

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



دانشگاه صنعتی شاهرود

دانشکده : برق و **ریاتیک**

گروه : الکترونیک

تشخیص صحت امضای دستنویس بصورت برون خط

دانشجو : نیوشا نوائی

استاد راهنما :

دکتر علی سلیمانی ایوری

استاد مشاور :

دکتر حسین مروی

پایان نامه ارشد جهت اخذ درجه کارشناسی ارشد

ماه و سال انتشار : شهریور ۹۱

شماره : ۰۹۷۹/آت.ب
تاریخ : ۹۱/۰۶/۲۷
ویرایش : -----

بسمه تعالی



فرم صورت جلسه دفاع پایان نامه تحصیلی دوره کارشناسی ارشد

با تأییدات خداوند متعال و با استعانت از حضرت ولی عصر (عج) جلسه دفاع از پایان نامه کارشناسی ارشد خانم / آقای :
نیوشا بنوایی رشته : برق گرایش : الکترونیک
تحت عنوان : **تشخیص و صحت امضای دستنویس بصورت بیرون خط**
که در تاریخ ۹۱/۰۶/۲۷ با حضور هیأت محترم داوران در دانشگاه صنعتی شاهرود برگزار گردید به شرح زیر است :

<input type="checkbox"/> مردود	<input type="checkbox"/> دفاع مجدد	<input checked="" type="checkbox"/> امتیاز (۱۷/۵)
--------------------------------	------------------------------------	---

۱- عالی (۲۰ - ۱۹)

۲- بسیار خوب (۱۸/۹۹ - ۱۸)

۳- خوب (۱۷/۹۹ - ۱۶)

۴- قابل قبول (۱۵/۹۹ - ۱۴)

۵- نمره کمتر از ۱۴ غیر قابل قبول

عضو هیأت داوران	نام و نام خانوادگی	مرتبه علمی	امضاء
۱- استاد راهنما	سید مرتضی	استاد	
۲- استاد مشاور	دکتر حسین برزنجی	استاد	
۳- نماینده شورای تحصیلات تکمیلی	مرتضی رحمان	استاد	
۴- استاد ممتحن	رضا زاهدی	استاد	
۵- استاد ممتحن	مسئول جناب	استاد	

رئیس دانشکده :

تقدیم به پدر عزیز و مهربانم که ستودنیست

تشکر و قدردانی

در ابتدا از استاد راهنمای خود، جناب آقای دکتر علی سلیمانی به خاطر راهنمایی و حمایتی

که از من در طی انجام این تحقیق کرده‌اند، نهایت تشکر را می‌نمایم.

همچنین از دکتر مروی، دکتر احمدی فرد و دکتر معروضی به خاطر کمک‌های بی‌دریغشان، بسیار

متشکرم.

در نهایت از خانواده ام که در تمام لحظات انجام این تحقیق مرا پشتیبان بودند، تشکر ویژه می

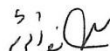
نمایم.

تعهد نامه

اینجانب نورینا نورانی دانشجوی دوره کارشناسی ارشد رشته مهندسی صنایع دانشکده مهندسی صنایع دانشگاه صنعتی شاهرود نویسنده پایان نامه تاثیر کیفیت ارضای دستپوش بر صورت بیرون خط تحت راهنمایی دکتر علی محمدی متعهد می شوم .

- تحقیقات در این پایان نامه توسط اینجانب انجام شده است و از صحت و اصالت برخوردار است .
- در استفاده از نتایج پژوهشهای محققان دیگر به مرجع مورد استفاده استناد شده است .
- مطالب مندرج در پایان نامه تاکنون توسط خود یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ جا ارائه نشده است .
- کلیه حقوق معنوی این اثر متعلق به دانشگاه صنعتی شاهرود می باشد و مقالات مستخرج با نام « دانشگاه صنعتی شاهرود » و یا « Shahrood University of Technology » به چاپ خواهد رسید .
- حقوق معنوی تمام افرادی که در به دست آمدن نتایج اصلی پایان نامه تأثیر گذار بوده اند در مقالات مستخرج از پایان نامه رعایت می گردد.
- در کلیه مراحل انجام این پایان نامه ، در مواردی که از موجود زنده (یا بافتهای آنها) استفاده شده است ضوابط و اصول اخلاقی رعایت شده است .
- در کلیه مراحل انجام این پایان نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته یا استفاده شده است اصل رازداری ، ضوابط و اصول اخلاق انسانی رعایت شده است .

تاریخ ۹۱، ۶، ۳۱

امضای دانشجو 

مالکیت نتایج و حق نشر

- کلیه حقوق معنوی این اثر و محصولات آن (مقالات مستخرج ، کتاب ، برنامه های رایانه ای ، نرم افزار ها و تجهیزات ساخته شده است) متعلق به دانشگاه صنعتی شاهرود می باشد . این مطلب باید به نحو مقتضی در تولیدات علمی مربوطه ذکر شود .
- استفاده از اطلاعات و نتایج موجود در پایان نامه بدون ذکر مرجع مجاز نمی باشد.

چکیده

امروزه استفاده از ویژگی‌های زیستی برای تأیید هویت افراد در حال افزایش است. از جمله معروف ترین خواص زیستی که در این زمینه به کار برده می شود، می توان به اثر انگشت، عنبیه، شبکیه، چهره، صوت و امضاء اشاره کرد. در نظام اداری، بخش‌های زیادی به ویژه در زمینه‌های اقتصادی و تجاری مانند بانک‌ها، مؤسسات و سازمان‌ها، خواهان روش‌هایی برای تعیین هویت افراد هستند. یکی از این ابزارها که به علت سادگی و داشتن هزینه کم گستردگی فراوانی دارد، امضاء می‌باشد. اما امضای افراد در معرض جعل توسط دیگران قرار دارد. بنابراین یافتن روشی کارا برای بازشناسی امضاءهای اصلی از امضاءهای جعلی، لازم و مفید می‌نماید.

تأیید امضاء بصورت خودکار می تواند از لحاظ نحوه دریافت داده‌ها به دو گروه برخط^۱ و برون خط^۲ تقسیم گردد. در روش بر خط به کمک یک صفحه و یک قلم الکترونیکی متصل به کامپیوتر، اطلاعات امضاء و اطلاعات مشخصه های دینامیکی آن از قبیل فشار و سرعت و ... به دست می‌آید. در حالیکه در روش برون خط، اطلاعات هنگامی جمع آوری می‌شود که عمل نوشتن امضاء تمام شده باشد. سپس در زمانی دیگر امضای نوشته شده، توسط یک اسکنر به تصاویر دیجیتال تبدیل می‌شود. در این صورت ویژگی‌های استخراج شده، ایستا^۳ خوانده می‌شوند. سیستم‌های برون خط کاربردی تر می‌باشند و نسبت به سیستم‌های بر خط، در بسیاری از نقاط دنیا به عنوان سیستم‌هایی با دسترسی ساده تر مورد استفاده قرار می‌گیرند. هر چند به دلیل عدم وجود اطلاعات دینامیکی نسبت به سیستم‌های برخط، مشکل تر فرض می‌شوند.

^۱ Online

^۲ Offline

^۳ Static

در این پایان نامه سیستم برون خط مبنای کار ما می‌باشد. ابتدا برای پیش پردازش، فضای سفید اضافی تصویر اسکن شده‌ی امضاء، حذف می‌شود و سپس با ایجاد ناحیه بندی خاص بر روی تصاویر، ویژگی‌های محلی جدیدی را بر مبنای تقسیم بندی شبکه‌ای استخراج می‌نماییم. در ادامه زیرمجموعه‌ای بهینه از این ویژگی‌ها به کمک الگوریتم رقابت استعماری^۱ (ICA) انتخاب می‌شود. در مرحله طبقه بندی برای هر کاربر از یک شبکه عصبی مصنوعی^۲ (ANN) مجزا استفاده می‌شود که برای فاز آموزش ۲۷ امضای آن کاربر را در نظر می‌گیریم. برای ارزیابی ۲۷ امضای باقیمانده هر فرد به شبکه عصبی آموزش دیده متناظر با آن کاربر داده می‌شود و تصمیم گیری در مورد تأیید یا رد امضاء انجام می‌گردد. در انتها نیز نتایج به دست آمده با کارهای مشابه مقایسه می‌شود.

ما در این پایان نامه از پایگاه داده ۳۰۰ GRAY GPDS برای ارزیابی الگوریتم پیشنهادی استفاده می‌کنیم. نرخ خطای برابر^۳ (EER) الگوریتم پیشنهادی برای این پایگاه داده ۶,۹۶ می‌باشد.

نتایج تجربی نشان می‌دهد که الگوریتم پیشنهادی در مقایسه با روش‌های ارائه شده از دقت قابل قبولی برخوردار است .

کلمات کلیدی

تشخیص امضاء، مشخصه‌های ایستا، انتخاب ویژگی، الگوریتم رقابت استعماری، ویژگی شبکه‌ای،

^۱ Imperialist Competitive Algorithm

^۲ Artificial Neural Network

^۳ Equal Error Rate

فهرست مطالب

فهرست شکلها	ل
فهرست جداول	ن
فصل ۱- مقدمه	۱
۱-۱- مقدمه	۲
۲-۱- مروری بر کارهای انجام شده	۳
۳-۱- هدف پایان نامه	۵
۴-۱- ساختار پروژه	۵
فصل ۲- سیستم تایید امضاء برون خط	۷
۱-۲- مقدمه	۸
۲-۲- جمع آوری داده و عملیات پیش پردازش	۸
۳-۲- استخراج ویژگی	۹
۴-۲- روش های تصمیم گیری	۹
۱-۴-۲- روش تطبیق الگو	۱۰
۲-۴-۲- روش های شبکه عصبی	۱۰
۳-۴-۲- روش مدل مخفی مارکوف	۱۱
۴-۴-۲- روش های آماری	۱۱
۵-۲- معیار های سنجش عملکرد یک سیستم تشخیص صحت امضا	۱۲
۶-۲- بررسی ویژگی های گوناگون امضا	۱۳
۱-۶-۲- نسبت طول به عرض یک امضا	۱۳
۲-۶-۲- ناحیه پیکسل های سیاه	۱۴
۳-۶-۲- ناحیه پیکسل های سیاه نرمال شده	۱۴
۴-۶-۲- تعداد نقاط لبه	۱۴
۵-۶-۲- تعداد نقاط تقاطع	۱۵
۶-۶-۲- نگاشت افقی	۱۵
۷-۶-۲- نگاشت عمودی	۱۶
۸-۶-۲- تعداد حلقه ها ی موجود در یک امضا	۱۷
۹-۶-۲- خط اصلی	۱۸
۱۰-۶-۲- ویژگی های مربوط به مرکز ثقل امضا	۱۸
۱۱-۶-۲- نقاط ویژگی بر اساس انشعاب دادن بصورت عمودی	۱۹
۱۲-۶-۲- ویژگی های محلی یا شبکه ای	۲۰
۷-۲- مجموعه امضاءهای مورد استفاده در این پایان نامه	۲۲
فصل ۳- انتخاب ویژگی	۲۴

۲۵	۱-۳-مقدمه
۲۵	۲-۳-تعاریف
۲۹	۳-۳-روشهای مختلف انتخاب ویژگی
۲۹	۱-۳-۳-توابع تولید کننده
۳۱	۲-۳-۳-تابع ارزیابی
۳۵	فصل ۴- الگوریتم رقابت استعماری
۳۶	۱-۴-مقدمه
۳۶	۲-۴-مروری تاریخی بر پدیده استعمار
۳۹	۳-۴-الگوریتم رقابت استعماری
۳۹	۱-۳-۴-شکل دهی امپراطوری های اولیه
۴۱	۲-۳-۴-سیاست جذب و حرکت مستعمره ها به سمت امپریالیست
۴۲	۳-۳-۴-انقلاب و تغییر ناگهانی در موقعیت یک کشور
۴۲	۴-۳-۴-جابجایی موقعیت مستعمره وامپریالیست
۴۳	۴-۳-۴-رقابت استعماری
۴۴	۶-۳-۴-سقوط امپراطوری های ضعیف
۴۶	فصل ۵- الگوریتم پیشنهادی
۴۷	مقدمه
۴۷	۱-۵-بلوک دیاگرام روش پیشنهادی
۴۸	۲-۵-پیش پردازش
۵۰	۱-۲-۵-از بین بردن فضای سفید اضافی
۵۱	۳-۵-ناحیه بندی تصویر امضاء بصورت M در N
۵۷	۴-۵-استخراج ویژگی
۵۷	۱-۴-۵-ویژگی های محلی
۵۹	۲-۴-۵-ویژگی های سراسری
۶۰	۵-۵-فرایند آموزش و آزمایش با استفاده از شبکه عصبی
۶۱	۱-۵-۵-آموزش شبکه عصبی
۶۲	۲-۵-۵-آزمایش شبکه های عصبی
۶۲	۳-۵-۵-تصمیم گیری با استفاده از شبکه عصبی
۶۳	۶-۵-معیار میزان خطای سیستم
۶۵	۷-۵-انتخاب ویژگی
۶۶	۱-۷-۵-استفاده از الگوریتم رقابت استعماری بعنوان الگوریتم انتخاب ویژگی
۶۷	۵-۷-۲-الگوریتم رقابت استعماری برای حل مسائل انتخاب ویژگی
۷۲	۳-۷-۵-تعیین تابع هزینه مناسب برای مرحله انتخاب ویژگی
۷۳	۴-۷-۵-تابع هزینه بر اساس خطای شبکه عصبی
۷۳	۵-۷-۵-تابع هزینه بر اساس معیار Fisher
۷۵	۸-۵-نتیجه گیری
۷۶	فصل ۶- پیاده سازی الگوریتم پیشنهادی و ارزیابی نتایج

