

Matematiksel Morfolojik İşleçlerin Rasgele Yapay Sinir Ağlarıyla Gerçeklenmesi

Tansu Küçüköncü¹, Ömer L. Gebizlioğlu²

¹Danışman, Eskişehir

²İstatistik Bölümü, Ankara Üniversitesi, Ankara

e-posta : ¹Tansu.Kucukoncu@gmail.com , ²Omer.L.Gebizlioglu@ankara.edu.tr

Özetçe

Bu çalışmada, çok girdili VE ve VEYA işleçleri Rasgele Yapay Sinir Ağı (RYSA) ile ilk kez gerçekleştirilmiştir. Matematiksel Morfolojik işleçler, RYSA ile ilk kez gerçekleştirilmiştir. Matematiksel Morfoloji ve RYSA ilk kez bir uygulamada bir arada kullanılmıştır.

1. Giriş

Bu çalışmanın amacı, Matematiksel Morfolojik işleçlerin RYSA ile gerçekleştirilmesidir. Hem problem hem de çözüm, istatistik, matematik, elektronik mühendisliği, bilgisayar mühendisliği, ve bilişsel bilimlerin çalışma konuları arasındadır ve geniş bir yelpazede uygulama alanına sahiptir.

2. İki-değerlikli Temel Morfolojik İşlemler

Matematiksel morfoloji (biçimbilim), iki değerlikli ve gri seviyeli görüntülerden geometrik bilgi çıkartan bir araçtır. Bir görüntü işleci elde etmek için *yapıtışı elemanı* (bkz. Şekil 1.a, b, c) olarak bilinen bir şekil *göstergesi* kullanılır. Görüntü işlecinin çıktısı, bu göstergenin verilen görüntüyle örtüşüp örtüşmediğine bağlıdır. Çıkarılan bilginin kullanılan göstergenin şekline ve büyüklüğüne bağlı olacaktır.

2.1. Aşınma ve Genişleme

Aşınma ve genişleme, matematiksel morfolojinin en temel işleçleridir. F_1 ve F_2 , birer küme olmak üzere, her (F_1, F_2) iki-değerlikli görüntü ikilisi için, ötelenmeden etkilenmeyen her *aşınma*, ε , (bkz. Şekil 2.b) ve her *genişleme*, δ , (bkz. Şekil 2.c) aşağıdaki şekillerde ifade edilir :

$$\varepsilon(F) = \bigcap_{b \in B} F - b = F \ominus B \quad (1)$$

$$\delta(F) = \bigcup_{b \in B} F + b = F \oplus B \quad (2)$$

burada B, bir yapıtaşı elemanıdır (bkz. Şekil 1.a, b, c).

Matematiksel morfolojinin temel önermesi *Matheron gösterimidir*. Ötelenmeden etkilenmeyen, ve artan her küme işleci aşınmaların bir birleşimi ya da genişlemelerin bir kesişimi olarak ifade edilebilir. Tam tersi de geçerlidir; bir işleç (filtre veya küme eşleştirme), W, sadece ve sadece çekirdek elemanlarının aşınmalarının birleşimi (veya genişlemelerinin kesişimi) olarak gösterilebiliyorsa ötelenmeden etkilenmeyen ve artandır [25, 35].

2.2. Açılış ve Kapanış

$F \circ B = (F \ominus B) \oplus B$ birleşimi *açılış* (veya *morfolojik açılış*), α , (bkz. Şekil 2.d) $F \bullet B = (F \oplus B) \ominus B$ birleşimi *kapanış* (veya *morfolojik kapanış*), κ , (bkz. Şekil 2.e) olarak adlandırılır [25, 35].

Bir F şeklini bir B yapıtaşı elemanı ile açmak F'nin B'den küçük tüm bileşenlerini çıkarır. Açma işleminden sonra F'nin B'nin herhangi bir ötelenmiş yansımasını içeren bir bileşeni kalmaz. Böylece, açma işleci, bir *düzleyici filtre* gibi davranır. Düzlemenin miktarı ve tipi, kullanılan yapıtaşı elemanın şekli ve büyüklüğü tarafından belirlenir. Bir F şeklini bir B yapıtaşı elemanı ile kapamak F'nin B'den küçük tüm bileşenlerini çıkarır. Kapama işleminden sonra F'nin B'nin herhangi bir ötelenmiş yansımasını içeren bir bileşeni kalmaz [15, 25, 35].

3. Rasgele Yapay Sinir Ağları

Gelenbe [6] tarafından geliştirilen *nokta-süreci* tipindeki rasgele yapay sinir ağı (RYSA) modelinde imgeler, birim şiddeteki darbeler halinde sinirler arasında dolaşır (bkz. Şekil 3). Artı imgeler uyarılma, eksi imgeler bastırılmaya karşılık gelir. Her bir *sinirin durumu*, $k_i(t)$, eksi olmayan bir tam sayıdır ve o *sinirin gizil-gücü* olarak adlandırılır.

At kadar zaman aralığı içinde bir sinirin $k_i(t)$ durumunu değiştirecek çeşitli geçişler olabilir :

a) Bir sinir ateşlediğinde, yaydığı imge ne olursa olsun (uyarıcı ya da bastırıcı), o sinirin $k_i(t)$ gizil-gücü 1 eksilecektir. Sinire ağından dışından bastırıcı bir imge geldiğinde de, $t+\Delta t$ 'de sinirin gizil-gücü $k_i(t)-1$ 'e düşer. Bunun ötesinde sinir, bir başka sinirden bastırıcı bir darbe imgesi aldığı anda da etki aynı olur, ve t anında $k_i(t)$, 1 eksilir,

b) Sinire ağından dışından ya da bir başka sinirden uyarıcı bir imge geldiğinde sinir gizil gücü 1 artarak $k_i(t)+1$ olacaktır,

c) Yukarıdaki olaylardan herhangi birisi olmazsa sinirin durumu değişmeyecek aynı kalacaktır.

n sinirden oluşan bir ağın t anındaki durumu, eksi olmayan tam sayılardan oluşan $k(t) = (k_1(t), \dots, k_n(t))$ vektörü ile gösterilir. Burada $k_i(t)$, i sinirinin gizil-gücü veya tam sayı durumudur, ve $k_i \geq 0, \forall i$ 'dir. i siniri, eğer gizil-gücü artıysa, *uyarılmış* hale gelecek, ve darbeler gönderebilecek, yani *ateşleyebilecektir*. Darbeler, bir i sinirinden diğer sinirlere ya da ağın dışına, eşit ve üstel dağılmış aralıklarla, $r(i)$ sıklığında gönderilir. Bir i siniri ateşlediğinde, ardışık imge yayımları arasındaki zamanlar, ortalama değeri $\frac{1}{r(i)}$ olan *üstel dağılımlı rasgele değişkenlerdir*.

