

شبکه عصبی فازی انعطاف پذیر برای شناسایی اثر انگشت

کارشناس حمید کوهساری

مقدمه:

شناسایی عبارت است از هنر تشخیص یک شیء به کلاسی از الگوها که به آن تعلق دارد که به زبان ریاضی یعنی تشخیص الگوهای ورودی به یکی از طبقاتی که فضای چند بعدی اقلیدسی برای تصمیم گیری به تعداد متناهی از آن ها تقسیم شده است. [1]

با توجه به این که هر شخص الگوی اثر انگشت منحصر به فرد خود را داراست و کثرت این الگوها به تعداد افراد بشر، پیچیدگی و انعطاف پذیری بی بدیل این الگو آشکار می گردد (تغییرات و دگرگونی های مختلف حتی تصویر یک انگشت بسته به ابزار اخذ تصویر، میزان آلودگی و رطوبت انگشت و حتی میزان فشار وارده بر صفحه اخذ تصویر اثر و نحوه قرار دادن آن روی این صفحه، تصاویر مختلف و متفاوتی از آن انگشت بدست می دهد). طبق موارد ذکر شده سیستم تشخیص الگو علاوه بر جبران چرخش، دگرگونی و تغییر اندازه این الگو، بایستی قابلیت تصمیم گیری در مورد جابه جایی، حذف و اضافه شدن مشخصه های الگو را نیز دارا باشد. بنابراین ما در اینجا با تصاویر و الگوهای کاملاً پویا و غیر قطعی سروکار داریم و لذا مسئله اثر انگشت هنری است که وابستگی زیادی به تجربه و معرفت افراد خبره و مهارت های تجربی دارد.

شبکه عصبی (Neural Network) NN یک ساختار عظیم موازی، متشکل از تعداد زیادی عناصر پردازشگر است که از طریق وزنها با یکدیگر در ارتباطند [1]. شبکه عصبی دارای قابلیت پردازش موازی، توانایی یادگیری و اخذ تصمیم می باشد که به صورت مدل پردازش موازی توسعه یافته PDP (Parallel Distributed Processing) بیان می شود. [2][3]

شبکه های عصبی NN در مسائل تشخیص الگو بویژه در سیستم های مستقل از انتقال و تغییر اندازه الگو مورد بررسی قرار گرفته و کاربرد دارند؛ مانند کارهای Lisoba و Fukumi برای تشخیص الگوهای مستقل از چرخش و دگرگونی [4]، اما در شبکه های اینچینی تعداد وزن ها با مرتبه ی شبکه بالاتر رفته و ساختار شبکه پیچیده تر می شود.

از طرف دیگر عقیده بر این است که یادگیری مغز انسان نه تنها از ادراک دقیق بوده، بلکه از تصاویر کلی و پیچیده نامفهوم نیز تاثیر می پذیرد لذا مسئله گروه بندی و تشخیص با عدم اطمینان مسئله مهمی در تشخیص الگو بوده و بنابراین تئوری شبکه فازی خود را به عنوان یک اهمیت ارزشمند در مسائل تشخیص الگو مطرح نموده است. [5],[6] ترکیب مشخصه های سیستم های فازی (توانایی پردازش اطلاعات فازی با استفاده از الگوریتم های فازی) و خصوصیات شبکه های عصبی (قابلیت یادگیری و ساختار موازی سرعت بالا) هدف ما برای تشکیل یک شبکه عصبی فازی با قابلیت یادگیری از محیط است. یکی از روش های نوین ترکیب این دو سیستم استفاده از نرون های فازی در

شبکه های عصبی است که بسته به قابلیت ها و انعطاف پذیری فوق العاده ی عملگرهای فازی کاربرد آن در حال توسعه است. از جمله این فعالیت ها معرفی یک مدل نرون فازی ساده توسط Yamakava و استفاده از آن در یک شبکه عصبی برای تشخیص کاراکترها منتهی بدون ذکر الگوریتم یادگیری این شبکه بوده است.[7]

Cai و Kwan نیز یک شبکه عصبی - فازی متشکل از نرون های فازی برای تشخیص حروف و اعداد معرفی نموده اند.[8] این شبکه با وجود انعطاف پذیری خوبی که دارد منتهی کار خوشه بندی (Clustering) را به جای کلاس بندی (Classifying) دقیق با استفاده از الگوریتم یادگیری بدون ناظر به انجام می رساند.