- ۶-۱- مقدمه..... ۷۷
- ۶-۲- نتایج شبیه سازی الگوریتم پیشنهادی برای پایگاه داده GPDS۳۰۰ GRAY..... ۷۸
- ۶-۲-۱- نتایج شبیه سازی برای استخراج ویژگی بصورت سنتی..... ۷۸
- ۶-۲-۲- نتایج شبیه سازی برای استخراج ویژگی با روش پیشنهادی..... ۸۰
- ۶-۲-۳- مقایسه روش پیشنهادی برای تقسیم بندی امضاء با روش سنتی تقسیم بندی برای
مرحله استخراج ویژگی ۸۲
- ۶-۳- نتایج شبیه سازی بازای ویژگی های انتخاب شده با الگوریتم رقابت استعماری..... ۸۴
- ۶-۳-۱- نتایج شبیه سازی با ویژگی های انتخاب شده ی الگوریتم رقابت استعماری با تابع
معیار خطای شبکه عصبی..... ۸۵
- ۶-۳-۲- نتایج شبیه سازی با ویژگی های انتخاب شده ی الگوریتم رقابت استعماری با تابع معیار
Fisher ۸۶
- ۶-۳-۳- مقایسه نتایج استفاده از ویژگی های انتخاب شده توسط الگوریتم رقابت استعماری
- بازای تابع هزینه های متفاوت..... ۸۷
- ۶-۴- مقایسه الگوریتم پیشنهادی با دیگر کارهای انجام شده..... ۹۰
- ۶-۵- مقایسه الگوریتم پیشنهادی با سیستم های ارائه شده در مسابقه ۲۰۱۰ NSIGCOMP۴..... ۹۱
- ۶-۶- نتیجه گیری..... ۹۳
- ۶-۷- پیشنهادات..... ۹۴
- ضمیمه ا - شبکه های عصبی مصنوعی چند لایه..... ۹۵**
- ا-۱- تاریخچه شبکه های عصبی مصنوعی..... ۹۵
- ا-۲- اساس شبکه های عصبی مصنوعی..... ۹۶
- ا-۲-۱- نرون بیولوژیکی..... ۹۷
- ا-۳- چگونگی کارکرد شبکه های عصبی..... ۹۸
- ا-۴- معماری شبکه عصبی مصنوعی..... ۹۹
- ا-۵- شبکه های عصبی چند لایه..... ۱۰۲
- ا-۶- آموزش شبکه های عصبی..... ۱۰۳
- ا-۶-۱- یادگیری نظارت نشده یا بدون ناظر..... ۱۰۳
- ا-۶-۲- یادگیری نظارت شده یا یادگیری با ناظر..... ۱۰۴
- ا-۷- شبکه عصبی پس انتشار (BNN)..... ۱۰۵
- ا-۸- آموزش شبکه پس انتشار..... ۱۰۸
- ضمیمه ب - مسابقه تأیید صحت امضای برون خط ۲۰۱۰ NSIGCOMP۴..... ۱۱۰**
- ب-۱- معرفی پایگاه داده های GPDS۹۶۰..... ۱۱۱
- ب-۲- نتایج مسابقه..... ۱۱۲

فهرست شکلها

- شکل ۱-۲ (الف) تبدیل تصویر با سطح خاکستری به تصویر سیاه و سفید (ب) عملیات حذف نویز (ج) حذف قسمت های سفید اضافی ۹
- شکل ۲-۲ نمودار تغییرات خطای FRR و FAR به ازای مقادیر مختلف آستانه ۱۳
- شکل ۳-۲ تعیین طول و عرض یک امضا ۱۴
- شکل ۴-۲ نمایش نقاط لبه و نقاط تقاطع یک امضاء ۱۵
- شکل ۵-۲ نمودار توزیع فراوانی پیکسل های سیاه در هر ستون ۱۶
- شکل ۶-۲ قله توزیع فراوانی ۱۶
- شکل ۷-۲ نمودار توزیع فراوانی پیکسل های سیاه در هر ردیف ۱۷
- شکل ۸-۲ حلقه های موجود در امضاهای مختلف ۱۷
- شکل ۹-۲ خط اصلی پایینی ۱۸
- شکل ۱۰-۲ مرکز ثقل یک امضا ۱۹
- شکل ۱۱-۲ تقسیم بندی به کمک مرکز ثقل ۲۰
- شکل ۱۲-۲ مثالی از ناحیه بندی بصورت ساده ۲۱
- شکل ۱۳-۲ مثالی از ناحیه بندی دایره ای ۲۲
- شکل ۱۴-۲ شکل هندسی چند نمونه از مجموعه امضاءهای GPDS۳۰۰ GRAY ۲۳
- شکل ۱-۳ فرآیند انتخاب ویژگی ۲۷
- شکل ۱-۴ اعمال سیاست جذب از طرف استعمارگران بر مستعمرات ۳۸
- شکل ۲-۴ تناظر متغیرهای بهینه سازی مسئله با ویژگی های اجتماعی سیاسی ۴۰
- شکل ۳-۴ نحوه تقسیم مستعمرات، میان کشور های استعمارگر ۴۱
- شکل ۴-۴ اعمال سیاست انقلاب ۴۲
- شکل ۵-۴ جابجایی موقعیت مستعمره و استعمارگر ۴۳
- شکل ۶-۴ رقابت استعماری میان چندین استعمارگر ۴۴
- شکل ۷-۴ سقوط امپراطوری هادر روند چرخه الگوریتم رقابت استعماری ۴۵
- شکل ۱-۵ بلوک دیاگرام کلی سیستم تأیید امضاء برون خط ارائه شده در این پایان نامه ۴۸

- شکل ۵-۲ عملیات حذف ناحیه سفید اضافی ۵۰
- شکل ۵-۳ تقسیم بندی امضاء به روش ساده ۵۲
- شکل ۵-۴ تقسیم بندی سنتی ۵۳
- شکل ۵-۵ ناحیه بندی امضاء بکمک روش پیشنهادی رکیک ۵۴
- شکل ۵-۶ ناحیه بندی امضاء بکمک روش پیشنهادی سوان پل ۵۴
- شکل ۵-۷ نمایش سلول‌های خالی در روش ناحیه بندی سنتی ۵۵
- شکل ۵-۸ تقسیم بندی امضاء در راستای افق به کمک روش پیشنهادی ۵۶
- شکل ۵-۹ تقسیم بندی امضاء به روش پیشنهادی ۵۷
- شکل ۵-۱۰ نمودار چگونگی به دست آوردن EER ۶۵
- شکل ۵-۱۱ تقاطع چند نقطه‌ای ۷۰
- شکل ۵-۱۲ تقاطع یکنواخت ۷۱
- شکل ۶-۱ مقایسه روش تقسیم بندی پیشنهادی با روش‌های تقسیم بندی سنتی و دایره ای ۸۳
- شکل ۶-۲ روند بهینه‌سازی به کمک الگوریتم رقابت استعماری با گذشت دهه ها ۸۵
- شکل ۶-۳ مقایسه کارایی سیستم با و بدون استفاده از الگوریتم رقابت استعماری و بازای توابع هزینه‌های گوناگون ۸۸

فهرست جداول

جدول ۱-۳	مقایسه توابع ارزیابی مختلف	۳۴
جدول ۱-۵	ویژگی‌های محلی استخراج شده از هر سلول	۵۷
جدول ۲-۵	ویژگی‌های محلی استخراج شده از هر سلول	۵۹
جدول ۱-۶	نتایج مربوط به تقسیم بندی شبکه ۱*۲ سنتی	۷۹
جدول ۲-۶	نتایج مربوط به تقسیم بندی شبکه ۳*۳ سنتی	۷۹
جدول ۳-۶	نتایج مربوط به تقسیم بندی شبکه ۳*۴ سنتی	۷۹
جدول ۴-۶	نتایج مربوط به تقسیم بندی شبکه ۴*۴ سنتی	۸۰
جدول ۵-۶	نتایج مربوط به تقسیم بندی شبکه ۵*۵ سنتی	۸۰
جدول ۶-۶	نتایج مربوط به تقسیم بندی شبکه دایره ای ۸ قطاعی	۸۰
جدول ۷-۶	نتایج مربوط به تقسیم بندی شبکه ۱*۲ و افقی	۸۱
جدول ۸-۶	نتایج مربوط به تقسیم بندی شبکه ۳*۳ و افقی	۸۱
جدول ۹-۶	نتایج مربوط به تقسیم بندی شبکه ۳*۴ و افقی	۸۱
جدول ۱۰-۶	نتایج مربوط به تقسیم بندی شبکه ۴*۴ و افقی	۸۲
جدول ۱۱-۶	نتایج خطای EER بازای تقسیم بندی های گوناگون	۸۲
جدول ۱۲-۶	نتایج استفاده از ویژگی‌های انتخاب شده به کمک ICA با تابع معیار خطای شبکه عصبی	۸۶
جدول ۱۳-۶	نتایج استفاده از ویژگی‌های انتخاب شده به کمک ICA با تابع معیار fisher و نرخ انقلاب ۰,۱	۸۶
جدول ۱۴-۶	نتایج استفاده از ویژگی‌های انتخاب شده به کمک ICA با تابع معیار fisher و نرخ انقلاب ۰,۲	۸۷
جدول ۱۵-۶	نتایج استفاده از ویژگی‌های انتخاب شده به کمک ICA با تابع معیار fisher و نرخ انقلاب ۰,۳	۸۷
جدول ۱۶-۶	زمان و خطای EER حاصل از استفاده ویژگی‌های انتخاب شده به کمک ICA بازای نرخ انقلاب و توابع هزینه گوناگون	۸۷

- جدول ۶-۱۷ مقایسه سیستم در حالت های با و بدون استفاده از الگوریتم رقابت استعماری از نظر زمانی و دقت سیستم ۸۹
- جدول ۶-۱۸ مقایسه الگوریتم پیشنهادی با کارهای مشابه ۹۱
- جدول ۶-۱۹ جدول نرخ خطای FRRS برای سیستم های شرکت داده شده در مسابقه ۲۰۱۰ NSigComp ۴ و سیستم ارائه شده در این پایان نامه ۹۲

مقدمه

۱-۱- مقدمه

سیستم شناسایی امضاء یک سیستم بیومتریک^۱ است که با استفاده از روش‌های هوشمند اتوماتیک، هویت انسان را بر اساس ویژگی‌های رفتاری، تشخیص و یا تأیید می‌کند. در حالت عمومی در یک سیستم بیومتریک از ویژگی‌های فیزیولوژی مانند اثر انگشت، شبکیه، عنبیه، صورت، گوش و اثر کف دست و یا از ویژگی‌های رفتاری مانند صدا، دستخط و امضاء به منظور شناسایی استفاده می‌شود. ارزیابی بیومتریک‌ها، یک مسئله چند بعدی می‌باشد و در طراحی یک سیستم بیومتریکی باید پارامترهایی نظیر دقت، هزینه، سرعت و قابلیت پذیرش در بین کاربران را در نظر داشت.

هر چند مجموعه‌ی بسیار گسترده‌ای از بیومتریک‌ها موجود است، اما هیچ کدام بطور کامل مشخصات مطلوب لازم برای یک سیستم بیومتریکی را نداشته‌اند [۱]. امضاءهای دستنویس بر اساس این واقعیت که زمانی طولانی به عنوان ابزاری برای تشخیص هویت، مورد استفاده بشر قرار گرفته است، جایگاه بسیار ویژه‌ای در بین این مجموعه وسیع بیومتریک‌ها دارد. حتی قبل از آنکه کامپیوترها اختراع شوند نیز در اسناد بانکی و حقوقی به عنوان ابزاری برای تأیید سند کاربرد داشته است و هنوز هم معروف‌ترین روش تأیید سند، در معاملات با مدارک و فرم‌های کاغذی می‌باشد.

در این فصل ابتدا در بخش ۱-۲، برخی کارهای انجام شده در زمینه تشخیص صحت امضاء را معرفی می‌کنیم و در بخش ۱-۳ به شرح انگیزه و هدف اصلی پروژه پرداخته، سپس ساختار کلی پروژه و محتوای فصول را در بخش ۱-۳ بیان می‌کنیم.

^۱ Biometric

۲-۱- مروری بر کارهای انجام شده

از حدود سه دهه پیش تحقیقات زیادی در زمینه‌ی تأیید امضاء انجام گرفته است. در دهه ۱۹۸۰، عمار^۱ و همکارانش [۲] اطلاعات آماری از نقاط پر فشار برای شناسایی تغییرات در امضاءها استخراج کردند. سابورین^۲ و پلامندن^۳ [۳] با بررسی جهت های تکه خط ها بصورت محلی و اطلاعات آماری که از آن به دست می‌آید، تمایز بین دستنوشته اشخاص را مورد بررسی قرار دادند. همچنین سابورین و همکارانش بار دیگر [۴، ۵] از یک کد سایه‌ای توسعه یافته برای بردار ویژگی‌ها استفاده کردند تا ویژگی‌های محلی و سراسری را برای تأیید امضاء به کار برند. هان^۴ و ستی^۵ [۶] از ویژگی‌های هندسی و همبندی برای توصیف امضاءها بهره گرفتند و توزیع فضایی این ویژگی‌ها را در نگاشت امضاء به یک رشته از نمادها بررسی کردند.

کاواکانی^۶ و همکارانش [۷] از ویژگی‌های ساختاری و همچنین ویژگی‌های شبه دینامیکی در مطالعات خود استفاده کردند. ازگوندوز^۷ و همکارانش [۸] یک روش تشخیص صحت امضاء با استفاده از ویژگی‌های سراسری و جهتی و ویژگی‌های قسمت بندی شده، ارائه دادند. آنها نشان دادند که برای روش پیشنهادیشان کلاس بندی به کمک ماشین بردار پشتیبان کارایی بهتری نسبت به شبکه‌های عصبی چند لایه دارد. کواتزر^۸ [۹] از تبدیل رادون^۹ گسسته برای استخراج ویژگی‌های سراسری و یک مدل مخفی مارکوف در یک الگوریتم جدید شناسایی امضاء استفاده

^۱ Ammar

^۲ Sabourin

^۳ Plamondon

^۴ Han

^۵ Sethi

^۶ Cavacanti

^۷ Ozgunduz

^۸ Coetzer

^۹ Radon Transform

کرد. در روش پیشنهادی آنها، تبدیل رادون گسسته در زوایای ۰ تا ۳۶۰ درجه محاسبه شد و هر کدام از رشته های مشاهده شده با یک مدل مخفی مارکوف که حالت‌هایش بصورت حلقوی بود، مدل شد. به منظور مدل‌سازی و تصحیح، برای امضای هر نفر یک مدل مخفی مارکوف در نظر گرفته شد. سیستم آنها نسبت به جهت، تغییرناپذیر و مقاوم بود. در سال ۲۰۱۰ اسمایل^۱ و همکارانش از ویژگی‌هایی بر اساس کد زنجیره ای^۲ و تبدیل فوریه^۳، برای توصیف امضاء استفاده کردند.

در ایران نیز کارهایی در این زمینه صورت گرفته است. محمدی در [۱۰] یک سیستم تشخیص امضای دستنویس فارسی با استفاده از الگوریتم آنالیز اجزای اصلی^۴ (PCA) و شبکه عصبی چند لایه ارائه داد. سیگاری و پور شهابی در [۱۲،۱۱] یک روش شناسایی امضاء بر اساس تبدیل ویولت گابور^۵ بعنوان استخراج کننده ویژگی و ماشین بردار پشتیبان برای کلاسه بندی پیشنهاد کردند. آنها پس از نرمال‌سازی و حذف نویز ضرایب گابور را برای هر نقطه به دست آوردند و سپس به عنوان بردار ورودی به ماشین بردار پشتیبان دادند. تعداد ماشین بردار پشتیبان برابر با تعداد کلاس‌ها بود و هر کدام تعیین می کرد که آیا تصویر ورودی به کلاس درست تعلق دارد یا خیر. آنها برای مقایسه، از دو مجموعه امضاء استفاده نمودند و به دقت ۹۶٪ برای مجموعه امضای فارسی و ۹۳٪ برای مجموعه امضاءهای ترکی دست یافتند. مجموعه امضاءهای فارسی که آنها در تحقیقشان استفاده کردند، همان مجموعه امضاءهای بکار رفته در [۱۰] بود. در زمینه‌ی تشخیص امضاءهای فارسی همچنین می‌توان به کار محرابیان و صفابخش در [۱۳] هم اشاره نمود.