Darbeler, bir j sinirine $p^+(i,j)$ olasılığıyla *uyarıcı*, $p^-(i,j)$ olasılığıyla *bastırıcı* imgeler olarak gidecektir (bkz. Şekil 3, 4).

$p(i,j) = p^+(i,j) + p^-(i,j)$ (3)
olsun. $p(i,j)$, imgelerin sinirler arasındaki hareketini temsil eden bir *Markov zincirinin geçiş olasılığıdır*.

Kendi kendine bastırmaya izin verilirse, bir sinirin durumunun değeri, bir anda 2 birim eksilebilir. Fakat bu durum ele alınmayacaktır. Aynı zamanda, *kendi kendine uyardırmaya* da izin verilmeyecektir. Çünkü bunun olması halinde, sinirin gizil-gücü sınırsız olarak yükselebilir, ve beraberinde kararsız durumun oluşmasına neden olabilir. Böylece, $p(i, i) = 0, \forall i$. Bazı geçişlerin oluşmasını engelleyen sınır koşulları da vardır. Herşeyden önce, bir sinir sadece pozitif gizil-güce sahip olduğunda ateşleyebilir. İkincisi, eğer bir sinirin gizil-gücü 0'sa, yeni gelen bastırıcı imgeler, onun değerini daha fazla eksiltmezler.

Bir sinir, $d(i)$ olasılığıyla, ağın dışına da imgeler gönderebilir, ve

$$d(i) + \sum_{j=1}^n [p^+(i, j) + p^-(i, j)] = 1, 1 \leq i \leq n \quad (4)$$

dir.

$$w(i, j) = r(i) \cdot p(i, j) \quad (5)$$

$$w^+(i, j) = r(i) \cdot p^+(i, j), w^-(i, j) = r(i) \cdot p^-(i, j) \quad (6)$$

olsun. Öyleyse, i sinirinin *ateşleme oranı*

$$r(i) = \sum_{j=1}^n [w^+(i, j) + w^-(i, j)] \quad (7)$$

dir. Burada "w"ler, özellikle uyarıcı ve bastırıcı darbe yayılımının sıklıklarını veriyor olsalar da, bağlantısal modellerdeki *sinaptik* ağırlıklarınkine benzer bir rol oynarlar. Bir i sinirine ağ dışından gelen uyarıcı ve bastırıcı imgeler, sırasıyla $\Lambda(i)$ ve $\lambda(i)$ sıklıklarına sahip *Poisson süreçlerini* oluştururlar. $\Lambda(i)$ ve $\lambda(i)$, sıklıkları ifade etseler de, benzer şekilde, bağlantısal modellerdeki girdi sinirlerine uygulanan girdilere karşılık gelmektedirler. Bu, rasgele yapıdaki bir *tekrarlayan ağ* modelidir, yani *geri besleme döngüsü* içerebilir [6, 7, 8, 10].

Bu modele ilişkin hesaplamalar, ağ durumunun olasılık dağılımına, $p(k, t) = P[k(t) = k]$, veya i sinirinin uyarılmasının olasılığına, $q_i(t) = P[k_i(t) > 0]$, dayanır. $\{k(t) : t \geq 0\}$, bir sürekli zaman Markov zinciri olduğu için, bir *Chapman-Kolmogorov eşitlikleri*' nin sonsuz sistemine karşılık gelir. Yani, a, b, x durumlar, ve $t_b > t_x > t_a \geq 0$ herhangi zamanlar olmak üzere, herhangi bir t_a anındaki a durumundan herhangi bir t_b anındaki b durumuna geçiş olasılığı :

$$P_{a,b}(t_a, t_b) = \sum_x P_{a,x}(t_a, t_x) \cdot P_{x,b}(t_x, t_b) \quad (8)$$

şeklinde ifade edilebilir.

Modele ilişkin durağan durum olasılık dağılımı, ağın çıktısı olarak alınacaktır :

$$q_i = \lim_{t \rightarrow \infty} P[k_i(t) > 0], i = 1, \dots, n \quad (9)$$

Yukarıda (9)'deki eşitlik

$$q_i = \frac{\lambda^+(i)}{r(i) + \lambda^-(i)} \quad (10)$$

ifadesine denktir. Burada, $\lambda^+(i)$, ve $\lambda^-(i)$ $i = 1, \dots, n$ için, aşağıdaki doğrusal olmayan eş-anlı eşitlikleri sağlar :

$$\lambda^+(i) = \sum_j q_j w^+(j, i) + \Lambda(i), \lambda^-(i) = \sum_j q_j w^-(j, i) + \lambda(i) \quad (11)$$

Yani, bir sinirin uyarılmasının durağan durum olasılığı, o sinire gelen uyarıcı imgelerin sıklıklarının toplamının, o sinire gelen bastırıcı imgelerin sıklıklarıyla o sinirin ateşleme sıklığının toplamına oranına eşittir.

Eğer, bir i siniri için,

$$\frac{\lambda^+(i)}{r(i) + \lambda^-(i)} \geq 1 \quad (12)$$

ise, o sinir *doymuş* olarak adlandırılır. Bu durumda, i siniri, durağan durumda sürekli ateşleyecektir. Doymuş bir sinir için durağan durum olasılığı $q_i = 1$ olarak kabul edilir. Pek çok uygulamada, bazı sinirlerin doymuş olması istenir.

