1)$  bir önceki çıkışı,  $\theta_i$  ise eşik değerini göstermektedir.



Şekil 2. Hopfield ağının mimari yapısı.

Bu hücrenin çıkışı ise,

$$\zeta_i(t) = \text{sgn}(g_i(t)) = \begin{cases} +1 & \text{eger } g_i(t) > \theta_i \\ -1 & \text{eger } g_i(t) < \theta_i \\ \zeta_i(t-1) & \text{eger } g_i(t) = \theta_i \end{cases} \quad (2)$$

şeklinde olur. Bu tür modellerde eşik değeri genelde 0 alınmaktadır,  $\theta_i = 0$ , (Fausett, 1994).

Şekilden de görüldüğü gibi Hopfield ağı simetrik bir yapıya sahiptir ve herhangi bir hücrenin kendine bağlantısı yoktur (Fausett, 1994). Dolayısıyla,

$$a_{ji} = a_{ij} \quad \text{ve} \quad a_{ii} = 0 \quad (3)$$

dir. Hopfield Ağı sinaptik modülasyon yönteminin matematiksel bir çıkarımı olan Hebbian (1949) öğrenme yöntemini kullanır. Hebb'in kuralına göre, eğer alıcı nöron ateşliyorken bir nöron başka bir nöronu uyarıyorsa, iki hücre arasındaki uyarım

$$a_{ij} = a_i a_j \quad (4)$$

bağlantısı ile ağırlıklandırılır.

Matematiksel olarak;

Bir ağırlıktaki değişim bağlandığı birimlerin aktivasyonları çarpımına eşittir. Böylelikle, eğer iki birim de açıksa ( $a_i = a_j = 1$ ) veya her iki birim de kapalıysa ( $a_i = a_j = -1$ ) ağırlığın gücü artar, aksi halde azalır.

Eğer birimler dizisine bir deseni anında verip üstteki kuralı uygularsak, o desen ağın attractoru haline gelir, bu da Hopfield ağının ilgilendiği Hebbian öğrenmenin önemli bir özelliğidir. Sonuç olarak, eğer ağın aktivasyonunun deseni depolanmış bir desene yakınsıyorsa, o desene doğru gitmeye çalışacaktır.

## 2.4. Ağırlıkların Belirlenmesi

Ardışık yinelemeler sonunda ağın durağan hale gelmesi beklenir ve ağ durağan hale gelince ürettiği çıktı ilgili örnek için ağın ürettiği çıktıdır. Örneğin  $x$  ( $x_1, x_2, x_3, \dots, x_n$ ) başlangıç değerlerine sahip bir girdi için bir sonraki çıktı için ağın iterasyonları.

$$g_i(t) = \sum_{i \neq j} a_j \zeta_j(t-1) - \theta_i \quad (5)$$

$$\zeta_i(t+1) = \text{sgn}\left(\sum_{j=1, j \neq i}^n a_j \zeta_j(t) - \theta_i\right)$$

bağlantısı ile devam eder. Burada girdi vektörü başlangıç değerleri olarak atandığından,

$$\zeta(0) = x = (x_1, x_2, \dots, x_n) \quad (6)$$

$t=0$  anındaki ağ çıktısı kabul edilir. Birçok iterasyon sonucu ağın durağan hale gelmesi kullanılan enerji fonksiyonunun değerinin minimuma inmesi demektir. Bu modelde genellikle kullanılan enerji fonksiyonu ile

$$E(t) = -\frac{1}{2} \sum_{i=1}^n \sum_{j=1, i \neq j}^n a_{ji} \zeta_i(t) \zeta_j(t) - \sum_{i=1}^n x_i \zeta_i(t) + \sum_{i=1}^n \zeta_i(t) \theta_i \quad (7)$$

Tanımlıdır. Ağın çıkışındaki değişim  $\Delta \zeta_i(t)$  ise, enerjideki değişim şu şekilde olacaktır.

$$\Delta E(t) = -\left( \sum_j a_{ji} \zeta_j + x_i - \theta_i \right) \Delta \zeta_i(t) \quad (8)$$

Ağın yeterli sayıda iterasyonu sonucunda enerji fonksiyonu minimum yerele yakınsayacak ve enerjinin yakınsadığı bu yerel noktada ilgili desen bulunmuş olacaktır.