^۱ Smail

^۲ Chain Code

^۳ Fourier Transform

^۴ Principal Component Analysis

^۵ Gabor Wavelet Transform

۳-۱- هدف پایان نامه

هدف این پایان نامه، ارائه‌ی یک الگوریتم مناسب برای تأیید برون خط امضاء با خطای EER قابل قبول می‌باشد.

در این پایان نامه از پایگاه داده GPDS۳۰۰ GRAY در تأیید امضاء برون خط استفاده می‌کنیم. الگوریتم‌های تأیید امضاء باید قابلیت تعمیم بر روی پایگاه داده‌های دیگر را نیز داشته باشند. پس ما برای بررسی این قابلیت در انتها، الگوریتم پیشنهادی را بر روی پایگاه داده GPDS۹۶۰ نیز اجرا می‌کنیم.

۴-۱- ساختار پروژه

همانطور که گفته شد هدف این پایان نامه، ارائه الگوریتمی مناسب در زمینه تأیید امضاء برون خط می‌باشد، به طوری که نرخ خطا بصورت مطلوبی کاهش یابد. لذا پس از معرفی بخش‌های مختلف یک سیستم تأیید امضاء و صحبت در مورد کارهای گذشته در فصل دوم، در فصل سوم به بررسی روش‌های مختلفی که برای انتخاب ویژگی وجود دارند می‌پردازیم و با توجه به انتخاب ویژگی بر مبنای الگوریتم رقابت استعماری، در فصل چهارم این الگوریتم به تفصیل ارائه می‌گردد. در این فصل خواهیم دید که این الگوریتم ابزاری قدرتمند جهت انتخاب ویژگی بهینه می‌باشد و به نظر می‌رسد که در کاربرد جاری هم موثر واقع شود. در فصل پنجم، نتایج سیستم پیشنهادی بر روی پایگاه داده موجود آورده می‌شود. در این فصل ابتدا نتایج حاصل از این الگوریتم با نتایج پیاده‌سازی الگوریتم‌های مشابه، مقایسه می‌گردد. سپس روش پیشنهادی را با نتایج حاصل از تحقیقات دیگری که بر روی این پایگاه داده انجام گرفته شده، مقایسه می‌کنیم. در نهایت در مورد

مزایا و معیب این الگوریتم، صحبت نموده و سپس نتیجه گیری و پیشنهادات برای ادامه کار و بهبود نتایج ارائه می شود.

سیستم تأیید امضاء برون خط

سیستم‌های تأیید امضاء مشابه اغلب سیستم‌های تشخیص الگو، دارای سه فاز عملیاتی هستند. در مرحله نخست داده‌ها، جمع آوری شده و عملیات پیش پردازش بر روی آنها انجام می‌پذیرد و در مرحله بعدی ویژگی‌های مختلف امضای افراد، استخراج شده و در پایگاه داده ذخیره می‌شود. سپس در فاز تصمیم‌گیری، ویژگی‌های استخراج شده‌ی امضای آزمون ورودی با اطلاعات موجود در پایگاه داده‌ها مقایسه می‌شود تا صحت یک امضاء تأیید شود.

۲-۲- جمع آوری داده و عملیات پیش پردازش

در روش برون خط امضاءها بر روی صفحه‌ی کاغذ یا هر سطح دیگری مثلاً صفحه چک یا اشیا (در کاربرد های جنایی) موجود هستند و ابتدا باید توسط اسکنری با رزولوشن مشخص، اسکن شوند. برای سادگی پیشنهاد می‌شود کلیه‌ی نمونه‌ها با یک رزولوشن اسکن شوند تا حجم تصاویر یکسان باشد. اما لزوم عملیات پیش پردازش، معلول عوامل مختلفی از جمله نویزی است که اسکنر تولید می‌کند و مثلاً در رابطه با امضای روی چک‌ها همواره این مشکل وجود دارد که پس زمینه‌ای پیچیده دارند. به علاوه در بیشتر مواقع امضاءکننده‌ها از قلم‌های گوناگون با ضخامت متفاوت نوک قلم و رنگ‌های متفاوت جوهر، امضاء می‌کنند و مسلماً برای یکسان سازی داده‌ها، نیاز به پیش پردازش داریم. با توجه به کاربرد معینی که از تشخیص هویت مورد نظرمان هست و با توجه به روش خاصی که برای حل مسئله به کار می‌بریم، در هر مورد خاص تنها به چند مورد از عملیات پیش پردازش نیاز داریم.

حذف نویز [۱۴]، تبدیل تصویر به تصویری با مقیاس خاکستری [۱۵]، باینری کردن [۱۶]، چرخاندن تصویر [۱۶]، حذف نواحی سفید اضافی [۱۷] و نازک سازی [۱۴] تعدادی از عملیات پیش پردازشی است که در مقالات مختلف استفاده شده است. در شکل (۲-۱) نمونه‌ای از عملیات پیش پردازش مذکور نمایش داده شده است.



شکل ۱-۲ (الف) تبدیل تصویر با سطح خاکستری به تصویر سیاه و سفید (ب) عملیات حذف نویز (ج) حذف قسمت های سفید اضافی

۲-۳- استخراج ویژگی

استخراج ویژگی نقش مهمی را در سیستم‌های تشخیص و تأیید امضاء ایفا می‌کند و به دو دسته سراسری و محلی تقسیم می‌شوند. ویژگی‌های سراسری از تصویر کل امضاء و ویژگی‌های محلی از قسمتی از تصویر امضاء به دست می‌آید [۱۶، ۱۸].

استخراج ویژگی‌هایی پایدار و تغییر ناپذیر، در کارایی سیستم موثر است. یک سیستم تأیید امضاء، در حقیقت یک مسئله کلاس بندی دو کلاسه می‌باشد که مسئله، قرار گرفتن یک امضای آزمون در دسته امضاءهای جعلی است. انتخاب ویژگی‌ها باید تا آنجا که ممکن است به گونه ای باشد که تفاوت درون کلاسی (امضاءهای گوناگون یک فرد)، کم بوده و بر عکس تفاوت بین کلاس‌های امضای جعلی و اصلی برای یک شخص زیاد باشد [۱۹].

۲-۴- روش‌های تصمیم‌گیری

در مسئله‌ی تشخیص صحت امضاء، امضای آزمون با تطبیق ویژگی‌هایش با اطلاعات پایگاه داده‌های موجود، مورد ارزیابی قرار می‌گیرد و پاسخ سیستم یک جواب آری یا خیر است. در حل

مسائل تشخیص و تأیید امضا، روش‌های متنوعی بکار گرفته شده‌اند و افراد مختلف به راه‌های ابتکاری گوناگونی دست زده‌اند. صحبت در مورد روش‌های انجام شده در این زمینه بسیار طولانی خواهد بود. با این حال در فصل بعد مختصری را به کارهای انجام شده در این زمینه می‌پردازیم. صرف نظر از روش‌های ابتکاری، روش‌های تأیید امضای برون خط را می‌توان بطور کلی به چهار دسته تقسیم بندی کرد.

- روش تطبیق الگو
- روش‌های شبکه عصبی
- روش مدل مخفی مارکوف
- روش‌های آماری

۲-۴-۱- روش تطبیق الگو

روش تطبیق الگو، یک روش مقایسه داده است که در آن یک امضای آزمون با الگوهای امضاءهای جعلی و حقیقی تطبیق داده می‌شود [۲۰، ۲۱].

۲-۴-۲- روش‌های شبکه عصبی

دلیل اصلی استفاده گسترده از شبکه‌های عصبی، در توانایی بالا در تشخیص الگو و سادگی کار با آنها نهفته است. روش‌های مختلفی برای آموزش شبکه‌های عصبی وجود دارد اما یک روش بسیار ساده این است که در مرحله اول، مجموعه‌ای مناسب از ویژگی‌های نمونه‌های امضاء، برای امضاء کننده‌های مختلف را استخراج کنیم. در مرحله بعد باید شبکه را طوری آموزش دهیم که رابطه بین یک امضاء و کلاس مربوط به آنرا بفهمد (جعلی یا حقیقی). هر زمان که این رابطه آموزش داده شد، سیستم می‌تواند امضای آزمون را که ادعا شده متعلق به فردی خاص است، مورد ارزیابی قرار دهد.

۲-۴-۳ - روش مدل مخفی مارکوف

مدل مخفی مارکوف^۱ (HMM) یک مدل بسیار مناسب برای تحلیل رشته‌ای در تشخیص صحت امضاء می‌باشد. امضای دستنویس دنباله‌ای از بردارهاست که مقادیرشان با سطوح روشنایی هر نقطه از تصویر متناظر است. بنابراین انتخاب مناسبی از مجموعه بردارهای ویژگی برای HMM می‌تواند منجر به طراحی یک سیستم تشخیص امضای کارآمد شود. این مدل‌ها، مدل‌های آماری هستند که قابلیت جذب شباهت‌ها و تفاوت‌های داده‌ها را دارند. در مدل مخفی مارکوف تطابق‌های آماری بین مدل و امضاء لازم است. این تطابق به کمک توزیع احتمال ویژگی‌های امضاء انجام می‌گیرد. اگر نتایج، احتمال بالاتری را نسبت به احتمال امضاءهای تست نشان دادند، امضاء متعلق به خود فرد است. در غیر این صورت باز گردانده می‌شود. انتخاب ساختاری مناسب برای مدل مخفی مارکوف باعث می‌شود که در مرحله آموزش و تشخیص به نتایج بهتری دست یابیم. ساختارهای گوناگونی برای مدل مخفی مارکوف وجود دارد که هر کدام با مشخصات خاصی سازگارند. برای مدل‌های گسسته دو فاکتور در انتخاب نوع ساختار دخالت دارد. یکی تعداد حالات بکار برده شده و دیگری تعداد گذر بین این حالات [۲۲]. در کنار انتخاب ساختاری برای مدل، تابع چگالی احتمال مدل‌کننده مدل مخفی مارکوف نیز نقشی اساسی در طراحی مدل‌های مناسب برای مسئله تشخیص صحت امضاء، ایفا می‌کند.

۲-۴-۴ - روش‌های آماری

به کمک دانش آمار براحتی می‌توان ارتباط و انحراف دو یا تعداد بیشتری داده را فهمید. بنابراین بمنظور یافتن ارتباط بین چند مجموعه از داده، از ضرایب همبستگی کمک می‌گیریم. در تشخیص صحت یک امضاء به کمک روش‌های آماری، ابتدا امضای میانگین را از مجموعه

^۱ Hidden Markov Model

امضاءهای مرجع به دست می آوریم و سپس برای تشخیص صحت امضای آزمون از تابع همبستگی برای سنجش شباهت‌های این دو امضاء استفاده می‌کنیم [۲۳].

۲-۵- معیار های سنجش عملکرد یک سیستم تشخیص صحت امضا

در این قسمت معیار هایی را که معمولاً برای ارزیابی سیستم تأیید صحت امضاء به کار برده می‌شود، معرفی می‌کنیم. در اکثر تحقیقات در زمینه‌ی تأیید امضاء، دو معیار بیشتر از بقیه مورد استفاده قرار می‌گیرند.

نرخ پذیرش اشتباه^۱ (FAR)، که از تقسیم امضاءهای جعلی که توسط سیستم مورد تأیید قرار می‌گیرند، به تعداد کل امضاءهای امتحان شده، به دست می‌آید.

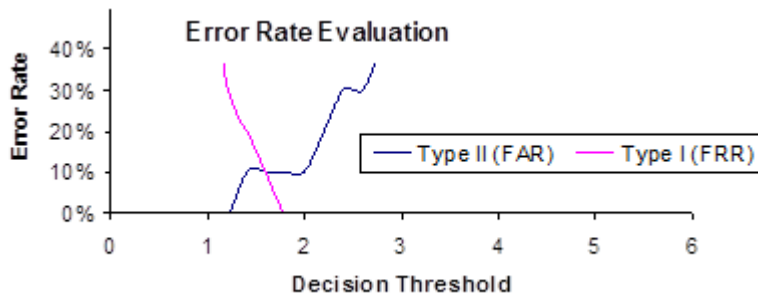
نرخ رد اشتباه^۲ (FRR)، که از تقسیم امضاءهای واقعی که توسط سیستم رد شده‌اند، به کل امضاءهای امتحان شده به دست می‌آید.

این دو معیار همواره خلاف جهت هم حرکت می‌کنند. معمولاً در مسائل، یک متغیر آستانه وجود دارد که قابلیت تنظیم سیستم را در دست دارد. با تغییر این آستانه، مقادیر FAR و FRR تغییر می‌کند و زیاد شدن یکی کاهش دیگری را در پی دارد. اگر در یک سیستم این متغیر آستانه طوری اتخاذ شود که این دو مقدار با هم برابر شود، در این حالت، این مقدار مساوی خطا، نرخ خطای برابر^۳ (EER) گفته می‌شود. در شکل (۲-۲) چگونگی یه دست آوردن EER نشان داده شده است.

^۱ False Acceptable Rate

^۲ False Rejected Rate

^۳ Equal Error Rate



شکل ۲-۲ نمودار تغییرات خطای FAR و FRR به ازای مقادیر مختلف آستانه

۲-۶- بررسی ویژگی‌های گوناگون امضا

در این بخش بطور اجمالی به معرفی برخی از ویژگی‌های امضاء که در تحقیقات مختلف از آنها استفاده شده، می‌پردازیم.

ویژگی‌های امضاء بطور کلی به دو دسته سراسری و محلی تقسیم می‌شوند. ویژگی‌های سراسری از تصویر کل امضاء به دست می‌آید. در حالیکه ویژگی‌های محلی از قسمتی از تصویر امضاء به دست می‌آید. ویژگی‌ها چه محلی باشند و چه سراسری، به دو نوع هندسی و آماری نیز قابل تقسیم هستند [۲۴]. ویژگی‌های هندسی از ساختار امضاء به دست می‌آید و ویژگی‌های آماری به توزیع فراوانی پیکسل‌های امضاء چه بصورت کلی و چه محلی می‌پردازد و از عملیات آماری در نواحی مختلف امضاء استفاده می‌کند.

در این قسمت به معرفی چندین ویژگی مهم در زمینه تأیید صحت امضاء برون خط می‌پردازیم.

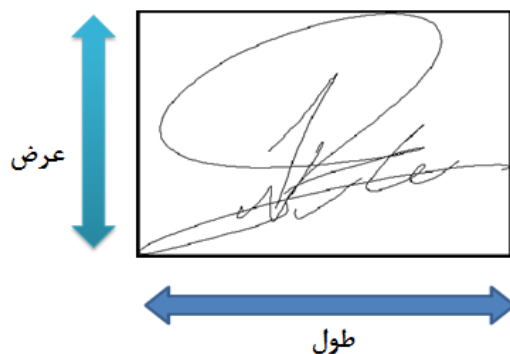
۲-۶-۱- نسبت طول به عرض یک امضا

ابتدا باید عملیات برش را بر روی تصویر انجام دهیم .