Bu model, *çarpımlarla ifade edilebilen* bir çözüme sahiptir [13, 14]. Yani, modelin durumunun durağan olasılık dağılımı, içerdiği tüm sinirlerin tek tek durumlarının (veya gizil-güçlerinin) olasılıklarının çarpımına eşittir. Bu, ağın durumu için basit ifadelerle yol açar. Çarpımlarla ifade edilebilen çözümlerin sadece artı imgelerin dolaştığı belirli ağlar, yani, bilgisayar ve iletişim sistemlerinin modellenmesinde ve yöneylem araştırmasında kullanılan kuyruklaama ağları, için varolduğu, bu model geliştirilmeden önce bilinmekteydi [13, 14].

Eğer, (10) ve (11) eşitliklerine her bir $q_i < 1$ olacak şekilde negatif olmayan bir çözüm varsa, RYSA'nın durağan olasılık dağılımı :

$$p(k) = \prod_{i=1}^n [1 - q_i] q_i^{k_i} \quad (13)$$

şeklinde ifade edilebilecektir. Böylece, bir sinirin durağan durumda sahip olacağı ortalama gizil-güç :

$$k_i = \frac{q_i}{1 - q_i} \quad (14)$$

olacaktır. Bir ağ ileri beslemeliyse, (10) ve (11)'in çözümleri vardır ve tektir.

Bu modelde, bilgi, darbelerin gezindiği frekans ile taşınır. Gelen uyarıcı ve bastırıcı darbe dizilerinin sıklıklarını bir şiddete dönüştürdüğü için, sırasıyla her sinir, doğrusal olmayan bir *frekans çözücüsü* olarak davranır. Bir i siniri, herhangi bir j sinirine $q_i(t) \cdot r(i) \cdot p^+(i, j)$ sıklığında uyarıcı, ve $q_i(t) \cdot r(i) \cdot p^-(i, j)$ sıklığında bastırıcı imgeler gönderdiği için (bkz. Şekil 4), bu modeldeki her sinir, aynı zamanda bir *frekans biçimleyicisidir*.

Her bir sinire pozitif veya negatif imgelerin ulaşma sıklığını tanımlayan, ağına imge akış eşitlikleri doğrusal değildir. Bu yüzden, *ileri besleme* ağları haricinde varlıklarının ve tekliklerinin gösterilebilmesi kolay değildir. [7]'de modelin kararlılığıyla ilgili koşullara açıklık getirilmiş, imge akış eşitliklerine çözüm varolması halinde bunun tek olduğu gösterilmiştir. Eğer, $q_i = q_j = q, \forall i, j$, yani, tüm q_i 'ler birbirine eşitse, o RYSA *dengeli* olarak adlandırılır. Eğer,

$$r(i) + \lambda(i) > \Lambda(i) + \sum_j q_j w^+(j, i) \quad (15)$$

ise, o RYSA *sönümlü* olarak adlandırılır. Ağ *sönümlü* olduğunda, q_i olacak şekilde, (10) ve (11)'e bir çözüm daima vardır, ve bu tektir. Bir RYSA'da geri besleme kullanılması durumunda, her seferinde, imge akış eşitliklerinin bir çözümü olup olmadığı belirlenmelidir. Bir çözüm bulunabilmesi halinde, bunun tek olduğu rahatlıkla söylenebilir. Geri beslemeli bir RYSA, dengelilik ya da *sönümlülük* koşullarını sağladığında, bu aynı zamanda, onun imge akış eşitliklerinin bir çözümü olduğu, dolayısıyla kararlılığını garantilediği anlamına da gelecektir.

3.1 RYSA öğrenme algoritması

RYSA modelinin geliştirilmesinin ardından [6, 7, 8, 9, 10], RYSA için *geri-sürüklenme* tipinde geliştirilen öğrenme

algoritması [11], ağa bir girdi/çıkıtkümesi uygulandığında, *karesel hata fonksiyonunun kademeli azalması yöntemi* kullanır. Algoritma, K girdi-çıkıtkümesi (I, Y) 'yi öğrenmek için W ağ parametrelerini seçer. Burada, ardışık girdilerin kümesi $1(i_1, \dots, i_K)$, ve $1_k = \{\Lambda_k, \lambda_k\}$, her bir sinire ağımdışından giren uyarıcı, ve bastırıcı imge akış sıklıklarıdır :

$$\Lambda_k = (\Lambda_k(1), \dots, \Lambda_k(n)), \lambda_k = (\lambda_k(1), \dots, \lambda_k(n)) \quad (16)$$

Ardışık olarak istenen çıkıtkümlerin vektörü, $Y = \{y_1, \dots, y_K\}$ 'dir. Her bir $y_k = \{y_{1k}, \dots, y_{nk}\}$, $y_{ik} \in [0, 1]$ olmak üzere, her bir sinir için istenen çıkıtküme değerlerine karşılık gelmektedir. Ağ, parametrelerini bir E_k maliyet fonksiyonunun en düşük değerleri üretmesini sağlayacak şekilde ayarlar :

$$E_k = \frac{1}{2} \sum_{i=1}^n a_i (q_i - y_{ik})^2, a_i \geq 0 \quad (17)$$

Bu ağda, tüm sinirler, çıkıtküme siniri olabilecek şekilde genelleştirilmiştir. Böylece, eğer bir j sinirinin ağ çıkıtkümesinden çıkartılması gerekiyorsa, hata fonksiyonunda $a_j = 0$ yapmak yeterlidir. Tüm sinirlerin durağan durum çıkıtküme sıklıkları :

$$q_i = \frac{\lambda^+(i)}{r(i) + \lambda^-(i)} = \frac{N(i)}{D(i)} \quad (18)$$

eşitlikleriyle verilir.

Her iki n x n ağırlık matrisleri de, $W_k^+ \{w_k^+(i, j)\}$ ve $W_k^- \{w_k^-(i, j)\}$, her girdi uygulandıktan sonra, her $1_k = \{\Lambda_k, \lambda_k\}$ girdisi için yeni bir W_k^+ ve W_k^- ağırlık matrisi değerleri hesaplayarak ayarlanmalıdır. Ağırlık matrisleri, bir sıklık ile bir olasılığın çarpımından oluştukları için, sadece matristeki tüm değerlerin pozitif olduğu sonuçlar geçerlidir.