نیخیل (Nikhil) و گوتم (Gautam) در سال 1999 طی بررسی مجدد این شبکه برای بهینه سازی عملکرد آن با استفاده از (Soft Computing) تعریف برخی از نرون های فازی را تغییر داده و یک رابطه برای تعیین پارامترهای شبکه با استفاده از بردارهای برچسب کلاس (Class label vectors) ذکر کرده اند.[9]

ما در این مقاله از تعدادی از نرون های فازی معرفی شده در مرجع [8] استفاده کرده و قسمتی از ساختار شبکه ی معرفی شده توسط Cai و Kwan را در شبکه پیشنهادی خود به کار برده ایم، منتهی با تعریف نرون های جدید و افزودن یک لایه مکمل ساختار شبکه را تا حد زیادی بهبود داده ایم به گونه ای که شبکه کاملاً به یک شبکه ی کلاسه بندی کننده تبدیل شده است و دقت تشخیص شبکه برای کاراکترهای حرفی و عددی نسبت به شبکه ابداعی Cai و Kwan افزایش قابل ملاحظه ای یافته و به 100٪ تشخیص درست رسیده است.

در بخش اول ما پس از تعریف نرون فازی، به معرفی نرون های فازی مورد استفاده در شبکه می پردازیم و در بخش های بعدی با ذکر ساختار ابتدایی شبکه ی Cai و Kwan که متشکل از 4 لایه رو به جلو است، به بررسی مشکلات و کاستی های شبکه پرداخته و راه بهبود آن را با افزودن یک طبقه ی پیشنهادی به شبکه قبل و طرح شبکه جدید 5 لایه ی روبه جلوی کلاسه بندی کننده همراه با تعریف نرون های جدید بیان می داریم.

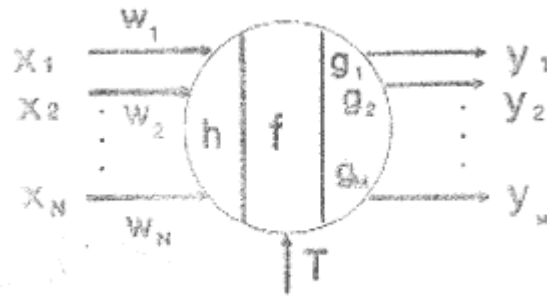
درک ساختار و عملکرد این FNN ساده و قابل فهم و سرعت یادگیری و شناسایی آن زیاد است و طبقات سوم، چهارم و پنجم آن طی فرآیند یادگیری سازماندهی می شوند. در قسمت آخر به بررسی عملکرد و ذکر نتایج بدست آمده از آن در هنگام اعمال الگوهای اثرانگشت می پردازیم.

نرون های فازی (Fuzzy Neurons, FNs)

الف - تعریف نرون فازی:

یک نرون فازی N ورودی وزن دار $(w_i, x_i, i=1 \text{ to } N)$ و M خروجی $(j=1 \text{ to } M)$ را داراست تمام ورودی ها و وزن ها مقادیر حقیقی بوده و خروجی ها نیز مقادیر حقیقی مثبتی در بازه $[0,1]$ می باشند، که در حقیقت بیانگر یک مقدار عضویت در مفهوم فازی هستند. یعنی خروجی ها بینگر این واقعیتند که یک الگوی ورودی با آرایه $\{x_1, x_2, \dots, x_N\}$ تا چه حد به یک مجموعه فازی در نظر گرفته شده تعلق دارد.

شکل (1) نرون فازی را که در این تحقیق مورد استفاده قرار گرفته است نشان داده و روابط آن نیز در ذیل آن بیان شده است.[13]



شکل ۱- نرون فازی

روابطی که بر عملکرد این نرون حاکم است عبارتند از:

a) تابع اجتماع (Aggregation Function) در اینجا z ورودی خالص (Net Input) نرون فازی می باشد.

$$(1) z = h[w_1x_1, w_2x_2, \dots, w_Nx_N]$$

b) تابع فعالیت:

$$(2) s = f[z - T]$$

c) تابع خروجی (Output Function) شبکه FN است.

$$(3) j = 1 \text{ to } M \quad \text{for} \quad y_j = g_j[s]$$

که توابع عضویت الگوهای ورودی به شکل آرایه $\{x_1, x_2, \dots, x_N\}$ را در تمام M مجموعه فازی بیان خواهد کرد. در نتیجه نرون های فازی می توانند اطلاعات فازی را توضیح داده و پردازش نمایند.

در حالت کلی، وزن ها، آستانه فعالیت و توابع خروجی که بیانگر روابط داخلی نرون های فازی و تعامل آن ها با یکدیگر می باشند، می توانند در طی فرایند یادگیری تنظیم شوند بنابراین نرون های فازی تطبیق پذیرند

(Adaptive) و یک شبکه عصبی فازی تشکیل شده از نرون های فازی، قابلیت آموزش و یادگیری از محیط را با الگوهای مختلف داراست.