طول امضا: فاصله بین دو نقطه‌ی انتهایی در نگاشت افقی یک امضاء که این فاصله باید بیشتر

از ۳ پیکسل باشد.

عرض امضا: فاصله بین دو نقطه‌ی انتهایی در نگاشت عمودی یک امضاء که این فاصله باید بیشتر از ۳ پیکسل باشد. شکل (۲-۳) طول و عرض یک امضا را نشان می‌دهد.



شکل ۲-۲ تعیین طول و عرض یک امضا

ممکن است طول و یا عرض امضاءهای متفاوت یک فرد، همواره یکسان نباشد ولی نسبت این دو، در امضای یک فرد تقریباً عدد ثابتی است و می‌تواند ویژگی خوبی باشد [۲۵].

۲-۶-۲ ناحیه پیکسل‌های سیاه

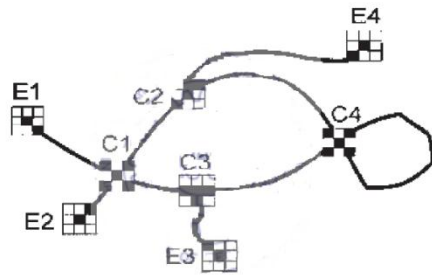
این ویژگی با شمردن پیکسل‌های سیاه در تصویر باینری امضاء به دست می‌آید [۱۸, ۲۳].

۲-۶-۳ ناحیه پیکسل‌های سیاه نرمال شده

این ویژگی از تقسیم ناحیه‌ی پیکسل‌های سیاه بر تعداد کل پیکسل‌های امضاء به دست می‌آید.

۲-۶-۴ تعداد نقاط لبه

نقاط لبه به آن دسته از پیکسل‌هایی گفته می‌شود که تنها یکی از پیکسل‌های همسایگی هشت تایی‌اش سیاه باشد. در شکل (۲-۴)، E_1, E_2, E_3, E_4 ، نقاط و پیکسل‌های لبه ی یک امضاء را نشان می‌دهد. برای این امضاء عدد ۴ که تعداد این نقاط را نشان می‌دهد، می‌تواند به عنوان یک ویژگی در نظر گرفته شود [۲۶].



شکل ۲-۳ نمایش نقاط لبه و نقاط تقاطع یک امضاء

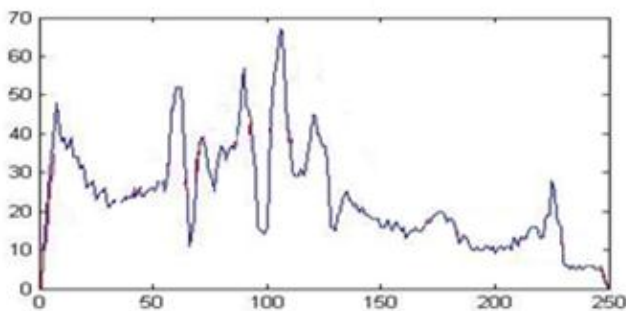
۲-۶-۵- تعداد نقاط تقاطع

نقاط تقاطع به آن دسته از پیکسل‌هایی گفته می‌شود که حداقل ۳ پیکسل از پیکسل‌های همسایگی هشت تایی‌اش سیاه باشد. در شکل (۲-۴)، C_1, C_2, C_3, C_4 نقاط تقاطع یک امضاء را نشان می‌دهد. برای این امضاء باز هم عدد ۴ می‌تواند به عنوان یک ویژگی در نظر گرفته شود [۲۶].

۲-۶-۶- نگاشت افقی^۱

اگر تصویر امضاء را از محور افقی نگاه کنیم و در هر نقطه از این محور تعداد پیکسل‌های سیاه آن ستون را بشماریم، به اندازه‌ی تعداد پیکسل‌های طول یک امضاء، عدد خواهیم داشت که این اعداد می‌توانند به عنوان ویژگی در نظر گرفته شوند [۲۷].

^۱ Horizontal Projection



شکل ۲-۴ نمودار توزیع فراوانی پیکسل‌های سیاه در هر ستون برای نگاشت افقی

شکل (۲-۵) نمودار توزیع پیکسل‌های سیاه، در راستای افق را برای یک امضاء را نشان می‌دهد. به عبارت دیگر محور افقی این نمودار، ویژگی‌ها و محور عمودی اش مقدار ویژگی را نشان می‌دهد. قله این نمودار معرف ستونی می‌باشد که دارای بیشترین پیکسل سیاه است. در شکل (۲-۶) محل قرار گرفتن قله‌ی توزیع فراوانی نشان داده شده است.



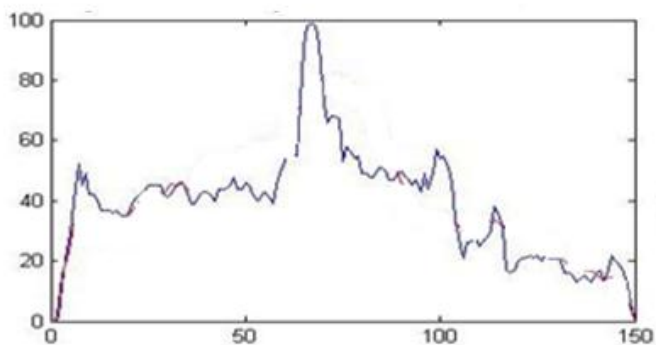
شکل ۲-۵ قله توزیع فراوانی

۲-۶-۷- نگاشت عمودی^۱

همچنین اگر به تصویر امضاء از محور عمودی نگاه کنیم و در هر نقطه از این محور، تعداد پیکسل‌های سیاه آن ردیف را بشماریم، به تعداد ردیف‌ها یعنی به تعداد پیکسل‌های عرض امضاء، ویژگی به دست می‌آوریم [۲۷].

شکل (۲-۶) نمودار توزیع فراوانی پیکسل‌ها در راستای عمودی یک امضاء را نشان می‌دهد.

^۱ Vertical Projection



شکل ۶-۲ نمودار توزیع فراوانی پیکسل‌های سیاه در هر ردیف برای نگاشت عمودی

۲-۶-۸- تعداد حلقه های موجود در یک امضا

تعداد حلقه های موجود در یک امضا، می‌تواند به عنوان ویژگی خوبی برای امضاء در نظر گرفته شود. حلقه می‌تواند بصورت ناحیه ای متصل که توسط پیکسل‌های امضاء احاطه شده است، تعریف شود [۲۷]. اگر در فرهنگی رسم بر این باشد که شکل امضاء بیشتر همانند نام نویسنده باشد، این ویژگی دیگر مناسب نیست بدین دلیل که دامنه تغییراتش کم است.



شکل ۷-۲ حلقه های موجود در امضاهای مختلف

در شکل (۲-۸) تعداد حلقه‌های موجود در امضای سمت راستی ۱۵ و برای امضای سمت چپی ۴ می‌باشد.

۲-۶-۹- خط اصلی ۱

خط اصلی در امضاء به دو خط اصلی بالایی و پایینی تقسیم می‌شود و بدینسان تعریف می‌شود که اگر در تصویر باینری امضاء در هر ستون، پایینی ترین (بالایی ترین) پیکسل سیاه آن ستون را نگه داریم و پیکسل‌های سیاه باقیمانده آن ستون را سفید کنیم، خط اصلی پایینی (بالایی) یک امضاء به شکل یک منحنی ظاهر می‌شود. خط اصلی پایینی برای یک امضا را می‌توانید در شکل (۹-۲) مشاهده کنید.

می‌توانیم اطلاعاتی نظیر تعداد پیکسل‌های سیاه این منحنی را استخراج کنیم و بعنوان ویژگی از آنها استفاده نماییم.



شکل ۸-۲ خط اصلی پایینی

۲-۶-۱۰- ویژگی‌های مربوط به مرکز ثقل امضا

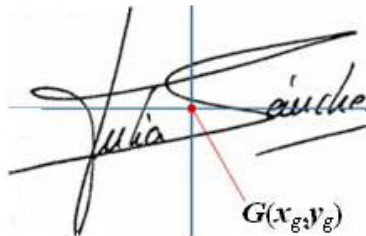
مختصات مرکز ثقل تصویر یک امضاء از روابط (۱-۲) و (۲-۲) به دست می‌آید.

$$x = \frac{\sum_{y=1}^m (y \cdot P_h(y))}{\sum_{y=1}^M P_h(y)} \quad (1-2)$$

$$y = \frac{\sum_{x=1}^n (x \cdot P_v(x))}{\sum_{x=1}^M P_v(x)} \quad (2-2)$$

^۱ Base Line

x و y مختصات عرض و طول هر پیکسل و P_h و P_v همان نگاشت های افقی و عمودی هستند که در پیشتر به آنها اشاره شد [۲۸]. در شکل (۲-۱۰) مرکز ثقل یک امضاء مشخص شده است.



شکل ۲-۹ مرکز ثقل یک امضا

مختصات مرکز ثقل یک امضاء می‌تواند به عنوان یک ویژگی در نظر گرفته شود [۳۰, ۲۹, ۲۶, ۱۴]. اما ویژگی‌های دیگری نیز بر مبنای مرکز ثقل وجود دارد. این ویژگی‌ها به شکل های مختلف در تحقیقات استفاده شده‌اند [۳۰, ۲۹, ۱۶]، که ما در اینجا نمونه ای ساده از آنها را معرفی می‌کنیم.

۲-۶-۱۱- نقاط ویژگی بر اساس انشعاب دادن بصورت عمودی

در استخراج این ویژگی‌ها ابتدا مرکز ثقل تصویر امضاء، V را پیدا می‌کنیم. عملیات برای به دست آوردن نقاط ویژگی مانند زیر می‌باشد [۳۰]:

الگوریتم:

ورودی: تصویر استاتیکی امضاء بعد از مراحل پیش پردازش

خروجی: ویژگی‌های عمودی v_1, v_2, \dots

تصویر امضاء را با عبور دادن خطی عمودی که از مرکز ثقل می‌گذرد، مطابق شکل (۲-۱۱)، به دو قسمت تقسیم می‌کنیم.



شکل ۱۰-۲ تقسیم بندی به کمک مرکز ثقل

مراکز ثقل ۷۱ و ۷۲ را برای نواحی به دست آمده محاسبه می کنیم.

هر دو بخش چپ و راست تقسیم شده را با گذراندن خط های افقی که از ۷۱ و ۷۲ می گذرند به دو بخش دیگر تقسیم می کنیم و حال ما تصویر را به ۴ قسمت تقسیم کردیم. برای هر کدام از قسمت های شمال غربی، جنوب غربی، شمال شرقی و جنوب شرقی، دوباره مراکز ثقل را محاسبه می کنیم و ۷۳, ۷۴, ۷۵, ۷۶ به دست می آید.

سپس مراحل تقسیم بندی بالا را برای هر کدام از ۴ بخش تکرار می کنیم. ۶ نقطه جدید برای هر ناحیه و در مجموع ۲۴ مرکز ثقل دیگر به دست می آید که در کل، ۷۱ تا ۷۳۰ حاصل می شود. می توانیم باز هم انشعاب دادن را تکرار کنیم تا ویژگی های بیشتری داشته باشیم.

۷۱ تا ۷۳۰ می تواند به عنوان بردار ویژگی استفاده شود.

به همین شکل می توانیم مجموعه دیگری از ویژگی ها را بر اساس انشعاب دادن بصورت عمودی به دست آوریم.

۲-۶-۱۲- ویژگی های محلی یا شبکه ای^۱

در مقابل ویژگی های سراسری که از کل یک امضاء به دست می آیند، برای به دست آوردن ویژگی های محلی، امضاء را به نواحی متعددی تقسیم می کنند و استخراج ویژگی، اینبار برای هر

^۱ Grid Base Feature

بخش بصورت مجزا، در نظر گرفته می‌شود. به این نوع ویژگی، ویژگی شبکه‌ای نیز گفته می‌شود. معمولاً در سیستم‌های تأیید امضاء، ترکیبی از این دو نوع ویژگی را بکار می‌برند.

اما برای به دست آوردن ویژگی‌های شبکه‌ای، امضاء را می‌توان بصورت‌های متعدد، ناحیه بندی کرد. در زیر به رایج ترین روش‌ها برای تقسیم بندی امضاء می‌پردازیم.

• ناحیه بندی ساده

در این روش امضاء بصورت ساده به چندین ناحیه تقسیم می‌شود و برای هر ناحیه ویژگی‌های در نظر گرفته شده محاسبه می‌شوند.

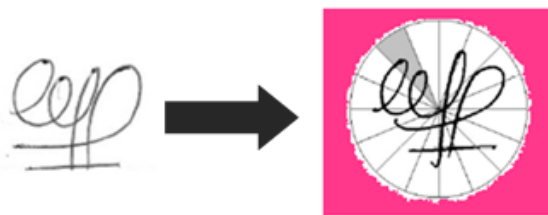


شکل ۱۱-۲ مثالی از ناحیه بندی بصورت ساده

در این مثال، ابتدا تصویر امضاء به ۶۰ ناحیه تقسیم شده و میانگین روشنایی پیکسل‌های آن ناحیه بعنوان ویژگی نماینده برای آن ناحیه در نظر گرفته شده است. شکل (۱۲-۲) مثالی از تقسیم بندی امضاء بصورت ساده است که سطح خاکستری میانگین برای هر ناحیه را با رنگ متناظر با آن نشان می‌دهد [۱۴].

• ناحیه بندی دایره ای

در این روش ابتدا یک دایره بر روی تصویر امضاء، طوری محاط می‌شود که مرکز دایره بر مرکز ثقل تصویر امضاء منطبق شود. سپس این دایره، امضاء را به قطاع‌های مساوی تقسیم می‌کند. و ویژگی‌های مورد نیاز از هر ناحیه استخراج می‌شود. نمونه‌ای از تقسیم بندی دایره‌ای در شکل (۲-۱۳) نشان داده شده است.



شکل ۲-۱۲ مثالی از ناحیه بندی دایره ای

۲-۷- مجموعه امضاءهای مورد استفاده در این پایان نامه

در این پایان نامه، ابتدا از پایگاه داده GPDS۳۰۰GRAY برای ارزیابی الگوریتم پیشنهادی استفاده شده است. سپس برای مقایسه این الگوریتم با سیستم‌های تیم های شرکت کننده در مسابقه ۲۰۱۰NSigComp، از مجموعه‌ی امضاء gpds۹۶۰ استفاده کردیم که در پیوست ب در مورد آن توضیح لازم ارائه شده است.

پایگاه داده GPDS۳۰۰GRAY که توسط گروهی با نام مستعار گروه پردازش سیگنال دیجیتال^۱ (SGPD) جمع آوری شده است، متشکل از امضای ۳۰۰ نفر می‌باشد. تصاویر امضاء، با دقت ۳۰۰ نقطه در اینچ و در اندازه ۴ cm* ۶,۳ cm اسکن شده‌اند. تعداد امضاءهای اصلی برای هر فرد در این پایگاه داده، برابر با ۲۴ و تعداد امضاءهای جعلی ۳۰ عدد می‌باشد. در جمع آوری امضاءهای جعلی تنها از جعل حرفه ای استفاده شده است.