$w(u, v)$, $w(u, v) \equiv w^+(u, v)$ ya da $w(u, v) \equiv w^-(u, v)$ olabilen, her hangi bir ağırlık terimine karşılık gelsin. Ağırlıklar *kademeli azalma yöntemine* göre güncellenmiştir :

$$w^{\text{yeni}}(u, v) = w^{\text{eski}}(u, v) - \eta \frac{\partial E}{\partial w^+(u, v)} \quad (19)$$

Maliyet fonksiyonunun kısmi türevi, güncelleme fark eşitliğini elde etmek için, hesaplanıp yerine konabilir :

$$w_k(u, v) = w_{k-1}(u, v) - \eta \sum_{i=1}^n a_i (q_i - y_{ik}) \left[\frac{\partial q_i}{\partial w(u, v)} \right]_k \quad (20)$$

Burada, $\eta > 0$, eğitimin her yinelenmesinde sabit olan *öğrenme parametresidir*.

a) q_{ik} , 1_k girdisini ve $w_k(u, v) = w_{k-1}(u, v)$ 'yi (18) ve (11)'de yerine koyarak hesaplanabilir,

$$b) \left[\frac{\partial q_i}{\partial w(u, v)} \right]_k \text{ 'nin değeri, } q_i = q_{ik}, w_k^+(u, v) = w_{k-1}^+(u, v), \text{ ve}$$

$w_k^-(u, v) = w_{k-1}^-(u, v)$ değerlerini kullanarak hesaplanabilir.

$\left[\frac{\partial q_i}{\partial w(u, v)} \right]_k$ 'yi hesaplamak için, (11) ve (18)'den aşağıdaki eşitlik türetilmiştir :

$$\left[\frac{\partial q_i}{\partial w(u, v)} \right] = \frac{1}{D(i)} \left[\sum_j \frac{\partial q_i}{\partial w(u, v)} [w^+(j, i) - w^-(j, i)q_i] - 1[u = i]q_i + 1[w(u, v) \equiv w^+(u, v)]q_u - 1[w(u, v) \equiv w^-(u, v)]q_u q_i \right] \quad (21)$$

burada

$$1[x] = \begin{cases} 1, & \text{eğer } x \text{ doğrudur} \\ 0, & \text{aksi takdirde} \end{cases}$$

dir.

$q = (q_1, \dots, q_n)$ olsun. $i, j = 1, \dots, n$ olmak üzere, n x n

$$W = \frac{w^+(j, i) - w^-(j, i)q_i}{D(i)} \quad (22)$$

matrisi tanımlansın. Bu durumda, vektör eşitlikleri,

$$\frac{\partial q_i}{\partial w(u, v)} = \frac{\partial q_i}{\partial w(u, v)} W + \gamma(u, v)q_u \quad (23)$$

$$\frac{\partial q_i}{\partial w^+(u, v)} = \frac{\partial q_i}{\partial w^+(u, v)} W + \gamma^+(u, v)q_u,$$

$$\frac{\partial q_i}{\partial w^-(u, v)} = \frac{\partial q_i}{\partial w^-(u, v)} W + \gamma^-(u, v)q_u \quad (24)$$

şeklinde yazılabilir. Burada, n-elemanlı $\gamma(u, v) = [\gamma_1(u, v), \dots, \gamma_n(u, v)] = \gamma^+(u, v) + \gamma^-(u, v)$ ve $\gamma^+(u, v) = [\gamma_1^+(u, v), \dots, \gamma_n^+(u, v)]$, $\gamma^-(u, v) = [\gamma_1^-(u, v), \dots, \gamma_n^-(u, v)]$ vektörlerinin elemanları :

$$\begin{cases} \frac{-1}{D(i)}, & \text{eğer } u=i, v \neq i \text{ ise} \end{cases}$$

$$\gamma^+(u, v) = \begin{cases} \frac{1}{D(i)}, & \text{eğer } u \neq i, v=i \text{ ise} \end{cases} \quad (25)$$

$$\begin{cases} 0, & (u, v) \text{ 'nin tüm diğer değerleri için} \\ \frac{-(1+q_i)}{D(i)}, & \text{eğer } u=i, v=i \text{ ise} \end{cases}$$

$$\gamma^-(u, v) = \begin{cases} \frac{-1}{D(i)}, & \text{eğer } u=i, v \neq i \text{ ise} \end{cases} \quad (26)$$

$$\begin{cases} \frac{-q_i}{D(i)}, & \text{eğer } u \neq i, v=i \text{ ise} \end{cases}$$

$$\begin{cases} 0, & (u, v) \text{ 'nin tüm diğer değerleri için} \end{cases}$$

Bu durumda

$$\frac{\partial q_i}{\partial w^+(u, v)} = \gamma^+(u, v)q_u [I - W]^{-1}, \text{ ve}$$

$$\frac{\partial q_i}{\partial w^-(u, v)} = \gamma^-(u, v)q_u [I - W]^{-1} \quad (27)$$

olduğuna dikkat ediniz. Burada, I, birim matristir. Böylece, algoritmadaki asıl hesaplama işi, $[I - W]^{-1}$ 'nin elde edilmesidir. Bunun hesaplama karmaşıklığı, $O(n^3)$, veya değişkenlerin ardışık yaklaşımlarına dayalı bir m-basamaklı *gevşeme yöntemi* kullanılırsa, $O(mn^2)$ 'dir.

Yukarıdakilerden, ağ için tüm öğrenme algoritması verilebilir.

Önce, uygun bir şekilde, W_0^+ , ve W_0^- matrislerine başlangıç değerleri atanır. Daha iyi bir yöntem belirlenemiyorsa, bu ilk an değerleri rasgele olarak atanabilir. Bir *öğrenme hızı* η değeri seçilir, ve ardından $k=1$ 'den başlayarak, k'nin her ardışık değeri için, aşağıdaki basamaklar uygulanır:

(Adım.1) Girdi değerlerini, $1_k = (\Lambda_k, \lambda_k)$ 'e ayarlanır.

(Adım.2) (18) ve (11)'de verilen doğrusal olmayan eşitlikler sistemini, *Gauss-Seidel* gibi yinelemeli bir yöntem kullanarak, çözülür.

(Adım.3) (2)'nin sonuçlarını kullanarak (27) çözülür.