توابع اجتماع و فعالیت از خصوصیات ذاتی یک نرون فازی می باشند و اگر توابع متفاوتی از $h[]$ و $f[]$ در نرون ها استفاده شوند، این نرون ها خصوصیات و ویژگی های متفاوتی خواهند داشت و مشخصات آن ها تغییر خواهد کرد

لذا انواع زیادی از نرون های فازی را می توان با انتخاب $h[]$ و $f[]$ تعریف نمود.

در ابتدا پنج نوع نرون فازی مورد استفاده در شبکه را به صورت زیر تعریف می کنیم.

1- نرون فازی ورودی (Input-FN)

هنگامی که از یک FN در لایه ورودی یک FNN به گونه ای استفاده شود که داشته باشیم

$$(4) z = x$$

2- نرون فازی بیشینه (Maximum-FN (Max-FN))

هنگامی که تابع به کار رفته به عنوان تابع اجتماع یک نرون فازی، تابع بیشینه یاب باشد

$$(5) Z = \max_{i=1}^N (w_i x_i)$$

3- نرون فازی کمینه (Minimum-FN (Min-FN))

هنگامی که تابع اجتماع نرون فازی به صورت کمینه یاب عمل کند .

$$(6) Z = \min_{i=1}^N (w_i x_i)$$

4- نرون فازی رقابتی (Competitive-FN (Comp-FN))

در صورتی که FN دارای یک آستانه متغیر T و فقط یک خروجی به فرم زیر داشته باشد .

$$(7) y = g[s - T] = \begin{cases} 0 & \text{if } s < T \\ 1 & \text{if } s \geq T \end{cases}$$

جایی که s وضعیت FN بوده، تابع $t[\]$ یک تابع آستانه می باشد و c_k ($k = 1 \text{ to } k$) به صورت

تابع $T = t[c_1, c_2, \dots, c_k]$ مبدل متغیرهای مقایسه ای FN می باشد.

5- نرون فازی جمع کننده (SUM-FN)

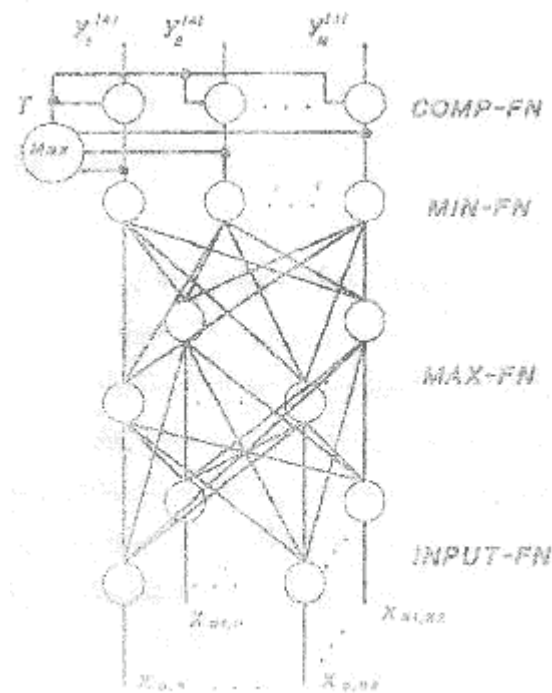
$$(8) Z = \sum_{i=1}^n w_i x_i$$

که این نرون مجموع ورودی ها را محاسبه می نماید .

ب- ساختار ابتدایی شبکه عصبی فازی مورد استفاده: [8]

شبکه مورد استفاده در حقیقت دارای چهار لایه پیش رو متشکل از نرون های فازی تعریف شده می باشد که ساختار

آن در شکل (2) نمایش داده شده است .



شکل ۲- ساختار شبکه ی عصبی فازی مورد استفاده

UPLOADED BY: RIAZIHA.COM

1- لایه نخست

هر نرون ورودی در این لایه مطابق با یک پیکسل (Pixel) الگوی ورودی است که می تواند هم مستقیماً مقدار واقعی پیکسل های تشکیل دهنده ی تصویر ورودی ما باشد (چنانکه در مورد الگوهای حروف الفبا مورد استفاده قرار گرفته) و هم می تواند حاوی مقدار کد گذاری شده (Encoded) مشخصه های مورد نظر طراح از تصویر ورودی باشد. (همان کاری که در این تحقیق برای تطبیق اثرانگشت انجام داده ایم). در این لایه از نرون های ورودی (Input-FN) استفاده شده است.