پس بطور کلی این مجموعه شامل $۷۲۰۰ = ۲۴ * ۳۰۰$ امضای اصلی و $۹۰۰۰ = ۳۰ * ۳۰۰$ امضای جعلی با جعل حرفه ای می‌باشد. مولفین، اجازه دسترسی این پایگاه داده را با قبول تعهدات لازم، به عموم می‌دهند [۳۱]. لازم به ذکر است که این پایگاه داده، روز به روز کاملتر می‌شود و از امضای امضاء کنندگان بیشتری استفاده می‌گردد. عدد موجود در نام این پایگاه داده، معرف تعداد افرادیست که امضای آنها در این زیرمجموعه، استفاده شده است. جمع آوری امضاء برای هر فرد در

^۱ Grupo de Procesado Digital de Sennales

یک جلسه صورت می‌گیرد. به علاوه در جمع آوری امضاءها، هم جاعلین و هم افراد حقیقی اجازه دارند از قلم خود برای امضاء کردن، استفاده نمایند.

نمونه‌ای از امضاءهای این پایگاه داده در شکل (۲-۱۴) آورده شده است.



شکل ۲-۱۳ شکل هندسی چند نمونه از مجموعه امضاءهای GPDS۳۰۰GRAY

انتخاب ویزگی

۳-۱- مقدمه

مسئله انتخاب ویژگی، یکی از مسائلی است که در مبحث یادگیری ماشین و همچنین شناسایی آماری الگو مطرح است. این مسئله در بسیاری از کاربردها مانند طبقه بندی اهمیت بسزائی دارد. زیرا در این کاربردها، تعداد زیادی ویژگی وجود دارد که بسیاری از آنها یا بلا استفاده هستند و یا اینکه بار اطلاعاتی چندانی ندارند. حذف نکردن این ویژگی‌ها مشکلی از لحاظ اطلاعاتی ایجاد نمی‌کند ولی بار محاسباتی را برای کاربرد مورد نظر بالا می‌برد و علاوه بر این، باعث می‌شود که اطلاعات غیر مفید زیادی را به همراه داده‌های مفید ذخیره کنیم.

برای مسئله انتخاب ویژگی، راه حل‌ها و الگوریتم‌های فراوانی ارائه شده است که بعضی از آنها قدمت سی یا چهل ساله دارند. مشکل بعضی از الگوریتم‌ها، در زمانی که ارائه شده بودند، بار محاسباتی زیاد آنها بود. اگرچه امروزه با ظهور کامپیوترهای سریع و منابع ذخیره سازی بزرگ، این مشکل به چشم نمی‌آید ولی از طرف دیگر، مجموعه‌های داده‌ای بسیار بزرگ، برای مسائل جدید باعث شده است که همچنان پیدا کردن یک الگوریتم سریع برای این کار مهم باشد.

در این فصل ما در ابتدا تعاریفی را که برای انتخاب ویژگی ارائه شده‌اند، ارائه می‌دهیم. سپس روش‌های مختلف برای این مسئله را بر اساس نوع و ترتیب تولید زیرمجموعه ویژگی‌های کاندید و همچنین نحوه ارزیابی این زیرمجموعه‌ها را بررسی می‌کنیم.

۳-۲- تعاریف

مسئله انتخاب ویژگی بوسیله نویسندگان مختلف، از دیدگاه‌های متفاوتی مورد بررسی قرار گرفته است. هر نویسنده نیز با توجه به نوع کاربرد، تعریفی را از آن ارائه داده است. در ادامه چند مورد از این تعاریف را بیان می‌کنیم [۳۲].

- **تعریف ایده‌آل:** پیدا کردن یک زیرمجموعه با حداقل اندازه ممکن برای ویژگی‌ها است که برای هدف مورد نظر، اطلاعات لازم و کافی را در بر داشته باشد. بدیهی است که هدف تمام الگوریتم‌ها و روش‌های انتخاب ویژگی، زیر مجموعه‌ای از همین تعریف است.
 - **تعریف کلاسیک:** انتخاب یک زیرمجموعه M عنصری از میان N ویژگی، به طوریکه $M < N$ باشد و همچنین مقدار یک تابع معیار^۱ برای زیرمجموعه مورد نظر نسبت به سایر زیرمجموعه‌های هم اندازه دیگر، بهینه باشد. این تعریفی است که فوکوناگا^۲ و نارندا^۳ در سال ۱۹۷۷ ارائه داده‌اند.
 - **افزایش دقت پیشگویی:** هدف انتخاب ویژگی، این است که یک زیرمجموعه از ویژگی‌ها برای افزایش دقت پیشگویی انتخاب شوند. به عبارت دیگر کاهش اندازه ساختار، بدون افزایش قابل ملاحظه در دقت پیشگویی طبقه‌بندی کننده‌ای که با استفاده از ویژگی‌های داده شده بدست می‌آید، کار بهبود یافته است.
 - **تخمین توزیع کلاس اصلی:** هدف از انتخاب ویژگی این است که یک زیرمجموعه کوچک از ویژگی‌ها انتخاب شوند. توزیع ویژگی‌هایی که انتخاب می‌شوند، بایستی تا حد امکان به توزیع کلاس اصلی با توجه به تمام مقادیر ویژگی‌های انتخاب شده نزدیک باشد.
- در روش‌های مختلف انتخاب ویژگی، تلاش می‌شود تا از میان 2^N زیر مجموعه ممکن، بهترین زیرمجموعه را پیدا کنند. در تمام این روشها بر اساس کاربرد و نوع تعریف، زیر مجموعه‌ای به عنوان جواب انتخاب می‌شود که بتواند مقدار یک تابع ارزیابی را بهینه کند.

^۱ Criterion Function

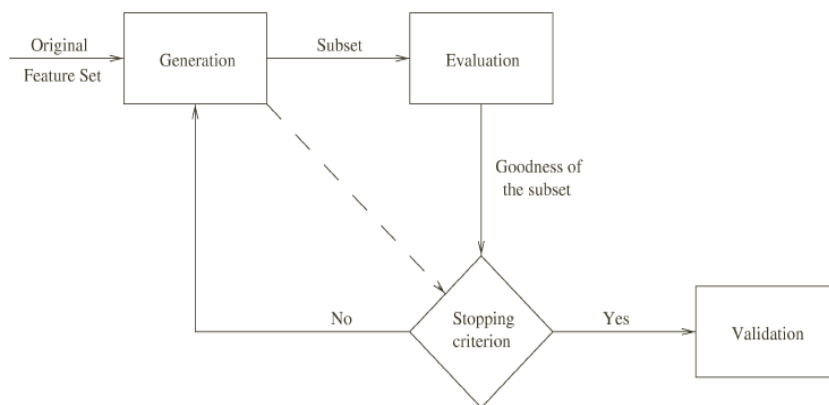
^۲ Fukunaga

^۳ Narendra

برای اینکه بتوانیم تقسیم بندی درستی از روش‌های مختلف انتخاب ویژگی داشته باشیم، به این صورت عمل می‌کنیم که فرآیند انتخاب ویژگی در تمامی روش‌ها را به قسمت‌های متفاوت، تقسیم می‌کنیم.

۱. تابع تولید کننده: این تابع زیر مجموعه کاندید را برای روش مورد نظر پیدا می‌کند.
۲. تابع ارزیابی^۱: زیرمجموعه به دست آمده توسط روش مورد نظر را، ارزیابی و یک عدد را به عنوان میزان کارایی روش باز می‌گرداند. روش‌های مختلف سعی در یافتن زیرمجموعه‌ای دارند که این مقدار را بهینه کند.
۳. شرط خاتمه: برای تصمیم‌گیری در مورد زمان توقف الگوریتم، به کار برده می‌شود.
۴. تابع تعیین اعتبار^۲: تصمیم می‌گیرد که آیا زیر مجموعه انتخاب شده معتبر است یا خیر.

شکل (۳-۱) فرآیند انتخاب ویژگی را در الگوریتم‌های بهینه سازی نشان می‌دهد.



شکل ۳-۱ فرآیند انتخاب ویژگی [۳۲]

^۱ Evaluation Function

^۲ Validation Procedure

تابع تولید کننده در واقع تابع جستجو است. این تابع زیرمجموعه‌های مختلف را به ترتیب تولید می‌کند، تا بوسیله تابع ارزیابی، سنجش صورت گیرد. تابع تولید کننده از یکی از حالت‌های زیر شروع به کار می‌کند.

- بدون ویژگی
- با تمام ویژگی‌ها
- با یک زیرمجموعه تصادفی

در حالت اول ویژگی‌ها به ترتیب به مجموعه اضافه می‌شوند و زیرمجموعه‌های جدید را تولید می‌کنند. این عمل آنقدر تکرار می‌شود تا به زیر مجموعه مورد نظر برسیم. به اینگونه روش‌ها، روشهای پائین به بالا^۱ می‌گویند.

در حالت دوم از یک مجموعه شامل تمام ویژگی‌ها، شروع می‌کنیم و به مرور و در طی اجرای الگوریتم، ویژگی‌ها را حذف می‌کنیم، تا به زیرمجموعه دلخواه برسیم. روش‌هایی که به این صورت عمل می‌کنند، روشهای بالا به پائین^۲ نام دارند.

یک تابع ارزیابی، میزان خوب بودن زیرمجموعه تولید شده را بررسی کرده و یک مقدار را به عنوان میزان مفید بودن زیرمجموعه مورد نظر بازمی‌گرداند. این مقدار با بهترین زیرمجموعه قبلی مقایسه می‌شود. اگر زیرمجموعه جدید، بهتر از زیرمجموعه‌های قدیمی باشد، بعنوان زیرمجموعه بهینه، جایگزین قبلی می‌شود.

باید توجه داشت که بدون داشتن یک شرط خاتمه مناسب، فرآیند انتخاب ویژگی ممکن است برای همیشه، درون فضای جستجو، در یافتن جواب سرگردان بماند.

^۱ bottom-up

^۲ top-down

شرط خاتمه می‌تواند بر پایه تابع تولید کننده باشد:

- هر زمان که تعداد مشخصی ویژگی انتخاب شدند.
- هر زمان که به تعداد مشخصی تکرار رسیدیم.

و یا اینکه بر اساس تابع ارزیابی انتخاب شود:

- وقتیکه اضافه یا حذف کردن ویژگی، زیرمجموعه بهتری را تولید نکند
- وقتیکه به یک زیرمجموعه بهینه بر اساس تابع ارزیابی برسیم.

تابع تعیین اعتبار، جزئی از فرآیند انتخاب ویژگی نیست، اما در عمل بایستی یک زیرمجموعه ویژگی را در حالت های مختلف آزمایش کنیم تا ببینیم که آیا شرایط مورد نیاز برای حل مسئله مورد نظر ما را دارد یا نه. برای این کار می‌توانیم از داده‌های نمونه‌برداری شده و یا مجموعه داده‌های شبیه سازی شده استفاده کنیم.

۳-۳- روش‌های مختلف انتخاب ویژگی

در این بخش ابتدا روش‌های مختلف انتخاب ویژگی را بر اساس دو معیار تابع تولید کننده و تابع ارزیابی، طبقه بندی می‌کنیم. سپس آنها را بر اساس عملکرد، دسته‌بندی و نحوه اجرای هر دسته را به اختصار شرح می‌دهیم.

۳-۳-۱- توابع تولید کننده

اگر تعداد کل ویژگی‌ها برابر با N باشد، تعداد کل زیرمجموعه‌های ممکن برابر با 2^N می‌شود. این تعداد حتی برای N های متوسط هم خیلی زیاد است. بر اساس نحوه جستجو در میان این تعداد زیر مجموعه، روشهای مختلف انتخاب ویژگی را می‌توان به سه دسته زیر تقسیم‌بندی نمود:

- جستجوی کامل
- جستجوی مکاشفه‌ای

• جستجوی تصادفی

در ادامه به معرفی هر کدام از این دسته‌ها می‌پردازیم

۳-۱-۱- جستجوی کامل

در روش‌هایی که از این نوع جستجو استفاده می‌کنند، تابع تولید کننده بر اساس تابع ارزیابی استفاده شده، تمام فضای جواب یعنی تمام زیرمجموعه‌های ممکن را برای یافتن جواب بهینه جستجو می‌کند. البته، اسکلیمر^۱ در [۱۶] استدلال کرده است که: "کامل بودن جستجو به این معنی نیست که جستجو باید جامع باشد".

توابع مکاشفه‌ای مختلف زیادی طراحی شده‌اند، تا جستجو را بدون از دست دادن شانس پیدا کردن جواب بهینه، کاهش دهند. با توجه به بزرگی فضای جستجو، این روش‌ها باعث می‌شوند که فضای کمتری جستجو شود. روش‌ها و تکنیک‌های مختلفی برای اینکار استفاده شده‌اند، بعضی از آنها از تکنیک بازگشت به عقب^۲ نیز در جریان کار استفاده کرده‌اند.

۳-۱-۲- جستجوی مکاشفه‌ای^۳

در روش‌های با این نوع جستجو، در هر بار اجرای الگوریتم، یک ویژگی به مجموعه ویژگی انتخاب شده، اضافه و یا از آن حذف می‌شود. به همین دلیل پیچیدگی زمانی آنها محدود و کمتر از $O(N^2)$ می‌باشد. در اینگونه موارد، اجرای الگوریتم خیلی سریع و پیاده سازی آنها نیز بسیار ساده است.

^۱ Schlimmer

^۲ Backtracking

^۳ Heuristic Search

۳-۱-۳-۳ - جستجوی تصادفی^۱

روش‌های بر اساس جستجوی تصادفی، محدوده‌ی کمتری از فضای کل حالات را جستجو می‌کنند. اندازه این محدوده به حداکثر تعداد تکرار الگوریتم، بستگی دارد. در این روش‌ها پیدا شدن جواب بهینه، به اندازه منابع موجود و زمان اجرای الگوریتم بستگی دارد. در هر بار تکرار، تابع تولید کننده تعدادی از زیرمجموعه‌های ممکن از فضای جستجو را به صورت تصادفی انتخاب می‌کند و در اختیار تابع ارزیابی قرار می‌دهد. تابع تولید کننده تصادفی پارامترهایی دارد که بایستی تنظیم شوند، تنظیم مناسب این پارامترها در سرعت رسیدن به جواب و پیدا شدن جواب‌های بهتر موثر است.

۳-۲-۳ - تابع ارزیابی

پیدا شدن یک زیرمجموعه‌ی بهینه از مجموعه ویژگی‌ها، به صورت مستقیم به انتخاب تابع ارزیابی بستگی دارد. این بدان علت می‌باشد که اگر تابع ارزیابی به زیرمجموعه ویژگی بهینه، یک مقدار نامناسب نسبت دهد، این زیرمجموعه هیچگاه بعنوان زیرمجموعه بهینه انتخاب نمی‌شود. مقادیری که توابع ارزیابی مختلف به یک زیرمجموعه می‌دهند، با هم متفاوت است.