(Adım.4) Eşitlik (20)'u ve (Adım.2) ile (Adım.3)'ün sonuçlarını kullanarak W_0^+ , ve W_0^- matrislerini güncelle. Eksi değer olmama koşulunu sağlayan en iyi matrisler (karesel maliyet fonksiyonunun kademeli azalması cinsinden) arandığı için, algoritmanın herhangi bir k basamağında, eğer yineleme, bir terim için '-' değer verirse, iki seçenek vardır :

a) Terimi O'a eşitle, ve bu k basamağında bu terim için yinelemeye son ver. Bir sonraki k+1 basamağında, k'nin o anki 0 değerinden başlayarak yinelemeleri yeniden yapılır,

b) Terimin bir önceki değerine dönülür ve daha küçük bir öğrenme hızı η değeri için yineleme yeniden yapılır.

Bu genel şema, *ileri-beslemeli ağlara* özelleştirilebilir. $[I - W]^{-1}$ matrisinin, üçgenel olacağına dikkat edilmelidir. Bu durum, her bir kademe yinelemesi için, hesaplama karmaşıklığının $O(n^3)$ yerine $O(n^2)$ olmasına yol açar. Bunun ötesinde, ileri-beslemeli bir ağda, (18) ve (11)'deki eşitlikler, $j < i$ için, q_i , sadece q_j 'e bağımlı olacak şekilde basitleştirilir. Bu, (18) ve (11)'i çözmek için gereken işlem gücünü azaltır.

3.2 Rasgele Yapay Sinir Ağı Mimarisi

Shih and Moh [36] ve Moh and Shih [26], morfolojik işlemleri de içeren genel amaçlı görüntü işlemleri için çok katmanlı bir yapay sinir ağı mimarisi önerdiler. Pessoa and Maragos [30] ve Pessoa [29], morfolojik sinir ağını, her düğümün bir morfolojik/rank işlemi olduğu, yani, aşınma, genişleme ve medyan işlemlerinin eşzamanlı ele alındığı, morfolojik/rank yapay sinir ağına genişlettiler, ve Pessoa [29] ve Pessoa and Maragos [31, 32]'de uygulama örneği verdiler. Uygun eğitim algoritmalarının geliştirilmesi (bkz. [1, 16]) ve yakınsama üzerinde hala çalışılması gerekse de morfolojik yapay sinir ağı kavramı, çoğu uygulamalar için çok kullanışlıdır [5, 17, 19, 20, 28, 33, 34, 38, 39, 40, 41, 42, 43].

Morfolojik işlemler, *en-küçük* ve *en-büyük* fonksiyonlarının kombinasyonlarıdır. İki-değerlikli morfolojik işlemlerde bu aynı zamanda *ve* ve *veya* fonksiyonlarına denktir. Dolayısıyla morfolojik işlemler için tasarlanmış yapay sinir ağları, *en-küçük* ve *en-büyük* fonksiyonlarını, iki-değerlikli girdiler içinse *ve* ve *veya* fonksiyonlarının, kombinasyonlarını gerçekleyen mimarilerdir.

Koza and Rice [22], Beiu and Moore [2], Cotona and Vassiliadis [3], ve Shimada and Saito [37], Boolean fonksiyonlarını, Morales and Ko [27] ve Lee *et al.* [24] morfolojik işlemleri (ikincisi açılış ve kapanış işlemlerini), Hohil [18] ise *çok girdili* ve *veya* fonksiyonlarını yapay sinir ağıyla gerçekleştirmişlerdir. Kim and Park [21], Boolean fonksiyonları için çok katmanlı yapay sinir ağı kullanmış ve öğrenme yöntemi geliştirmişlerdir. Deolalikar [4] ise iki katmanlı yapay sinir ağıyla gerçekleştirmiş, iki-değerlikli ağırlıklar ve sıfır eşik değerleri kullanmıştır.

Bu çalışmada kullanılan morfolojik işlemleri gerçeklemek için geliştirilen RYSA mimarileri *n-girdili* 1-çıkıtlı, 2 katmanlı ileri beslemelidir. Başlıca işlevleri, *n-girdili* ve (Şekil 5.a) ve *n-girdili veya* (Şekil 5.b) işlemlerini gerçekleştirmektedir. Öğrenme aşamasında *n-girdili* ve *veya* işlemlerinin doğruluk tabloları bu RYSA mimarilerine öğretilmekte, ezberletilmektedir. Ezberleme sayesinde bu RYSA mimarileri, %100 performansla, hatasız sonuç üretmektedir. *n-girdili* ve işlemlerinin ardışık uygulanmasıyla aşınma (Şekil 6.a), *n-girdili veya* işlemlerinin ardışık uygulanmasıyla genişleme (Şekil 6.b) işlemleri elde edilmektedir. Önce aşınma, ardından genişleme uygulanmasıyla açılış (Şekil 7) işlemleri, ters sırayla önce genişleme, ardından aşınma uygulanmasıyla kapanış (Şekil 8) işlemleri elde edilmektedir. $m > 2$, $m \times m$ noktada tanımlı yapıtaşı elemanları kullanılmıştır, bu noktaların bir kısmı yapıtaşı elemanınca içerilmekte, kalan kısmı içerilmemektedir. Yapıtışı elemanın tanımlı olduğu $m \times m$ nokta satırlarında yapıtaşı elemanının içerdiği noktalar, *g-girdili*, $2 \leq g \leq m$ (her satırda yapıtaşı elemanı farklı sayıda nokta içerebileceği için), *ve* ve *veya* işlemlerinin girdileridir. Satırlarda yapılan işlemlerin çıktılarının *m-girdili* ve işleminin çıktısı aşınma, *m-girdili veya* işleminin çıktısıysa genişlemedir.

Bu çalışmayı gerçekleştirmek için geliştirilen uygulama yazılımı paketi, QT (GNU) platformunda C++ programlama diliyle geliştirilmiştir. Dr. Erol Gelenbe'nin doktora öğrencisi Dr. Christopher Eric Cramer'in (Duke Uni., ECE Dept., USA) C++ ile

geliştirdiği ve diğer doktora öğrencilerinin de kullandığı RYSA kodu, geliştirilen yazılıma uyarlanmıştır.