## 2.5. Depolama Kapasitesi

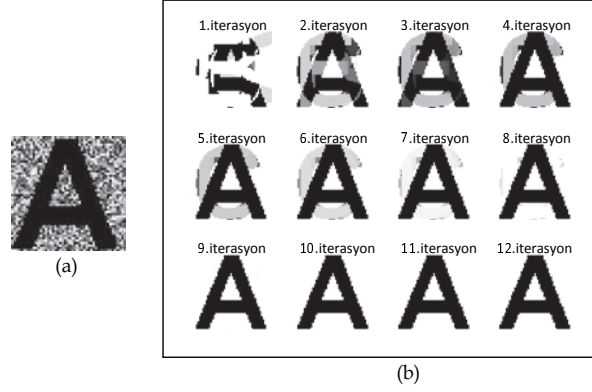
Hopfield (1982) deneysel olarak  $P \approx 0.5$   $n$  tane örüntü hafızada saklanabildiğini göstermiştir. Burada  $n$  ağdaki hücre sayısını göstermektedir. Daha sonra McEliece ve arkadaşları (1987), tane örüntünün Hopfield hafızada saklanabileceğini bulmuşlardır.

$$P \approx \frac{n}{2 \log_2 n} \quad (9)$$

Bu çalışmamızda kullanılan ağ modelinde  $n=48 \times 48$  tane hücre mevcuttur ve dolayısıyla en fazla 148 tane örüntü saklanabilir.

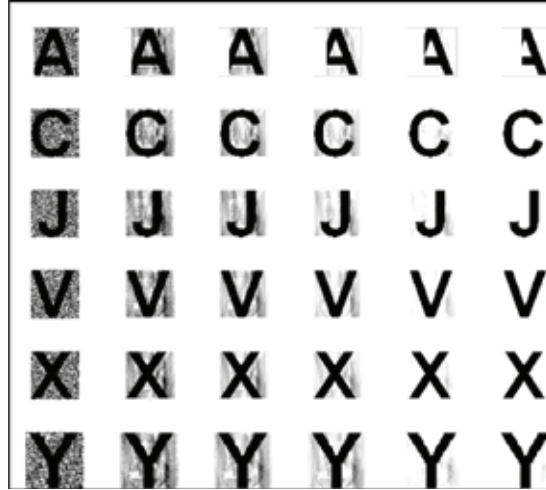
## 3. BULGULAR VE TARTIŞMA

Matlab 7.0 programı kullanılarak arial 26 karakter (A-Z) hopfield ağına verilerek ağın eğitimi gerçekleştirildi. Bu çalışmada kullanılan ağ  $n=48 \times 48$  tane hücreden oluşmaktadır. Dolayısıyla ağın giriş ve çıkış hücre sayısı 2304'tür. Daha sonra eğitilen ağda, rastgele üretilen  $48 \times 48$  boyutundaki gürültü, orijinal karaktere eklenmek suretiyle elde edilen bozuk karakter Hopfield ağına test edildi. Örnek olarak bmp formatında %40 gürültülü arial A karakteri için 1-12 iterasyonları sonucunda ağın çıkışında elde edilen karakterler aşağıda gösterilmiştir (Şekil 3). Eğitilen ağa verilen gürültülü A karakteri 12 iterasyon sonunda ihmal edilebilir bir gürültüyle doğru olarak tanınmıştır.



Şekil 3. (a) Gürültülü A karakteri, (b) Ağın 1-12 arası iterasyon çıktıları

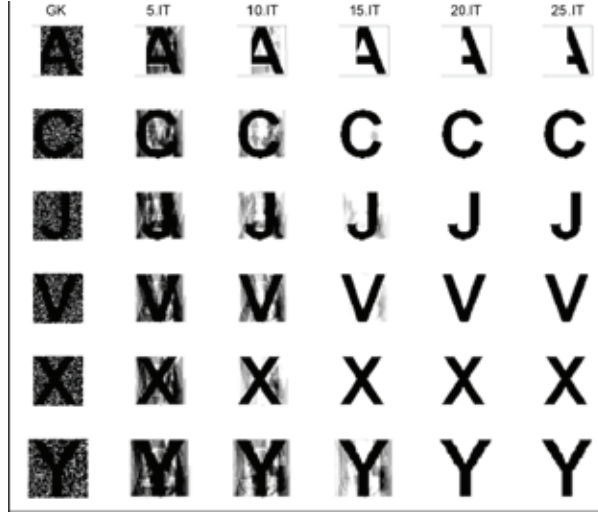
Daha sonra eğitilen ağı 10' ar defa rastgele %25, %50 ve %75 oranında gürültü eklenmiş her 3 gruptan bozulmuş 26 karakter (toplamda her grup için  $10 \times 26 = 260$  karakter) test için ağı verildi. Yeterli bir iterasyon sonucunda tasarlanan bu ağı; %25 bozulmuş grubun tamamını (%100), %50 bozulmuş grubun %97'sini ve %75 bozulmuş grubun %62.4'ünü doğru olarak tespit etmiştir. Şekil 4, 5 ve 6' da bu karakterlerden 6 tanesinin belirli iterasyonlar sonucunda ağı tarafından nasıl ve hangi karaktere yakınsadığı gösterilmiştir.