بسته به تصاویر دو بعدی مورد استفاده ، FN های ورودی این لایه به صورت دو بعدی نمایش داده شده و شماره گذاری شده اند. اگر الگوهای ورودی ما یک ماتریس $N_1 \times N_2$ باشد آنگاه تعداد نرون های فازی در این لایه برابر با $N_1 \times N_2$ نرون خواهد بود که روابط حاکم بر نرون فازی (i,j) ام در لایه نخست به صورت زیر بیان می گردد .

$$(9) \quad s_{ij}^{[1]} = z_{ij}^{[1]} = x_{ij} \quad \text{for} \quad i = 1 \text{ to } N_1, j = 1 \text{ to } N_2$$

$$(10) \quad y_{ij}^{[1]} = s_{ij}^{[1]} / X_{\max} \quad \text{for} \quad i = 1 \text{ to } N_1, j = 1 \text{ to } N_2$$

x_{ij} مقدار ورودی (i,j) ام از آرایه ورودی بوده و مقدار آن $0 \leq x_{ij} \leq X_{\max}$ می تواند باشد. با این کار تمامی خروجی های $y_{ij}^{[1]}$ این لایه نرمالیزه می شوند.

2- لایه دوم

این لایه به صورت دو بعدی و شامل $N_1 \times N_2$ نرون فازی بیشینه (Max-FN) با وظیفه فازی سازی الگوهای

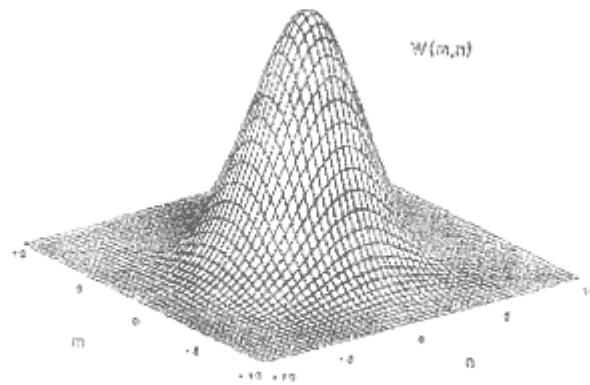
ورودی به واسطه یک تابع وزنی $w\{m,n\}$ است. حالت نرون (p,q) ام در این لایه به صورت:

$$(11) \quad s_{sq}^{[2]} = \max_{i=1}^{N_1} \left[\max_{j=1}^{N_2} [W[p-i, q-j] y_{ij}^{[1]}] \right]$$

جایی که $w[p-i, q-j]$ وزن ارتباطی (i,j) نرون ورودی در نخستین لایه به (p,q) امین نرون بیشینه در لایه دوم می باشد که به فرم زیر نیز تعریف می گردد.

$$(12) \quad w[m,n] = \exp(-\beta^2(m^2 + n^2)) \quad \text{for} \quad \begin{matrix} m=-(N_1-1)\text{to}(N_1-1) \\ n=-(N_2-1)\text{to}(N_2-1) \end{matrix}$$

تابع وزن $w\{m,n\}$ ، در حقیقت تابع فازی ساز در این شبکه می باشد، که طرحی از آن برای $\beta = 0.3$ در شکل (3) نشان داده شده است.



شکل ۳- تابع وزنی فازی ساز $w[m,n]$ با $\beta = 0.3$
 UPLOADED BY: RIJAZIHA.COM

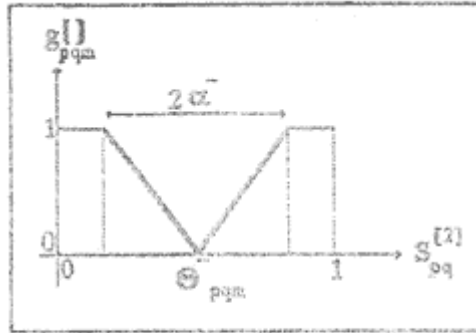
با استفاده از این تابع، وزن ارتباطی هر FN برای لایه دوم مانند یک لنز عمل می نماید که توسط آن هر FN روی یک پیکسل مرکزی از الگوی ورودی متمرکز (FOCUS) می شود و بسته به مقدار ضریب علاوه بر آن نقطه، نقاط پیرامون آن را نیز می بیند که توسط الگوریتم یادگیری بر روی آن تصمیم گیری می شود. هر نرون فازی بیشینه در این لایه به تعداد M خروجی متفاوت یکی به ازای هر FN در سومین لایه خواهد داشت، که به فرم زیر محاسبه می گردد.

$$(13) \quad y_{pqm}^{[2]} = g_{pqm}[s_{pq}^{[2]}] \quad \text{for } p = 1 \text{ to } N_1, q = 1 \text{ to } N_2, m = 1 \text{ to } M$$

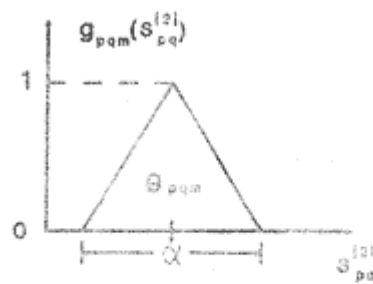
جایی که $y_{pqm}^{[2]}$ امین خروجی (p,q) امین Max-FN در لایه دوم است که به امین نرون فزی کمینه (Min-FN) در لایه سوم متصل شده است.