4. Yöntem ve Uygulama Bağlamında Değerlendirmeler

Bu çalışmada geliştirilen yöntem matematiksel morfolojide yaygın olarak kullanılan başlıca yapıtaşı elemanlarıyla sınırlanmıştır. Kullanılan yapıtaşı elemanları, 1) 4-komşuluklu (paralel kenar, elmas, baklava dilimi) (Şekil 1.a), 2) 8-komşuluklu (kare) (Şekil 1.b), ve 3) daire'dir (Şekil 1.c). Geliştirilen yöntem, [23]'te Boolean (Boole) rasgele kümelerin modellenmesinde kullanılmıştır.

Morfolojik işlemler, bütün görüntü baştan sonra taranarak uygulanmaktadır. Dolayısıyla, gerekmeyen noktalarda işlemler uygulanmaksızın geçilebilmesi sağlanmışsa da, hesaplama maliyeti yüksektir. Bir görüntüye bir işlemin bir kez uygulanmasının *hesaplama karmaşası* yaklaşık olarak $O(m.n).k_{max}$ düzeyindedir, burada m ve n , gözlem penceresinin, yani incelenen görüntünün, boyutlarıdır, k_{max} ise görüntünün içerdiği en büyük şekil büyüklüğü katsayısıdır.

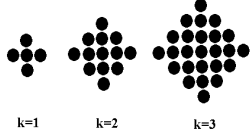
5. SONUÇ

Şekil inceleme çalışmaları, Matematiksel Morfoloji (Biçimbilim) gibi şekillerin incelenmesinde kullanılan işlemler ve yöntemler sağlayıcı, ve RYSA gibi genel amaçlı çözüm yöntemleri sağlayıcı kuramsal çalışmalarla desteklenmektedir. Bu çalışmada hem Matematiksel Morfoloji, ve RYSA'nın ilk kez bir uygulamada bir arada kullanılması önerilmiş ve kullanılmış, hem çok girdili ve *veya* işlemleri RYSA ile ilk kez gerçekleştirilmiş, hem de morfolojik işlemler RYSA ile ilk kez gerçekleştirilmiştir. Geliştirilen yöntem, mekansal istatistik, örüntü tanıma, ve görüntü analizi uygulamalarına yönelik, ilk kez önerilen ve gerçekleştirilen yeni bir gereçtir.

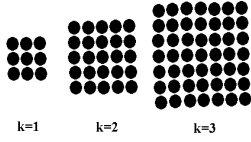
Geliştirilen yöntemi oluşturan temel atomik işlemler, RYSA mimarileri ile gerçekleştirilen çok girdili ve *veya* işlemleridir. Bunlar için geliştirilen RYSA mimarilerinin, aynı amaçlı farklı YSA modeli mimarileriyle, ve farklı yöntemlerle karşılaştırılacak olursa, performansının diğerlerininkinin altında kalmayacağı beklenmektedir. Böyle bir karşılaştırmada temel ölçüt olarak yukarıda verilen hesaplama karmaşası, $O(\cdot)$, kullanılabilir. Aşınma ve genişleme işlemleri ile bunların ardışık uygulanmasından elde edilen açılış ve kapanış işlemleri için geliştirilen RYSA mimarilerinin performansının ise, morfolojik YSA'larınınkinin altında kalmayacağı beklenmektedir; ayrıca, *n-girdili* ve *veya* işlemlerinin doğruluk tablolarının ezberletilmesi sayesinde, %100 performansla, hatasız sonuç üretmeleri, geliştirilen RYSA mimarilerine üstünlük sağlamaktadır, oysa diğerlerinin belli bir yüzdeyle hatalı sonuç üretebilmesi söz konusudur.

Geliştirilen yöntem, matematiksel morfolojinin diğer tüm işlemlerini de kapsayan bir gereç haline kolayca genişletilebilir.

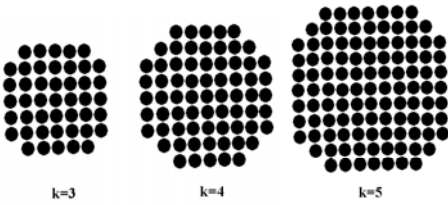
Geliştirilen yöntem, kolayca paralelleştirilebilir, şekil incelemedeki yinelemelerin azaltılması yönünde iyileştirmeler yapılabilmesi de mümkün olabilir. Bu şekilde hesaplama karmaşası, ve dolayısıyla hesaplama maliyeti azaltılabilir. Bu iyileştirmeler, yeni hesaplama kaynakları ihtiyacını ya da anlık hesaplama kaynakları kullanma ihtiyacını beraberinde getirebilir.



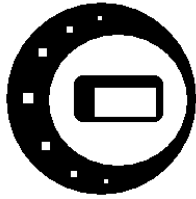
Şekil 1.a. 4-komşuluklu (paralel kenar, elmas, baklava dilimi) yapıtaşı elemanı ailesinin k=1,2,3 üyeleri



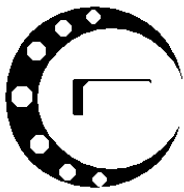
Şekil 1.b 8-komşuluklu (kare) yapıtaşı elemanı ailesinin k=1,2,3 üyeleri



Şekil 1.c. Daire yapıtaşı elemanı ailesinin k=3,4,5 üyeleri (k=1,2 üyeler, kare ile aynı)



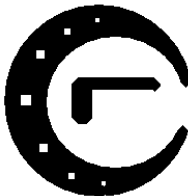
(a)



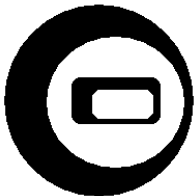
(b)



(c)

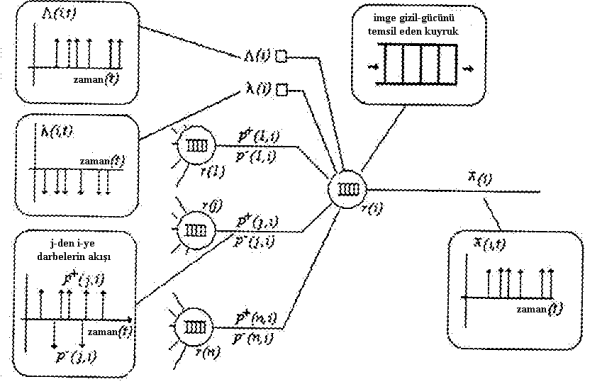


(d)