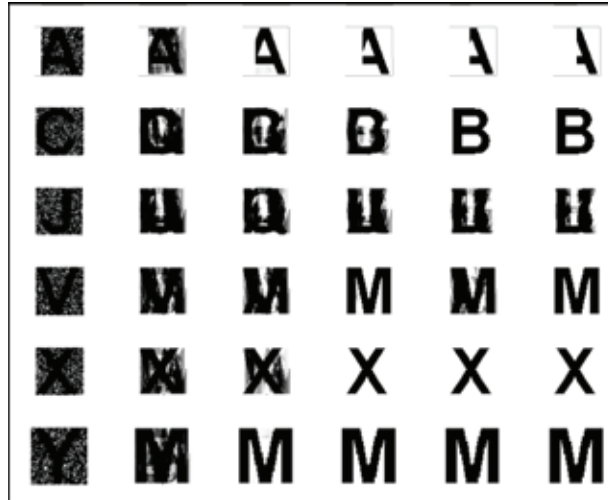
Şekil 4. %25 gürültü eklenerek bozulmuş bazı karakterlerin 1-5 arası iterasyon sonucunda ağın çıktıları

GK: Gürültülü Karakter

IT: İterasyon



Şekil 5. %50 gürültü eklenerek bozulmuş bazı karakterlerin 5, 10, 15, 20, 25 iterasyon sonucunda ağın çıkışları



Şekil 6. %75 gürültü eklenerek bozulmuş bazı karakterlerin 10, 20, 30, 40 ve 50 iterasyon sonucunda ağın çıkışları

Çizelge 1’de 3 gruba ait ağın vermiş olduğu doğruluk yüzdeleri gösterilmektedir. Karakterdeki bozulma yüzdesine göre ağın hedef karaktere yakınsaması için gerekli iterasyon sayısı artmaktadır. Buna karşın belirli bir bozulmadan sonra ağın test karakterlerine verdiği doğruluk oranı da hızlıca düşmektedir.



**Çizelge 1.** Ağın 3 grup test karakterleri için doğruluğu.

	Ağın Doğruluğu
Grup 1 (%25 bozulma)	%100
Grup 2 (%50 bozulma)	%97
Grup 3 (%75 bozulma)	%62.4

Karakter ve örüntü tanıma çabaları 1960 lardan beri süregelmiştir (Block ve arkadaşları (1962), Minsky ve Papert (1969)). Bu iş için Hussain ve Kabuka (1994) ilk aşama alt örüntüyü tanıma ve ikinci aşama karakterleri tanıma olmak üzere iki aşamalı bir ağ geliştirdiler. Maragos ve Pessoa (2000) yaptıkları çalışmada, çok katmanlı ağ ve biçimsel derece sinir ağının özelliklerini birleştirerek el yazısı karakterlerini tanımaya çalışmışlardır. Bu çalışmada birleştirilmiş yaklaşımın çok katmanlı ağa göre daha az işlem zamanıyla daha yüksek oranda karakter tanıma gerçekleştirilebileceği gösterilmiştir. Yakın zamanda Sitamahalakshmi ve arkadaşları (2010) Dempster-Shafer Teorisini kullanarak farklı mesafe ölçümlerini birleştirip karakter tanımaya çalışmışlardır. Onlar bu şekilde Benzerlik fonksiyonu, Hamming mesafesi, Doğrusal İlişki, Çapraz İlişki ve En Yakın Komşuluk mesafe ölçüm metotlarını Dempster-Shafer Teorisi ile birleştirerek en iyi hassasiyetle karakteri tanımaya çalışmışlardır. Test için kullanılan el yazısı 10x10 boyutunda resimler olup ve en iyi olasılıkla 4 rakamını %97.9 doğrulukta bulmuşlardır. Bizim çalışmamıza benzer bir çalışmada Singh ve arkadaşları (2010), 7x7 boyutundaki karakterleri tanımaya çalışmışlar. Yapılan çalışmada denklem (9)' a göre en fazla 6 karakter hopfield sinir ağında depolanabilmiş ve test aşamasında çağrılabilmiştir. Bizim çalışmamızda ise karakterler 48x48 piksel olduğundan 148 karakter depolanabilmektedir (yaklaşık 25 kat). Ancak tasarladığımız ağda giriş hücrelerinin fazla olması eğitim süresini artırmaktadır. Dolayısıyla depolanmak istenen karakter sayısına göre karakterlerin boyutunun seçilmesi eğitim süresi açısından özellikle önemlidir.