توابع ارزیابی را می‌توان به روش‌های مختلف دسته بندی کرد. در اینجا ما دسته بندی را که توسط داش^۲ و لیو^۳ برای توابع ارزیابی ارائه شده است را بیان می‌کنیم [۱۵]. آنها این معیارها را به پنج دست تقسیم کرده‌اند.

معیارهای مبتنی بر فاصله^۱: در این معیارها، مثلاً برای یک مسئله دو کلاسه، یک ویژگی یا یک مجموعه ویژگی مثل X بر یک ویژگی یا یک مجموعه ویژگی دیگر مثل Y ارجحیت دارد، اگر

^۱ Random Search

^۲ Dash

^۳ Liu

که با آن مجموعه ویژگی مقادیر بزرگتری برای اختلاف بین احتمالات شرطی دو کلاس داشته باشیم. نمونه‌ای از این معیارها، همان معیار فاصله اقلیدسی می‌باشد.

معیارهای مبتنی بر اطلاعات^۲: این معیارها میزان اطلاعاتی را که بوسیله یک ویژگی بدست می‌آید، در نظر می‌گیرند. ویژگی X در این روش‌ها بر ویژگی Y اولویت دارد، اگر اطلاعات بدست آمده از ویژگی X بیشتر از اطلاعاتی باشد که از ویژگی Y بدست می‌آید. نمونه‌ای از این معیارها همان معیار آنتروپی می‌باشد.

معیارهای مبتنی بر وابستگی^۳: این معیارها که با عنوان معیارهای همبستگی^۴ نیز شناخته می‌شوند، قابلیت پیشگویی مقدار یک متغیر بوسیله یک متغیر دیگر را اندازه‌گیری می‌کنند. ضریب وابستگی^۵، یکی از معیارهای وابستگی کلاسیک است و می‌توانیم آنرا برای یافتن همبستگی بین یک ویژگی و یک کلاس به کار ببریم. اگر همبستگی ویژگی X با کلاس C بیشتر از همبستگی ویژگی Y با کلاس C باشد، در اینصورت ویژگی X بر ویژگی Y برتری دارد. با یک تغییر کوچک، می‌توانیم وابستگی یک ویژگی با ویژگی‌های دیگر را اندازه‌گیری کنیم. این مقدار درجه افزونگی این ویژگی را نشان می‌دهد. همه توابع ارزیابی بر پایه معیار وابستگی را می‌توانیم بین دو دسته معیارهای مبتنی بر فاصله و اطلاعات تقسیم کنیم. اما به خاطر اینکه این روش‌ها از یک دید دیگر به مسئله نگاه می‌کنند، این کار را انجام نمی‌دهیم.

^۱ Distance Measures

^۲ Information Measures

^۳ Dependence Measures

^۴ Correlation

^۵ Coefficient

معیارهای مبتنی بر سازگاری^۱: این معیارها جدیدتر هستند و اخیراً توجه بیشتری به آنها شده است. این معیارها خصوصیات متفاوتی نسبت به سایر معیارها دارند، زیرا که به شدت به داده-های آموزشی تکیه دارند و در انتخاب یک زیرمجموعه از ویژگی‌ها تمایل دارند که مجموعه ویژگی‌های کوچکتری را انتخاب کنند. این روش‌ها، زیرمجموعه‌های با کمترین اندازه را بر اساس از دست دادن یک مقدار قابل قبول سازگاری که توسط کاربر تعیین می‌شود، پیدا می‌کنند.

معیارهای مبتنی بر خطای طبقه بند^۲: روش‌هایی که این نوع از تابع ارزیابی را استفاده می‌کنند با عنوان "wrapper methods" شناخته می‌شوند. دقت عملکرد در این روش‌ها برای تعیین کلاسی که نمونه داده شده، متعلق به آن است، برای نمونه‌های دیده نشده بسیار بالا است، اما هزینه‌های محاسباتی در آنها نیز نسبتاً زیاد است.

در جدول (۱-۳) مقایسه‌ای بین انواع مختلف تابع ارزیابی، صرف نظر از نوع تابع تولید کننده مورد استفاده، انجام شده است. پارامترهایی که برای مقایسه استفاده شده‌اند به صورت زیر می‌باشند.

- **عمومیت^۳:** اینکه بتوان زیرمجموعه انتخاب شده را برای طبقه‌بند های متفاوت به کار ببریم.

- **پیچیدگی زمانی:** زمان لازم برای پیدا کردن زیرمجموعه ویژگی جواب.

- **دقت:** دقت پیشگویی با استفاده از زیرمجموعه انتخاب شده.

علامت "----" که در ستون آخر آمده است، به این معنی است که در مورد میزان دقت حاصل، نمی‌توانیم مطلبی بگوئیم. به جز خطای طبقه‌بندی، دقت سایر توابع ارزیابی، به مجموعه داده‌ی

^۱ Consistency Measures

^۲ Classifier Error Rate Measures

^۳ Generality

مورد استفاده و کلاسه بندی که بعد از انتخاب ویژگی برای جداسازی کلاس‌ها استفاده می‌شود، بستگی دارد.

جدول ۱-۳ مقایسه توابع ارزیابی مختلف

دقت	پیچیدگی زمانی	عمومیت	نوع تابع ارزیابی
---	پائین	دارد	معیار فاصله
---	پائین	دارد	معیار اطلاعات
---	پائین	دارد	معیار وابستگی
---	متوسط	دارد	معیار سازگاری
خیلی زیاد	بالا	ندارد	خطای طبقه بندی

الگوریتم رقابت استعماری

۴-۱- مقدمه

در این فصل، الگوریتم جدیدی برای جستجوی عام معرفی می شود که از مدلسازی ریاضی رقابت های امپریالیستی الهام گرفته شده است. بطور خلاصه این الگوریتم از چندین کشور در حالت اولیه شروع میشود. کشورها، در حقیقت جوابهای ممکن مسئله هستند و معادل کروموزوم^۱ در الگوریتم ژنتیک^۲ (GA) و ذره در الگوریتم بهینه سازی گروه ذرات^۳ (PSO) هستند. همه کشورها به دو گروه امپریالیست و مستعمره تقسیم می شوند. کشورهای استعمارگر با اعمال سیاست جذب (همگون سازی) در راستای محورهای مختلف بهینه سازی، کشورهای مستعمره را به سمت خود می کشند. امپریالیسم در کنار سیاست همگون سازی، هسته اصلی این الگوریتم را تشکیل می دهند و باعث می شود کشورها به سمت مینیمم مطلق تابع حرکت کنند. در این فصل به استعمار به عنوان جزو لاینفک سیر تکامل تاریخی انسان نگریسته شده و از چگونگی اثر گذاری آن بر کشورهای استعمارگر و مستعمره در کل تاریخ، به عنوان منبع الهام یک الگوریتم کارا و نو در زمینه محاسبات تکاملی استفاده شده است.

۴-۲- مروری تاریخی بر پدیده استعمار

امپریالیسم^۴ در لغت به سیاست توسعه قدرت و نفوذ یک کشور در حوزه خارج از قلمرو شناخته شده برای آن، اطلاق می شود. یک کشور می تواند کشور دیگر را از طریق قانونگذاری به طور مستقیم، یا با اعمالی نظیر کنترل کالاها و مواد خام به طور غیر مستقیم، کنترل کند. مورد اخیر استعمار نو خوانده می شود [۳۳]. استعمار یک پدیده ذاتی در تاریخ می باشد که در مراحل ابتدایی

^۱ Chromosome

^۲ Genetic Algorithm

^۳ Particle Swarm Optimization Algorithm

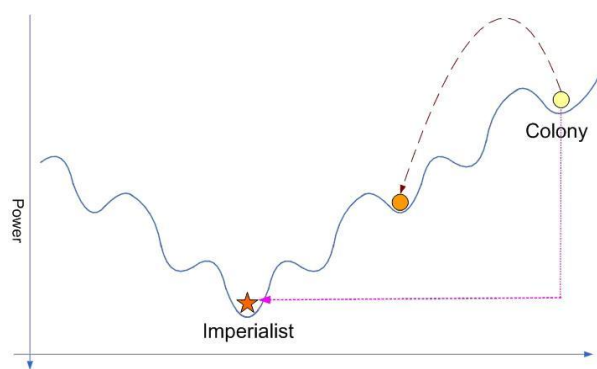
^۴ Imperialism

به صورت نفوذ سیاسی - نظامی در کشورها و به منظور استفاده از منابع زمینی، انسانی و سیاسی بوده است. بعضی مواقع نیز استعمار به دلیل جلوگیری از نفوذ کشور استعمارگر رقیب انجام می‌شد. به هر حال کشورهای استعمارگر رقابت شدیدی را برای به استعمار کشیدن مستعمرات یکدیگر نشان می‌دادند. این رقابت به نوبه خود باعث رشد و توسعه کشورهای استعمارگر از لحاظ سیاسی، نظامی و اقتصادی گردید. زیرا کشورها برای داشتن امکان رقابت، مجبور به توسعه بودند.

در حالت‌های قدیمی تر، استعمارگران با بهره‌گیری از منابع زمینی، انسانی و غیره‌ی کشور مستعمره، فقط در صدد افزایش قدرت خود بودند و اینکه آیا مستعمرات پیشرفت می‌کنند یا نه برایشان مهم نبود. اما بعدها با افزایش ارتباط میان ملل و رشد انسانی، استعمارگران برای ادامه نفوذ خود، به نوعی از اقبال عمومی و حمایت مردمی نیز نیاز پیدا کردند.

بدین ترتیب کشورهای استعمارگر شروع به ایجاد عمران و آبادانی هر چند ظاهری، در مستعمراتشان نمودند و مستعمرات شاهد پیشرفت در زمینه‌های اقتصادی، اجتماعی و انسانی شدند که عامل این پیشرفت به اجبار، کشور استعمارگر بود. دلیل نامگذاری این فرایند با نام "استعمار" که ریشه در کلمه عمران و آبادانی دارد نیز همین مسئله می‌باشد. البته دریافت اقبال عمومی تنها دلیل ایجاد عمران توسط استعمارگران در مستعمرات نبود. دلیل دیگر ایجاد سلطه فرهنگی بر مستعمرات در راستای اجرای سیاست همگون‌سازی بود. به عنوان مثال کشورهای نظیر فرانسه و انگلیس به ایجاد مدارس انگلیسی زبان و فرانسوی زبان در مستعمرات خود پرداختند. این اقدام به دلایل مختلفی صورت می‌گرفت که در راس آن، افزایش نفوذ فرهنگی در مستعمرات بوده است. ناگفته نماند که فرایند استعمار حداقل از بعد فرهنگی، با همه تبعات منفی آن، در بعضی از کشورهای امپریالیست به چشم یک جهاد فکری برای نجات بشر نیز نگریسته می‌شد. امپریالیسم، نگرش عمومی نسبت به تمدن غرب را تغییر داد. داروینیست‌های اجتماعی، امپریالیسم را تفسیر کرده و این ایده که فرهنگ غرب نسبت به فرهنگ شرق برتر است را تقویت کردند [۳۴]. در آفریقا تنها آنهایی که بعضی از استانداردهای فرهنگی غرب را داشتند، دارای بخشی از حقوق اجتماعی

خود بودند. پرتغالی‌ها، این مردم را جذب شده و فرانسوی‌ها بطور توهین آمیزی آن‌ها را تکامل یافته می‌نامیدند. به هر حال مستقل از اثرات و تبعات مثبت و منفی آن، استعمار به عنوان یک فرایند ذاتی در تاریخ بشر ایجاد شد و در عین وارد کردن خسارتهای جبران ناپذیر به زیربناهای اساسی یک کشور خصوصاً زیربناهای فرهنگی، در بعضی موارد اثرات مثبتی را نیز برای کشورها مستعمره داشت. از دید بهینه سازی، استعمار بعضی از کشورها را که در یک دره معمولی تمدن قرار داشتند، خارج کرده و آنها را به یک حوزه مینیمم دیگر برد که در بعضی موارد وضعیت این حوزه مینیمم بهتر از موقعیت قبلی کشور مستعمره بود. اما به هر حال این حرکت مستلزم پیشروی مستعمره در راستای محورهای مختلف اقتصادی و فرهنگی به سمت یک امپریالیست قویتر بود (یعنی از میان رفتن بعضی از ساختارهای فرهنگی و اجتماعی). شکل (۴-۱) این وضعیت را به خوبی نشان می‌دهد. در این شکل، مستعمره در نتیجه سیاست همگونسازی از یک ناحیه مینیمم خارج شده و وارد یک ناحیه مینیمم دیگر می‌شود که در آن وضعیت بهتری را دارا می‌باشد. به هر حال هزینه‌ای که بابت این حرکت پرداخت شده است، نزدیکی به کشور استعمارگر در راستای محورهای مختلف اقتصادی، سیاسی و اجتماعی می‌باشد. ادامه این حرکت می‌تواند به جذب کامل کشور مستعمره توسط کشور استعمارگر، بیانجامد.



شکل ۴-۱ اعمال سیاست جذب از طرف استعمارگران بر مستعمرات

۴-۳- الگوریتم رقابت استعماری

همانند دیگر الگوریتم های تکاملی، این الگوریتم نیز با تعدادی جمعیت اولیه تصادفی که هر کدام از آنها یک " کشور " نامیده می شوند، شروع می شود. تعدادی از بهترین عناصر جمعیت (معادل نخبه ها در الگوریتم ژنتیک) به عنوان امپراتوری^۱ انتخاب می شوند. باقیمانده جمعیت نیز به عنوان مستعمره^۲ در نظر گرفته می شوند. استعمارگران بسته به قدرتشان، این مستعمرات را با یک روند خاص که در ادامه به آن اشاره می کنیم، به سمت خود می کشند. قدرت کل هر امپراطوری، به هر دو بخش تشکیل دهنده آن، یعنی کشور امپریالیست به عنوان هسته مرکزی و مستعمراتش بستگی دارد. در حالت ریاضی، این وابستگی با تعریف قدرت امپراطوری به صورت مجموع قدرت کشور امپریالیست به اضافه در صدی از میانگین قدرت مستعمرات آن، مدل شده است. با شکل گیری امپراطوری های اولیه، رقابت امپریالیستی میان آن ها شروع می شود. هر امپراطوری که نتواند در رقابت استعماری موفق عمل کرده و بر قدرت خود بیفزاید و یا حداقل از کاهش نفوذش جلوگیری کند، از صحنه رقابت استعماری حذف خواهد شد. بنابراین بقای یک امپراطوری، وابسته به قدرت آن در جذب مستعمرات امپراطوری های رقیب و به سطره در آوردن آنها خواهد بود. در نتیجه، در جریان رقابت های امپریالیستی به تدریج بر قدرت امپراطوری های بزرگتر افزوده شده و امپراطوری های ضعیف تر حذف خواهند شد. امپراطوری ها برای افزایش قدرت خود مجبور خواهند شد تا مستعمرات خود را نیز پیشرفت دهند.