تابع خروجی $g_{pqm}[s_{pq}^{[2]}]$ توسط الگوریتم یادگیری تعیین می گردد که بستگی به نظر طراح و رویه ای که برای تطبیق در نظر گرفته است دارد. در حقیقت این تابع بیانگر شکل و نحوه تصمیم گیری ما بوده و تابع اساسی برای تصمیم گیری و نتایج آتی مقایسه می باشد. می توان این تابع را به گونه ای طراحی کرد که خروجی آن بیانگر عدم

شباهت بوده و بر اساس میزان عدم شباهت تصمیم‌گیری نماید [10] شکل (5)، یا طبق روش مورد استفاده ما تابع را بر اساس میزان شباهت طرح نموده و بر اساس آن نتایج آتی را پایه‌گذاری نماید .
 برای سادگی ما مثلث‌های متساوی‌الساقین با ارتفاع برابر با یک و قاعده برابر α مطابق شکل (6) را به عنوان تابع خروجی نرون‌های فازی بیشینه در لایه دوم انتخاب نموده‌ایم .



شکل ۴- تابع g ، خروجی لایه دوم
 (بر اساس عدم شباهت)



شکل ۵- تابع g ، خروجی مورد استفاده برای لایه دوم
 (میزان شباهت)

$$y_{pqm}^{[2]} = g_{pqm}[s_{pq}^{[2]}] = \begin{cases} 1 - 2|s_{pq}^{[2]} - \theta_{pqm}| / \alpha & \text{if } \phi \leq |s_{pq}^{[2]} - \theta_{pqm}| \leq \alpha / 2 \\ \phi & \text{if } \text{Otherwise} \end{cases} \quad (14)$$

for $\alpha \geq 0$, $p=1$ to N_1 , $q=1$ to N_2 , $m=1$ to M

UPLOADED BY: RIAZIHA.COM

در اینجا θ_{pqm} نقطه مرکزی قاعده تابع $g_{pqm}[s_{pq}^{[2]}]$ می‌باشد و در حقیقت تابع با دوری و نزدیکی به این نقطه‌ی مرکزی است که میزان شباهت یا عدم شباهت به یک الگوی مشخص را تشخیص می‌دهد، هر قدر از θ_{pqm} در تابع دورتر شویم میزان شباهت کمتر شده و مقدار خروجی نرون فازی نیز به سمت اعداد کوچکتر متمایل می‌شود یعنی در اینجا

$$y_{pqm} = 1 \quad \text{تطبیق کامل}$$

$$y_{pqm} = 0 \quad \text{(عدم شباهت کامل) عدم تطبیق کامل}$$

3- لایه سوم

نرون‌های فازی کمینه در لایه سوم یک الگوی یاد گرفته شده را معرفی نموده لذا تعداد آن‌ها در سومین لایه پس از پایان پروسه‌ی یادگیری مشخص خواهد گردید.

خروجی m امین، نرون کمینه در لایه سوم عبارتست از

$$(15) \quad y_m^{[3]} = s_m^{[3]} = \min_{p=1}^{N_1} (\min_{q=1}^{N_2} (y_{pqm}^{[2]})) \quad \text{for } m = 1 \text{ to } M$$

4- لایه چهارم

با استفاده از نرون‌های رقابتی (Comp-FN) در لایه چهارم هر الگوی مجزای آموزش یافته یک نرون فازی مقایسه‌ای را به خود اختصاص خواهد داد، بنابراین در این لایه M نرون مجزا، با نتیجه‌ی خروجی غیر فازی خواهیم داشت.

اگر یک آرایه ورودی، خیلی شبیه به m امین الگوی یاد گرفته شده باشد آنگاه خروجی m امین نرون مقایسه‌ای طبق روابط حاکم بر این نرون در لایه‌ی چهارم یک می شود، در حالی که خروجی‌های دیگر صفر خواهد بود.

$$s_m^{[4]} = z_m^{[4]} = y_m^{[3]} \quad \text{for } m = 1 \text{ to } M \quad (16)$$

$$y_m^{[4]} = g[s_m^{[4]} - T] = \begin{cases} 0 & \text{if } s_m^{[4]} < T \\ 1 & \text{if } s_m^{[4]} = T \end{cases} \quad \text{for } m = 1 \text{ to } M \quad (17)$$

$$T = \max_{m=1}^M (y_m^{[3]}) \quad \text{for } m = 1 \text{ to } M \quad (18)$$