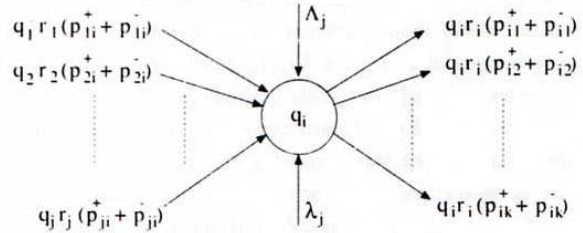


(e)

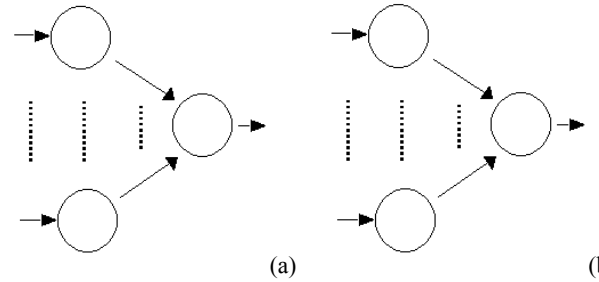
Şekil 2.a. Örnek görüntü F , b. Aşınma işlemi sonucu, $F \ominus B_k$, c. Genişleme işlemi sonucu, $F \oplus B_k$, d. Açılış işlemi sonucu, $F \odot B_k$, e. Kapanış işlemi sonucu, $F \bullet B_k$; (B : paralel kenar tipi, $k=5$)



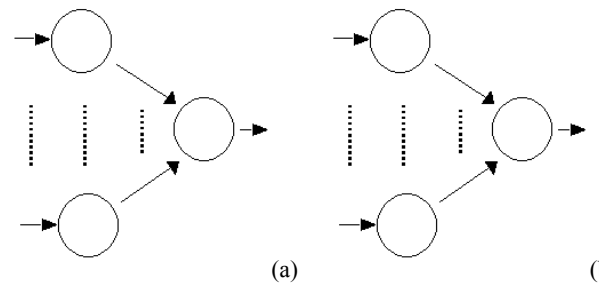
Şekil 3. RYSA modeli, imge grafikleriyle gösterim [12].



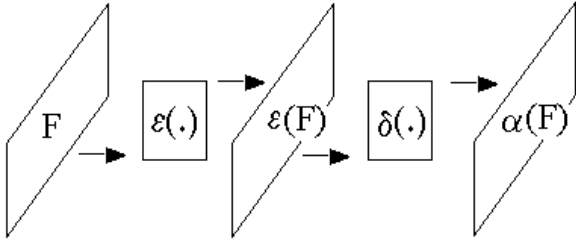
Şekil 4. RYSA modeli, tek sınır gösterimi [12].



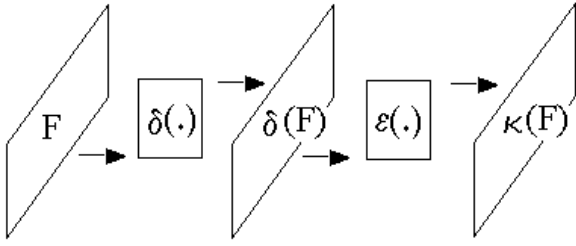
Şekil 5.a. n-girdili ve işleci RYSA mimarisi, b. n-girdili veya işleci RYSA mimarisi; (girdiler: nokta renk değerleri)



Şekil 6.a. Aşınma işleci, $\epsilon(\cdot)$, RYSA mimarisi (girdiler: yapıtaşı elemanı satırlarının n-girdili ve işlemleri çıktısı), b. Genişleme işleci, $\delta(\cdot)$, RYSA mimarisi (girdiler: yapıtaşı elemanı satırlarının n-girdili veya işlemleri çıktısı)