۴-۳-۱- شکل دهی امپراطوری های اولیه

در بهینه سازی، هدف یافتن یک جواب بهینه بر حسب متغیرهای مسئله می باشد. ما یک آرایه از متغیرهای مسئله را که باید بهینه شوند، ایجاد می کنیم. در الگوریتم ژنتیک این آرایه کروموزوم

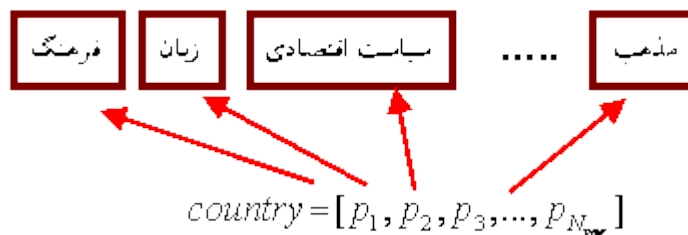
^۱ Imperial

^۲ Colony

نامیده می‌شود. در اینجا نیز آن را یک کشور می‌نامیم. در یک مسئله بهینه‌سازی، یک کشور، یک آرایه بطول $1 * N_{var}$ است. که N_{var} تعداد متغیرهای تابعی است که می‌خواهیم بهینه شود. این آرایه به صورت رابطه‌ی (۱-۴) تعریف می‌شود.

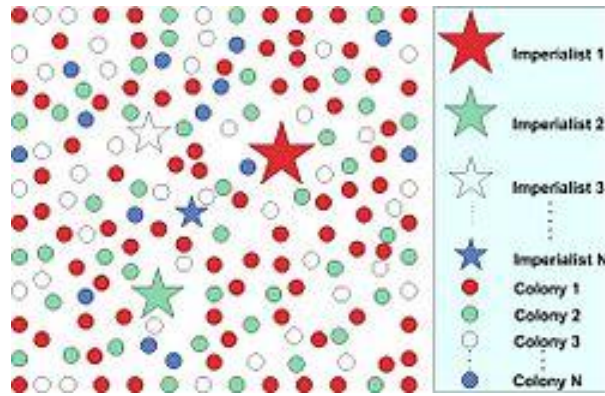
$$\text{Country} = [p_1, p_2, \dots, p_{N_{var}}] \quad (1-4)$$

مقادیر متغیرها در یک کشور به صورت اعداد اعشاری نمایش داده می‌شوند. از دیدگاه تاریخی و فرهنگی، اجزای تشکیل دهنده یک کشور را می‌توان ویژگی‌های اجتماعی-سیاسی آن کشور همچون فرهنگ، زبان، ساختار اقتصادی و سایر ویژگی‌ها در نظر گرفت. شکل (۲-۴) نحوه تناظر متغیرهای بهینه‌سازی مسئله با ویژگی‌های اجتماعی-سیاسی را نشان می‌دهد.



شکل ۲-۴ تناظر متغیرهای بهینه‌سازی مسئله با ویژگی‌های اجتماعی سیاسی

برای شروع الگوریتم، تعداد $N_{country}$ کشور اولیه را ایجاد می‌کنیم تا N_{imp} از بهترین اعضای این جمعیت که کشورهای دارای کمترین مقدار تابع هزینه هستند را به عنوان امپریالیست انتخاب می‌کنیم. باقیمانده N_{col} از کشورها، مستعمراتی را تشکیل می‌دهند که هرکدام به یک امپراطوری تعلق دارند. برای تقسیم مستعمرات اولیه بین امپریالیست‌ها، به هر امپریالیست تعدادی مستعمره، متناسب با قدرت آن امپراطوری می‌دهیم. در شکل (۳-۴) نحوه تقسیم مستعمرات، میان کشورهای استعمارگر به صورت نمادین نشان داده شده است.

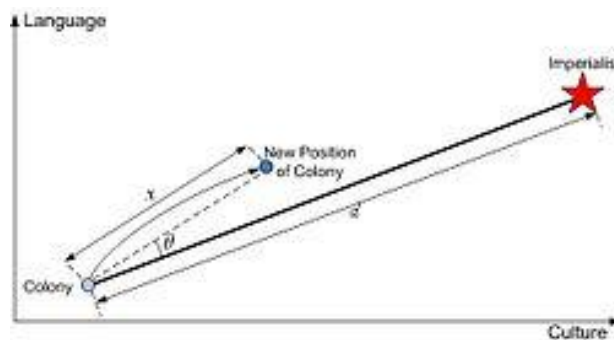


شکل ۳-۴ نحوه تقسیم مستعمرات، میان کشور های استعمارگر

۲-۳-۴ سیاست جذب و حرکت مستعمره ها به سمت امپریالیست

سیاست همگون سازی یا جذب، با هدف تحلیل فرهنگ و ساختار اجتماعی مستعمرات در فرهنگ حکومت مرکزی انجام می گرفت. همانگونه که قبلاً نیز بیان شد، کشورهای استعمارگر برای افزایش نفوذ خود شروع به ایجاد عمران (ایجاد زیرساخت های حمل و نقل، تاسیس دانشگاه و ...) کردند. به عنوان مثال کشورهای نظیر انگلیس و فرانسه با تعقیب سیاست همگون سازی در مستعمرات خود در فکر ایجاد انگلیسی نو و فرانسه نو در مستعمرات خویش بودند. با در نظر گرفتن شیوهی نمایش یک کشور در حل مسئله بهینه سازی، این بخش از فرایند استعمار در الگوریتم بهینه سازی به صورت حرکت مستعمرات به سمت کشور امپریالیست، مدل شده است [۳۵].

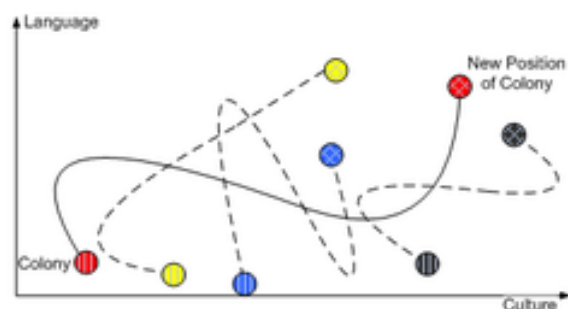
در راستای این سیاست، مطابق شکل (۴-۵) کشور مستعمره، به اندازه x واحد در جهت خط واصل مستعمره به استعمارگر، حرکت کرده و به موقعیت جدید x ، کشانده می شود.



شکل ۴-۵ اعمال سیاست جذب در الگوریتم رقابت استعماری

۴-۳-۳- انقلاب و تغییر ناگهانی در موقعیت یک کشور

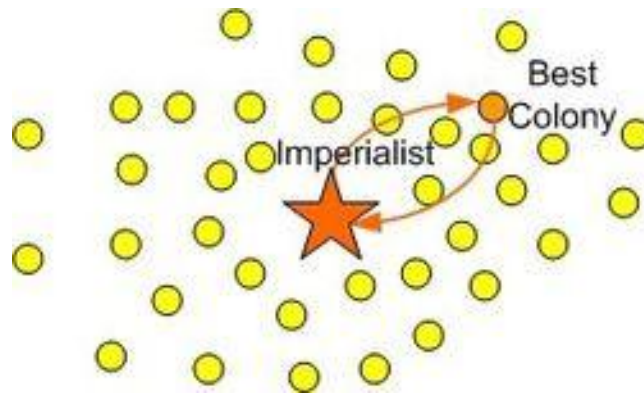
بروز انقلاب، تغییرات ناگهانی را در ویژگی‌های اجتماعی و سیاسی یک کشور ایجاد می‌کند. در الگوریتم رقابت استعماری، انقلاب با جابجایی تصادفی یک کشور مستعمره به یک موقعیت تصادفی جدید مدل‌سازی می‌شود. انقلاب از دیدگاه الگوریتمی باعث می‌شود کلیت حرکت تکاملی از گیر کردن در دره‌های محلی بهینگی نجات یابد که در بعضی موارد باعث بهبود موقعیت یک کشور شده و آن را به یک محدوده بهینگی بهتری می‌برد [۳۵]. شکل (۴-۴)، چگونگی مدل‌سازی پدیده انقلاب در الگوریتم رقابت استعماری را نشان می‌دهد.



شکل ۴-۴ اعمال سیاست انقلاب

۴-۳-۴- جابجایی موقعیت مستعمره و امپریالیست

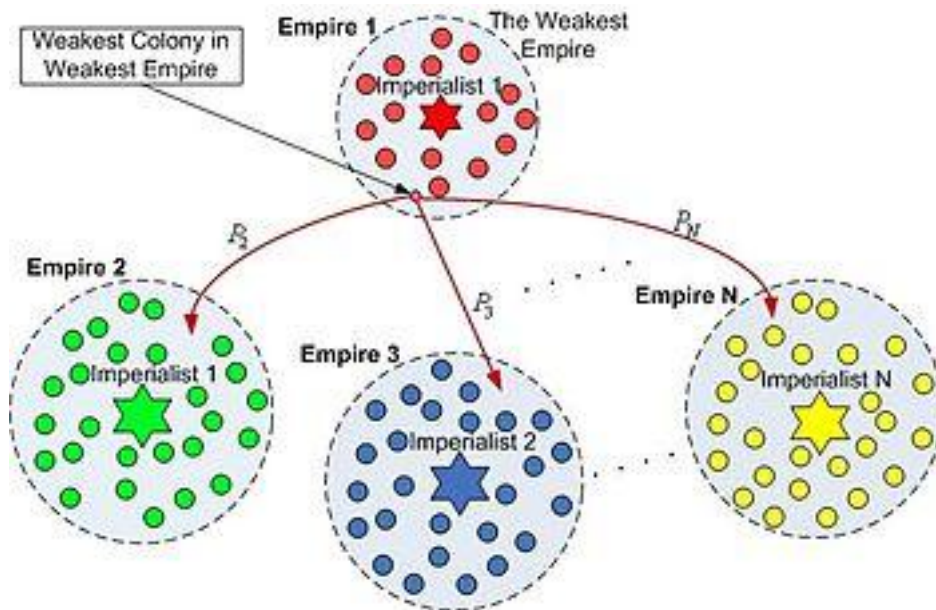
در حین حرکت مستعمرات به سمت کشور استعمارگر، ممکن است بعضی از این مستعمرات به موقعیتی بهتر از امپریالیست برسند. به عبارت دیگر به نقاطی در تابع هزینه برسند که هزینه کمتری را نسبت به مقدار تابع هزینه در موقعیت امپریالیست، تولید می‌کند. در این حالت کشور استعمارگر و کشور مستعمره جای خود را با همدیگر عوض کرده و الگوریتم با کشور استعمارگر در موقعیت جدید ادامه می‌یابد و اینبار این کشور امپریالیست جدید است که شروع به اعمال سیاست همگون‌سازی بر مستعمرات خود می‌کند. نحوه جابجایی موقعیت مستعمره و استعمارگر در شکل (۴-۵) نشان داده شده است.



شکل ۴-۵ جایابی موقعیت مستعمره واستعمارگر

۴-۳-۵- رقابت استعماری

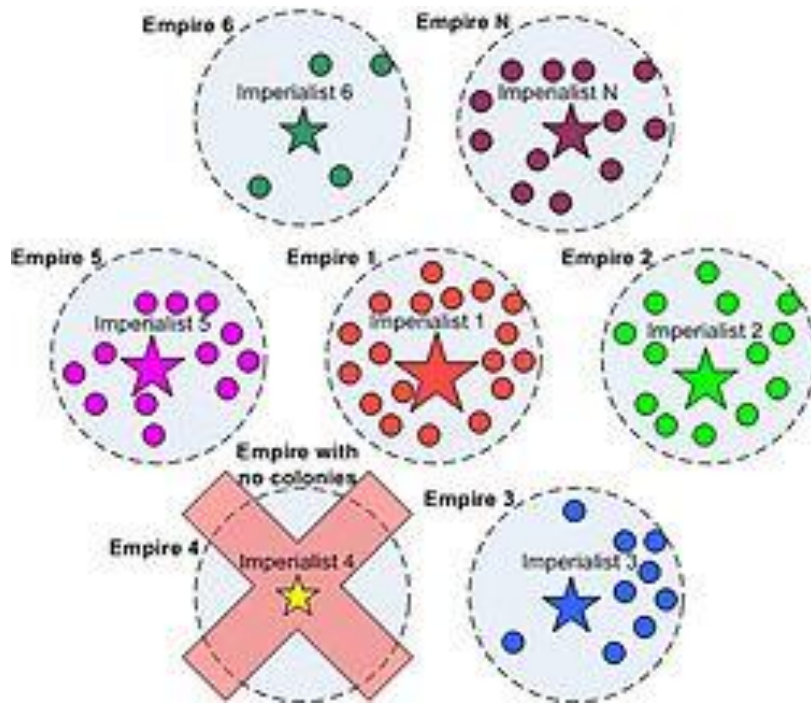
قدرت یک امپراطوری به صورت قدرت کشور استعمارگر به اضافه درصدی از قدرت کل مستعمرات آن تعریف می‌شود. هر امپراطوری‌ای که نتواند بر قدرت خود بیفزاید و قدرت رقابت خود را از دست بدهد، در جریان رقابت‌های امپریالیستی حذف خواهد شد. این حذف شدن به صورت تدریجی صورت می‌پذیرد. بدین معنی که به مرور زمان امپراطوری‌های ضعیف، مستعمرات خود را از دست داده و امپراطوری‌های قویتر، این مستعمرات را تصاحب کرده و بر قدرت خویش می‌افزایند. در الگوریتم رقابت استعماری امپراطوری در حال حذف، ضعیف‌ترین امپراطوری موجود است. بدین ترتیب در تکرار الگوریتم، یک یا چند مورد از ضعیف‌ترین مستعمرات ضعیف‌ترین امپراطوری را برداشته و برای تصاحب این مستعمرات، رقابتی را میان کلیه امپراطوری‌ها ایجاد می‌کنیم. مستعمرات مذکور لزوماً توسط قویترین امپراطوری تصاحب نخواهند شد و رقابت بین کل انجام می‌پذیرد. رقابت استعماری میان چندین استعمارگر برای جذب مستعمرات یکدیگر، در شکل (۴-۶) بخوبی نشان داده شده است [۳۵].



شکل ۴-۶ رقابت استعماری میان چندین استعمارگر

۴-۳-۶- سقوط امپراطوری های ضعیف

در جریان رقابت‌های امپریالیستی، بتدریج امپراطوری‌های ضعیف به تدریج سقوط کرده و مستعمراتشان به دست امپراطوری‌های قوی‌تر می‌افتد. شروط متفاوتی را می‌توان برای سقوط یک امپراطوری در نظر گرفت. در الگوریتم پیشنهاد شده، یک امپراطوری زمانی حذف شده تلقی می‌شود که تمامی مستعمرات خود را از دست داده باشد [۳۵]. **Error! Bookmark not defined.** شکل (۴-۷) یک نمای کلی از سقوط امپراطوری‌ها در روند چرخه الگوریتم، ارائه می‌دهد.