تغییر و بهینه‌سازی شبکه

این شبکه دارای قابلیت‌های بالقوه‌ی بسیاری است که می تواند الگوهای مختلف با ویژگی‌ها و شاخصه‌های متنوع را تطبیق نماید. این شبکه با همین ساختار و تعداد 4 لایه برای تشخیص حروف الفبا و الگوهایی که دارای تغییرات محدود می باشند (مثل شیفت به طرفین به اندازه یک یا دو پیکسل، یا چرخش محدود) البته با تنظیم مناسب پارامترهای آن مثل α و β به خوبی عمل می کند اما اگر تغییر و دگرگونی تصویر به حدی باشد که میزان

شبهات تشخیص داده شده توسط شبکه از یک حد آستانه، بسته به نظر طراح، کمتر شود آن را به عنوان یک الگوی جدید قلمداد کرده و یک کلاس مجزا برای آن در نظر می‌گیرد و لذا تعداد نرون‌های فازی ما در لایه‌های سوم و چهارم شبکه یکی افزایش می‌یابد.

در این شبکه دچار این نقطه ضعف می‌باشیم که اگر دو تصویر از یک الگوی واحد، منتهی با قدری جابجایی، چرخش، اعوجاج، تاثیر نویز و حتی پاک شدن برخی قسمت‌ها به شبکه اعمال کنیم، دیگر قادر به تشخیص شبهات آن با الگوی اصلی آموزش دیده، نشده (البته میزان توانایی شبکه در تشخیص و تطبیق به پارامترهای شبکه نیز وابستگی زیادی دارد) و به صورت یک الگوی جدید تحت یک کلاس جدید شناخته می‌شود. مخصوصا در مورد تصاویر اثرانگشت که یکی از متغیرترین و انعطاف‌پذیرترین الگوهای تصویری می‌باشند. همانگونه که ذکر شد، ایجاد یک تصویر اثرانگشت مناسب به عوامل بسیار زیادی وابسته است و با کوچکترین تغییر در هنگام اخذ تصویر اثر، مشخصه‌ها به شدت تغییر می‌یابند.

بنابراین در اعمال این آرایه‌های ورودی که از یک انگشت بدست آمده به شبکه ما به جای یک کلاس که مشخص‌کننده‌ی اثرانگشت یک شخص باشد چندین کلاس (مختلف) بسته به توانایی شبکه در تشخیص الگوهای مشابه) خواهیم داشت بنابراین تصمیم‌گیری و شناخت ما دچار مشکل می‌شود. ما در اینجا راه حلی پیشنهاد کرده و به کار بسته‌ایم که تنها مختص اثرانگشت نبوده و به‌طور کل قابلیت شبکه را ارتقاء می‌دهد و به‌ویژه در مسئله شناخت و تشخیص حروف که در ابتدا این شبکه برای همین کاربرد توسط Cai و Kwan بداع گردید، نتایج بسیار عالی و بهینه ایجاد می‌کند.

افزودن طبقه پیشنهادی و طرح شبکه جدید 5 لایه

در این شیوه ما برای هر الگو و تصویر که مدنظر داریم و در مورد اثرانگشت، برای هر شخص، یک کلاس مجزی و منفرد در نظر می‌گیریم. سپس در این کلاس تصاویر اثرانگشت شخص را ذخیره می‌نماییم این تصاویر تشکیل می‌شود از حالت‌های مختلف ولی متعارفی که شخص انگشت خود را بر روی صفحه‌ی حساس اخذ تصویر قرار می‌دهد و به این طریق ما تصویر تمام حالات ممکن و متعارفی را که شخص ممکن است انگشت خود را قرار دهد بدست آورده‌ایم.

آنگاه پس از مراحل مختلف بازیافت و پردازش، مشخصات و پیکسل‌های هر اثر را به صورت آرایه مناسب تبدیل کرده و به شبکه اعمال می‌نماییم. شبکه شروع به یادگیری نموده و طی فرآیند یادگیری خود، تصاویر و آرایه‌های مشابه هم را تشخیص داده و همه را به صورت یک زیر کلاس (Sub class) در یک مجموعه واحد ذخیره می‌نماید.