Şekil 7. Açılış tabanlı morfolojik elek (elek büyüklüğü k sabit) RYSA mimarisi



Şekil 8. Kapanış tabanlı morfolojik elek (elek büyüklüğü k sabit) RYSA mimarisi

6. Kaynakça

- [1] Araujo, R. de A., Madeiro, F., Sousa, R.P. de and Pessoa, L.F.C. 2006. Modular morphological neural network training via adaptive genetic algorithm for designing translation invariant operators acoustics. Proc. of IEEE Int. Conf. on Speech and Signal Processing, 2006. ICASSP 2006, vol.2, 14-19 May, II-873- 876.
- [2] Beui, V. and Moore, K.R. 1998. On analog implementation of discrete neural networks. Proc. of the 3rd Int. Symp. on Fuzzy Logic and Intelligent Technologies for Nuclear Science and Industry, Sept. 14-16, Antwerp, Belgium.
- [3] Cotofana, S. and Vassilidis, S. 1998. Periodic symmetric functions, serial addition, and multiplication with neural networks. IEEE Trans. on Neural Networks, vol.9, no.6, Nov., 1118-1128.
- [4] Deolalikar, V. 2000. Mapping Boolean functions with neural networks having binary weights and zero thresholds. (unpublished).
- [5] Gader, P.D., Khabou, M.A. and Koldobsky, A. 2000. Morphological regularization neural networks. Pattern Recognition, vol.33, no.6, 935-944.
- [6] Gelenbe, E. 1989. Random neural networks with negative and positive signals and product form solution. Neural Computation, vol.1, no.4, 502-511.
- [7] Gelenbe, E. 1990. Stability of the random neural network model. Neural Computation, vol.2, no.2, 239-247.
- [8] Gelenbe, E. 1991a. Distributed associative memory and the computation of membership functions. Information Sciences, vol.57-58, 171-180.
- [9] Gelenbe, E. 1991d. Product-form queueing networks with negative and positive customers. J. of Applied Probability, 28, 656-663.
- [10] Gelenbe, E. 1993a. The random neural network model. Elektrik, vol.1, no.1, 27-46.
- [11] Gelenbe, E. 1993b. Learning in the recurrent random neural network. Neural Computation, vol.5, no.1, 154-164.
- [12] Gelenbe, E. and Halıcı, U. 1994. The Random Neural Network. (Preliminary Copy, unpublished, 234).
- [13] Gelenbe, E. and Mitriani, I. 1980. Analysis and Synthesis of Computer Systems. Academic Press, 239, London.
- [14] Gelenbe, E. and Pujolle, G. 1998. Introduction to Networks of Queues. (2nd ed.) Wiley, New York, 244.
- [15] Goutsias, J. and Batman, S. 2000. Morphological methods for biomedical image analysis. Handbook of Medical Imaging, vol.2, Medical Image Processing and Analysis, M. Sonka, J.M. Fitzpatrick (ed.s), SPIE Press, Bellingham, Washington, 175-272.
- [16] Gray, D. and Michel, A.N. 1992. A training algorithm for binary feedforward neural networks. IEEE Trans. on Neural Networks, vol.3, 176-194, March.
- [17] Hocaoglu, A.K. 2000. Choquet integral-based morphological operators with applications to object detection and information fusion. PhD Dissertation, University of Missouri - Columbia, EE Dept., USA, 144.
- [18] Hohil, M.E. 1998. Linear Programming for Minimization of Neural Networks with Binary Inputs. PhD Dissertation, Stevens Institute of Technology, ECE Dept., New Jersey, USA, 111.
- [19] Hu, J. and Deng, W. 2005. A robust morphological associative memory endowed with dendrites. Int. Conf. on Neural Networks and Brain, 2005. ICNN&B'05. vol.1, 13-15 Oct., 147-149.
- [20] Khabou, M.A. and Solari, L.F., 2006. A Morphological Neural Network-Based System for Face Detection and Recognition. Proc. of the IEEE SoutheastCon, 2006, March 31 - April 2, 296-301.
- [21] Kim, J. H. and Park, S.-K. 1995. The geometrical learning of binary neural networks. IEEE Trans. on Neural Networks, vol.6, no.1, Jan., 237-247.
- [22] Koza, J.R. and Rice, J.P. 1992. Genetic generation of both the weights and architecture for a neural network. (unpublished).
- [23] Küçüköncü, T. 2007. Örüntü Tanıma Ve Görüntü Analizinde Rasgele Yapay Sinir Ağı Kullanımıyla Boolean Rasgele Kümelerin Modellenmesi. Doktora Tezi, Ankara Üniversitesi, FBE, 165.
- [24] Lee, K.-H., Morales, A. and Ko, S.-J. 1997. Adaptive basis matrix for the morphological function processing opening and closing. IEEE Trans. Image Processing, vol.6, no.5, May, 769-774.
- [25] Matheron, G. 1975. Random Sets and Integral Geometry. Wiley, 261, New York.
- [26] Moh, J. and Shih, F.Y. 1995. A general purpose model for image operations based on multilayer perceptrons. Pattern Recognition, vol.28, no.7, 1083-1090.
- [27] Morales, A. and Ko, S.-J. 1992. Efficient neural network implementation of morphological operations. SPIE Nonlinear Image Processing III, vol.1658, 276-286.
- [28] Oh, J. 1998. Adaptive fuzzy morphological filter design and its applications. PhD Dissertation, University of Pittsburg, EE Dept., USA, 116.
- [29] Pessoa, L.F.C. 1997. Nonlinear systems and neural networks with hybrid morphological/rank/linear nodes: Optimal design and applications to image processing and pattern recognition. PhD Dissertation, Georgia Institute of Technology, EE Dept., USA, 100.
- [30] Pessoa, L.F.C. and Maragos, P. 1996. Morphological / rank neural networks and their adaptive optimal design for image processing. Proc. of the IEEE Int. Conf. on Acoustics, Speech, & Signal Processing, vol.6, Atlanta, GA, May, 3393-3401.
- [31] Pessoa, L.F.C. and Maragos, P. 1998. MRL-Filters : a general class of nonlinear systems and their optimal design for image processing. IEEE Trans. Image Processing, vol.7, July, 966-978.
- [32] Pessoa, L.F.C. and Maragos, P. 2000. Neural networks with hybrid morphological/rank/linear nodes : a unifying framework

- with applications to handwritten character recognition. *Pattern Recognition*, vol.33, no.6, 945-960.
- [33] Ritter, G.X. and Beaver, T.W. 1999. Morphological perceptrons. *Int. Joint Conf. on Neural Networks*, 1999. IJCNN'99, vol.1, 10-16 July, 605-610.
- [34] Ritter, G.X. and Sussner, P. 1996. An introduction to morphological neural networks. *Proc. of the 13th Int. Conf. on Pattern Recognition*, 1996, vol.4, 25-29 Aug., 709-717.
- [35] Serra, J. 1982. *Image Analysis And Mathematical Morphology*. Academic Press, 610, London.
- [36] Shih, F.Y.-C. and Moh., J. 1992. Implementing morphological operations using programmable neural networks. *Pattern Recognition*, vol.25, no.1, 89-99.
- [37] Shimada, M. and Saito, T. 2002. A GA-based learning algorithm for binary neural networks. *IEICE Trans. Fundamentals*, vol.E85-A, no.11, Nov., 2544-2546.
- [38] Sussner, P. 1998. Morphological perceptron learning. *Proc. of the IEEE Int. Symp. on Intelligent Systems and Semiotics (ISAS)*, 14-17 Sept., 477-482.
- [39] Sussner, P. 2005. New results on binary auto- and heteroassociative morphological memories. *Proc. of IEEE Int. Joint Conf. Neural Networks*, 2005, IJCNN'05, vol.2, 31 July-4 Aug., 1199-1204.
- [40] Villaverde, I., Grana, M. and D'Anjou, A. 2006. Morphological Neural Networks for Localization and Mapping. *Proc. of 2006 IEEE Int. Conf. on Computational Intelligence for Measurement Systems and Applications*, July, 9-14.
- [41] Won, Y. 1995. Nonlinear correlation filter and morphology neural networks for image pattern and automatic target recognition. PhD Dissertation, Cornell University, EE Dept., USA, 1995, 235.
- [42] Yun, Z., Ling, Z. and Yimin, Y. 2004. Using multi-layer morphological neural network for color images retrieval. *Fifth World Congress on Intelligent Control and Automation*, 2004, WCICA 2004, vol.5, 15-19 June, 4117-4119.
- [43] Zhang, L., Zhang, Y. and Yang, Y.-M. 2003. Color images restoration with multi-layer morphological (MLM) neural network. *Int. Conf. on Machine Learning and Cybernetics*, 2003, vol.5, 2-5 Nov., 2831-2834.