#### 4. SONUÇ

Yapay sinir ağları örüntü tanıma (pattern recognition) problemleri için kullanılan iyi bir yöntemdir. Bu çalışmamızda %25 bozulmuş 26 arial karakterin tamamının doğru olarak tespit edilmesi iyi bir başarıdır olduğu düşünülmektedir. Test karakterlerine eklenen gürültünün artırılmasıyla ağın çıkışında elde edilen doğruluk oranı düşmektedir. Örneğin: binir ağına %75 bozulmuş karakter grubu verildiğinde ağın çıkışında %62.4 oranında bir doğruluk gerçekleşmiştir. Ayrıca ağ tarafından belirli bir iterasyon sonucunda harf

üzerindeki gürültünün temizlenebileceği de gösterilmiştir. Böylece istenen bir objenin seçilmesi, istenen görüntünün veya harfin bulunması Hopfield ağıyla tespit edilebileceği sonucuna varılmıştır. Bu şekilde otomatik karakter veya örüntü tanınmanın robotik alanda kullanılabileceği düşünülmektedir.

## 5. KAYNAKLAR

- [1] Block, HD, Knight, BW, Rosenblatt, F, 1962, Analysis of A Four Layer Serious Coupled Perceptron, *II. Rev. Modern Physics*, vol.34, pp.135-152.
- [2] Fausett, L, 1994, Fundamentals of Neural Networks, Architectures, Algorithms and Applications.
- [3] Hopfield, JJ, 1982, Neural Networks and Physical Systems with Emergent Collective Computational Abilities, *Proceeding of the National academy of Scientists*, 79:2554- 2558. Reprinted in Anderson and rosenfeld (1988), pp. 460-464.
- [4] Hussain, B, Kabuka, M, 1994, A Novel Feature Recognition Neural Network and its Application to Character Recognition, *IEEE TransPattern and Machine Intelligence*, vol. 16, no. 1, pp.99-106.
- [5] McEliece, RJ, Posner, EC, Rodemich, ER and Venkatesh, SS, 1987, The capacity of the Hopfield Associative memory, *IEEE Transaction on Information Theory*, IT:33:461-482.
- [6] Minsky, ML, Papert, S, 1969, Perceptrons: An Introduction to Computational Geometry, MIT Press.
- [7] Pessoa, LFC, Maragos, P, 2000, Neural Networks with Hybrid Morphological/Rank/Linear Nodes: A Unifying Framework with Applications to Handwritten Character Recognition, *Pattern Recognition*, vol.33, pp. 945-960.
- [8] Singh, YP, Khare, A, and Gupta, A, 2010, Analysis of Hopfield Autoassociative Memory in the Character Recognition, *(IJCSE) International Journal on Computer Science and Engineering*, Vol.02, No. 03, pp. 500-503.
- [9] Sitamahalakshmi,T, Vinay Babu, A, Jagadeesh, M, 2010, Character Recognition Using Dempster-Shafer Theory combining Different Distance Measurement Methods, *International Journal of Engineering Science and Technology*, Vol. 2(5), pp. 1177-1184.
- [10] Ufer, J, 1970, Direct data processing with the IBM 1287 multipurpose document reader for standard article-fresh service to Joh. Jacob and Co. Bremen, *IBM Nachr. (Germany)*, 20, 35-40.