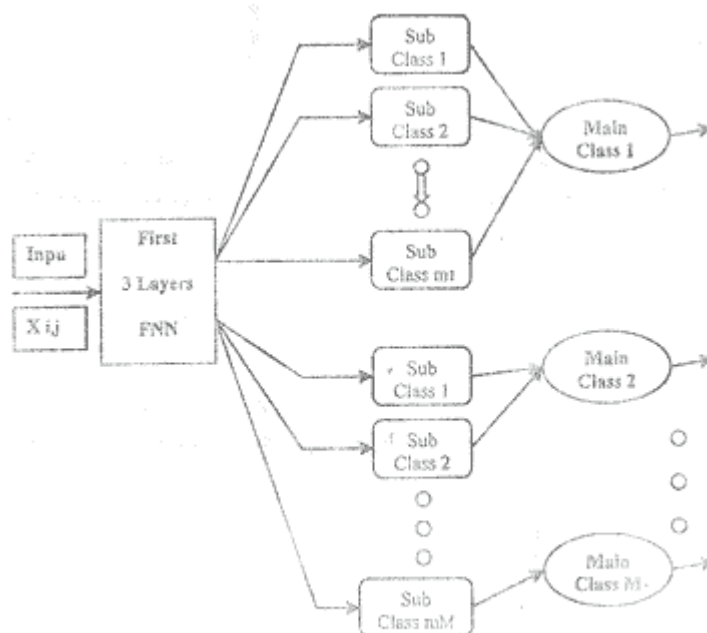
تصاویری را نیز که قادر به تشخیص تشابه آن‌ها با دیگر تصاویر گروه نباشد در یک زیر کلاس مجزای دیگر ذخیره می‌کند و برای آن یک نرون فازی مجزا در لایه سوم و چهارم در نظر می‌گیرد ما طبق شکل (6) تمام کلاس‌هایی را که در لایه چهارم ایجاد می‌شوند به عنوان زیر کلاس در نظر می‌گیریم آنگاه تمام این زیر کلاس‌ها را بسته به تعداد کلاس‌های اصلی طبقه‌بندی کرده و بررسی می‌کنیم هر کلاس (شخص) مجزی دارای چند زیر کلاس می‌باشد و

آنگاه خروجی‌های همه زیر کلاس‌های مربوط به آن را به یک نرون واحد در طبقه پنجم وصل می‌نماییم که این نرون بیانگر کلاس یا شخص منفرد اصلی ماست. تعریف لایه پنجم وقتی بخواهیم بدانیم الگو یا اثر انگشت ورودی به کدام کلاس اصلی وابسته است:

$$s_m^{[5]} = z_m^{[5]} = \sum_{m=1}^{m_j} y_m^{[4]} \quad (19)$$

$$y_j^{[5]} = g_j[s_m^{[5]}] \quad \text{for } j=1 \text{ to } M \quad (20)$$

$$y_j^{[5]} = g[s_m^{[5]}] = \begin{cases} 1 & \text{if } sm^{[5]} > 0 \\ 0 & \text{if } sm^{[5]} \leq 0 \end{cases} \quad (21)$$



شکل ۶- شبکه عصبی قازی بهینه با یک لایه پنجم

UPLOADED BY: RIAZIHA.COM

نتایج و جمع‌بندی

با افزودن لایه پنجم به شبکه قابلیت تشخیص شبکه افزایش چشم‌گیری می‌یابد چرا که تمام حالات متعارف یک الگو به شبکه آموزش داده شده و الگوی ورودی مثلاً متعلق به اولین کلاس وقتی به شبکه اعمال می‌شود حداقل با یکی از زیر کلاس‌های الگوی اصلی تشابه داشته و لذا خروجی مربوط به کلاس اصلی آن فعال می‌گردد.

امتیاز بسیار خوب دیگری که به این شبکه با این لایه اضافه شده این است که شبکه ابتدایی (4 لایه نرون قازی) یک شبکه در حقیقت دسته‌بندی کننده بود که از ابتدا ما نمی‌توانستیم بر تعداد کلاس‌های نهایی خروجی اشراف داشته باشیم و بدانیم که سر آخر چند کلاس خواهیم داشت و شبکه بسته به تنظیم پارامترهای آن کلاس‌های خروجی را بدون کنترل ناظر ایجاد می‌کرد که می‌توانست بیشتر از تعداد کلاس‌های مد نظر باشد. اما با اضافه کردن این لایه ما دقیقاً تعداد کلاس‌های خروجی را از ابتدا می‌دانیم و شبکه را نیز با اشراف کامل آموزش می‌دهیم بنابراین شبکه دسته‌بندی کننده تبدیل به شبکه کلاسه‌بندی کننده می‌شود آموزش آن نیز یادگیری با سرپرست می‌باشد. یعنی ما هم

الگوهای ورودی را داریم و هم می‌دانیم که این الگوها بایستی چه نتیجه‌ای را در خروجی ایجاد نمایند. تست این شبکه با یک بانک آثار انگشت مشتمل بر حدود صد اثر انگشت که از انگشت ده نفر متفاوت با حالت‌های مختلف اخذ شده بود انجام گردید.

شبکه بسته به مقدار پارامترهای تنظیمی خود بین 3 تا 7 زیر کلاس برای هر اثر انگشت در بین ده کلاس اصلی نماینده‌ی ده نفر ایجاد نمود. جدول یک، یکی از جداولی است که با تنظیم پارامترهای مورد نظر، تشابهات یافته شده توسط شبکه را در هنگام اعمال الگوهای اثر انگشت نشان می‌دهد.

آرایه ورودی پس از عبور از سه لایه اول با الگوهای ذخیره شده در شبکه (که طی فرآیند یادگیری آموخته شده‌اند) مقایسه شده و به هر کدام که شبیه‌تر بود جزء آن طبقه قلمداد خواهد گردید و یا در صورت عدم شباهت به هیچ‌کدام، خود یک شبکه جدید را ایجاد می‌کند یعنی در اینجا ما در مرحله تست و بررسی، باز هم مرحله یادگیری را خواهیم داشت و هر دو عمل در این شبکه توأمان قابل اجراست. همچنین از آنجا که خروجی لایه سوم بیانگر میزان شباهت به الگوهای آموزش یافته است ما می‌توانیم علاوه بر طبقه‌بندی الگوها درصد شباهت الگوی ورودی به الگوهای کلاس‌مان را نیز به این صورت به دست آوریم .

جدول ۱- تشابهات یافت شده با شبکه عصبی_فازی بهینه با پارامترهای $\alpha = 2.5, Tf = 0.3, \beta = 0.1$

شخص	تعداد الگوها		شباهتهای یافت شده			
	کامل	گروه‌بندی	۱	۲	۳	۴
۱	۱۰	۱	تمام الگوها	-	-	-
۲	۱۰	۲	۱۰۲ و ۳ و ۴ و ۵ و ۶ و ۷	۱۰ و ۹ و ۸	-	-
۳	۱۲	۴	۱ و ۲ و ۳ و ۴ و ۵ و ۶ و ۷ و ۸ و ۹	۳ و ۲	۵ و ۶ و ۷ و ۸ و ۹ و ۱۰ و ۱۱ و ۱۲	۱۰
۴	۱۲	۵	۱ و ۲ و ۳ و ۴ و ۵ و ۶ و ۷ و ۸ و ۹ و ۱۰	۳ و ۵ و ۶ و ۷ و ۸ و ۹	۸ و ۷	۱۱
۵	۱۰	۴	۱	۲	۳ و ۵ و ۶ و ۷ و ۸ و ۹ و ۱۰	۴ و ۷ و ۹ و ۱۰
۶	۵	۲	۲ و ۱	۳ و ۵ و ۶ و ۷ و ۸ و ۹ و ۱۰	-	-
۷	۵	۲	۱ و ۲ و ۳ و ۴	۵	-	-
۸	۷	۴	۱	۲ و ۳ و ۴	۵ و ۶	۷
۹	۵	۴	۱	۲ و ۳	۴	۵
۱۰	۱۲	۲	۱ و ۲ و ۳ و ۴ و ۵ و ۶ و ۷ و ۸ و ۹ و ۱۰ و ۱۱ و ۱۲	۶ و ۷ و ۸ و ۹	۱۱	-

UPLOADED BY: RIAZIHA.COM

منابع:

- [1] منہاج، محمدباقر، (۱۳۷۸)، "مبانی شبکه‌های عصبی، انتشارات دانشگاه صنعتی امیرکبیر.
- [2] Rumelhart D. E, McClelland J. L.,(1986),"Parallel Distributed Processing: Explorations in the Microstructure of Cognition",MIT Press, Vol.1:Foundations.
- [3] Lau C. Ed., "Neural Networks",(1992),NJ:IEEE Press,Theoretical Foundations and Analysis .Piscataway.
- [4] Fukumi M,Omatu. S, Takeda. F,"Rotation-invariant neural pattern recognition system with application to coin recognition."IEEE Trans. Neural Networks,vol.3,no.2,pp.272-279,Mar.1992
- [5] Bezdek J.C and Pal .S.,,Eds., "Fuzzy Models for Pattern recognition".Piscataway,NJ:IEEE Press,1992
- [6] Hariri .M,"Using Neuro-Fuzzy systems in pattern recognition", fifth conference on Intelligent Systems 14-16 oct,2003.Mashhad,Iran
- [7] Yonakawa T,and Tomoda S.,(1989),"A fuzzy neuron and its application to pattern recognition" , Proc.3rd,Fuzzy System Associate Congress.Japan, pp.30_38
- [8] Kwan.H.K, Cai .Y,"A Fuzzy Neural Network and it's Application to Pattern Recognition".IEEE Trans.Fuzzy Systems, August 1994,vol2,no-3,
- [9] Nikhil R. Pal,Gautam K. Mandal,and Eluri V. K.,Comments on" Fuzzy Neural Network and it's Application to Pattern Recognition"IEEE Trans. on fuzzy systems, August 1999 vol.7,No.4.
- [10] منہاج، محمدباقر، عزیززادہ، حمید، طبقہ‌بندی و تشخیص الگو با استفاده از شبکه عصبی ... فازی، مجله امیرکبیر سال هشتم، شماره ۳۱، ص ۱۹۵-۲۰۵.

h_kohsari@yahoo.com