شکل ۴-۷ سقوط امپراطوری هادر روند چرخه الگوریتم رقابت استعماری

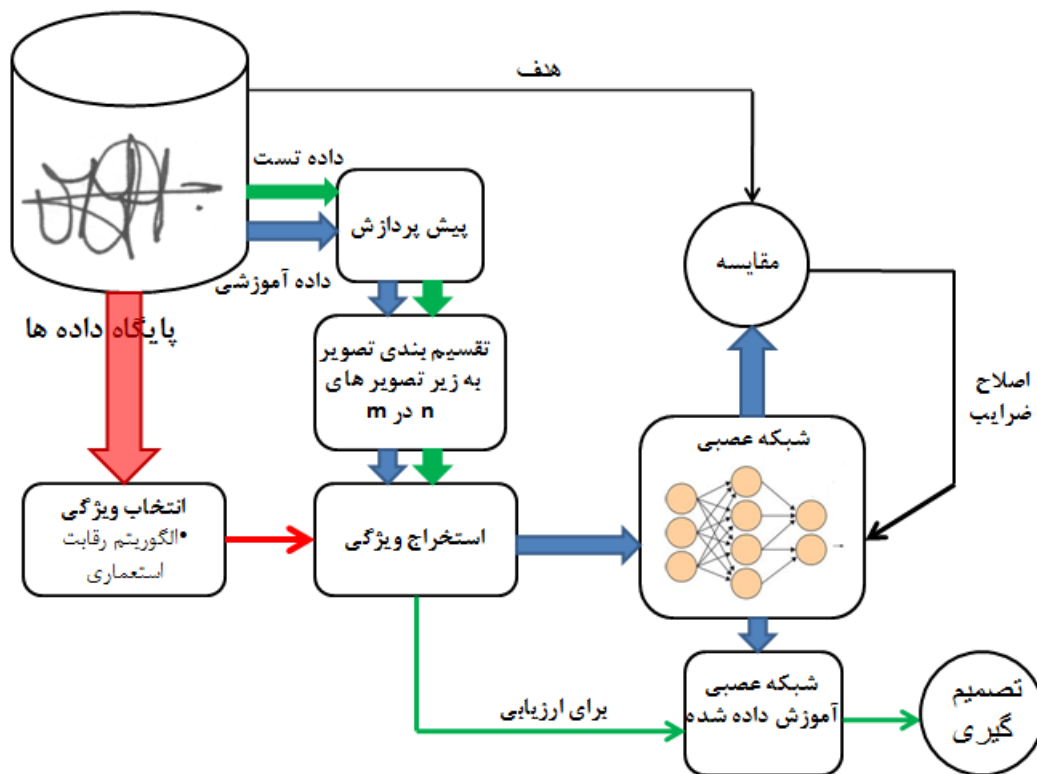
الگوریتم پیشنهادی

همانطور که در فصل اول اشاره شد، در این پایان نامه الگوریتم جدیدی برای استخراج ویژگی جهت تشخیص صحت امضاءهای برون خط ارائه می‌گردد. همانند اکثر سیستم‌های تشخیص امضاء، سیستم ارائه شده شامل مراحل استخراج ویژگی و طبقه‌بندی می‌باشد. در این الگوریتم با توجه به اهمیت ویژگی‌های محلی، روش جدیدی برای به دست آوردن این ویژگی‌ها بر اساس ناحیه بندی افقی، ارائه می‌شود و الگوریتم رقابت استعماری که به تفصیل در فصل چهارم معرفی شد، جهت انتخاب ویژگی‌های بهینه به کار برده می‌شود. برای طبقه بندی ویژگی‌های استخراج شده، از شبکه عصبی پرسپترون استفاده شده است.

۵-۱- بلوک دیاگرام روش پیشنهادی

شکل (۵-۱) بلوک دیاگرام کلی الگوریتم پیشنهادی برای امضاءهای برون خط ارائه شده در این پایان نامه را نشان می‌دهد. امضاءهای جمع آوری شده‌ی هر کاربر به دو قسمت امضاءهای آموزشی و امضاءهای آزمون تقسیم می‌شوند. همانطور که در شکل نشان داده شده است، در الگوریتم پیشنهادی ابتدا امضاءهای آموزشی هر کاربر پس از اسکن و قطعه بندی مناسب، پیش پردازش شده و در ادامه ویژگی‌های مناسب برای مسئله تأیید امضاء، استخراج می‌شوند. سپس از الگوریتم رقابت استعماری برای انتخاب زیرمجموعه‌ای بهینه از این ویژگی‌ها استفاده می‌کنیم. بهینه به این معنا که هم عمل کاهش ابعاد بردار ویژگی را جهت کاهش پیچیدگی محاسباتی انجام دهد و هم باعث بهبود کارایی سیستم شود. در مرحله بعد ویژگی‌های انتخاب شده را برای آموزش شبکه عصبی به کار می‌بریم (برای هر کاربر یک شبکه در نظر می‌گیریم) و شبکه آموزش می‌بیند و ضرایبش تصحیح می‌شود. در مرحله ارزیابی امضاءهای آزمون همانند امضاءهای آموزشی وارد مراحل پیش پردازش و استخراج ویژگی می‌شوند و سپس ویژگی‌های استخراج شده به عنوان

ورودی به شبکه عصبی آموزش دیده شده متناظر با هر کاربر داده می‌شوند و در نهایت شبکه یک عدد را به عنوان خروجی معرفی می‌کند و برای تأیید و یا رد امضای ورودی تصمیم گیری می‌شود.



شکل ۵-۱ بلوک دیاگرام کلی سیستم تأیید امضاء برون خط ارائه شده در این پایان نامه

در ادامه توضیح در مورد هر کدام از بخش‌های مختلف بلوک دیاگرام الگوریتم پیشنهادی را، در بخش‌های جداگانه‌ای عنوان می‌کنیم.

۵-۲- پیش پردازش

در روش برون خط از افراد خواسته می‌شود تا بر روی یک صفحه چندین بار امضاء کنند. سپس این صفحات اسکن شده و بصورت یک فایل تصویری ذخیره می‌شود. قبل از آنکه از تصویر امضاء، ویژگی‌هایی را استخراج کنیم، نیاز به این است که تغییراتی را در این فایل ایجاد نماییم که به عملیات فوق پیش پردازش می‌گویند. با توجه به روشی که برای استخراج ویژگی در نظر گرفتیم و با در نظر گرفتن سیستم مورد استفاده در کل پروسه تأیید امضاء، ما نیازمند به ایجاد تنها بعضی از

این تغییرات هستیم. این در حالی است که تغییرات بیشتر نه تنها بهبودی در وضعیت ایجاد نمی‌کند، بلکه ممکن است کارایی سیستم ما را کمتر نیز کند. در فصل دوم، رایج ترین مراحل پیش پردازش به تفصیل بررسی شد که در اینجا نیز بصورت فهرست وار بیان می‌کنیم.

- تبدیل تصویر رنگی به تصویر خاکستری

- تبدیل تصویر خاکستری به باینری

- از بین بردن فضای سفید اضافی

- حذف تغییرات اندازه امضاءها

- حذف چرخش

- به دست آوردن اسکلت امضاء

همانطور که در فصل دوم گفته شد امضاءهای متفاوت یک فرد با هم اختلاف دارد و نمونه‌های امضاء یک فرد خاص نیز از نظر طول، اندازه و جهت با هم متفاوت می‌باشد. پس تغییر در این ویژگی‌ها جعلی بودن و اصلی بودن امضا را مشخص نمی‌کنند و باید اصلاح شوند. برای رفع این مشکل ما ناگزیر به استفاده از عملیات هم اندازه کردن امضاءها و حذف چرخش می‌باشیم. اما در جمع‌آوری نمونه‌های بعضی پایگاه داده‌ها، از همان ابتدا از امضاءکنندگان خواسته می‌شود تا درون کادر مشخصی امضاء نمایند و در هنگام امضاء نیز جهت را همواره رعایت کنند. پس دیگر نیازی به این عملیات نیست. پایگاه داده‌هایی که ما در این پروژه از آنها استفاده می‌کنیم نیز، با همین دقت تهیه شده‌اند و در نتیجه ما از این موارد صرف نظر کردیم .

به علاوه مرحله تبدیل تصویر رنگی به تصویر با مقیاس خاکستری نیز قبلا توسط گردآورنده پایگاه انجام شده بود و با توجه به این که یک زیرمجموعه موثر از ویژگی‌ها بر مبنای تغییر در سطوح روشنایی قسمت های مختلف امضاء بود، از اینکه تصویر را به تصویر باینری تبدیل کنیم نیز خودداری می‌کنیم.

همچنین در مورد به دست آوردن اسکلت امضاء و جایگزین کردن آن به جای تصویر اولیه، درست است که با این کار حجم فایل های ذخیره شده کمتر می شود و این موضوع اهمیت خاصی دارد، ولی کاملاً مشهود است که با این تغییر، اطلاعات مهمی را از دست می دهیم. ضخامت منحنی امضاء، یکی از ویژگی های منحصر به فرد امضای یک کاربر می باشد. پس در این پروژه برای پیش پردازش تنها فضای سفید اضافی را از بین می بریم.

۵-۲-۱- از بین بردن فضای سفید اضافی

همانطور که گفته شد در به دست آوردن پایگاه داده به امضاء کنندگان صفحاتی داده می شود که در محل های تعیین شده امضاء کنند. افراد در فضای مشخص شده برای هر امضاء در مکان های متفاوتی امضاء می کنند. بنابراین بعد از جداسازی امضاها از هم، همواره فضای سفید اضافی وجود دارد که اطلاعات ناچیزی را در بر دارد و تنها حجم فایل های پایگاه داده را بالا می برد. پس به عنوان پیش پردازش، این فضای سفید را از بین می بریم. شکل (۵-۲) امضاءها را قبل و بعد از این پیش پردازش نشان می دهد.



شکل ۵-۲ عملیات حذف ناحیه سفید اضافی

۵-۳- ناحیه بندی تصویر امضاء بصورت M در N

استخراج ویژگی یکی از مراحل مهم تأیید امضاست. قدرت تمایز بالا و قابلیت تعمیم ویژگی‌ها به همه کاربرها، نقش مهمی در فرایند تأیید امضاء ایفا می‌کند [۳۶]. منظور از فرایند استخراج ویژگی آن است که ویژگی‌های یکتایی برای تمام حالات کاربر، که در تمام یا اکثریت نمونه‌های امضاء آن کاربر تکرار می‌شود، استخراج گردد. با این هدف که به فرایند تأیید امضاء کمک شود تا با صحت قابل قبولی اجرا شود.

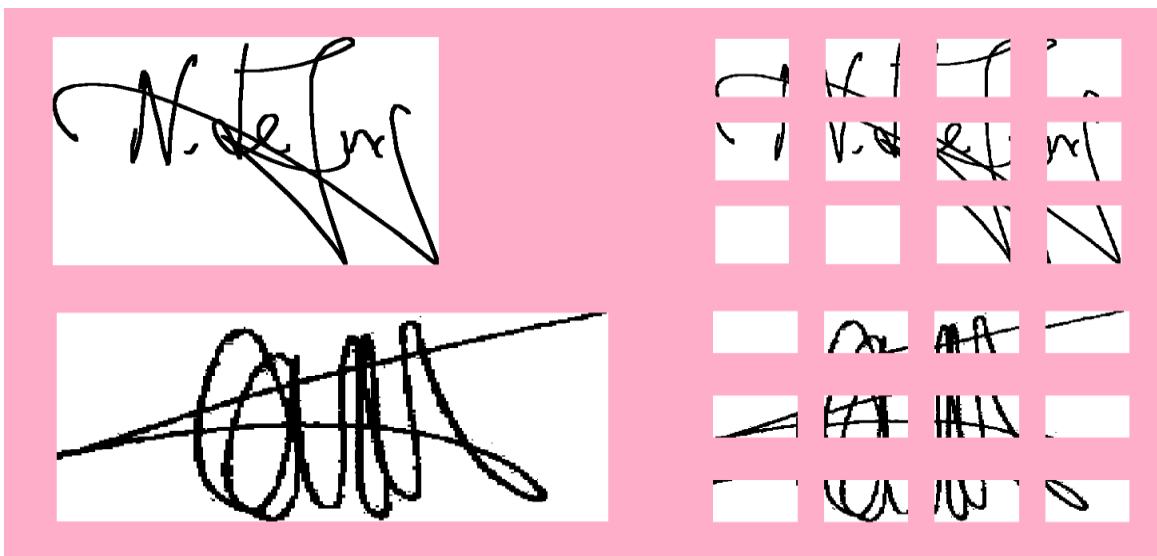
در فصل دوم، درباره انواع ویژگی‌های استفاده شده در سیستم‌های تأیید صحت امضاء صحبت شد. همانطور که گفته شد ویژگی‌های استفاده شده، بر اساس نقطه نظرهای مختلف قابل تقسیم بندی بصورت‌های گوناگون هستند. از یک منظر می‌توان ویژگی‌ها را به دو دسته کلی ویژگی‌های سراسری و محلی تقسیم کرد. ویژگی‌هایی که برای به دست آوردن آنها تصویر اسکن شده امضاء را بصورت کلی در نظر می‌گیرند، ویژگی‌های سراسری نامیده می‌شوند و ویژگی‌هایی که در آن از بخشی از امضاء استفاده می‌شود و یا اینکه تصویر امضاء را به قسمت‌های مختلفی تقسیم بندی می‌کنند، ویژگی‌های محلی خوانده می‌شوند. این تقسیم بندی می‌تواند بصورت ساده باشد و یا بصورت‌های دیگر مانند دایره ای و

سیستم‌های سراسری، بسیار سریع می‌باشند ولی در عین حال عملکرد پایینی نسبت به سیستم‌های محلی دارند. در حالیکه سیستم‌های محلی به علت درجه بندی اطلاعات امضاء، کارایی بالاتری دارند [۳۷].

در انجام این پایان نامه از همان ابتدا، همه‌ی کارهای انجام شده حول این هدف بود که الگوریتمی ارائه شود تا خطای تصمیم‌گیری را حداقل کند. مطالعه پژوهش‌ها و تحقیق‌های گذشته به ما ثابت کرد که خطا در سیستم‌های محلی نسبت به سیستم‌های سراسری کمتر می‌باشد. به علاوه اینکه در بعضی کارهای انجام شده که از هر دو نوع ویژگی استفاده کرده بودند،

خطا به مراتب کمتر بود. این مهم ما را بر این داشت تا الگوریتمی را بر مبنای ویژگی‌های محلی و شبکه‌ای پیشنهاد کنیم.

از آنجاییکه تقسیم بندی شبکه‌ای، دسترسی به ویژگی‌های محلی را آسانتر می‌کنند، مقالات زیادی وجود دارند که در آنها استخراج ویژگی بر مبنای این تقسیم بندی، انجام شده است [۴۱,۴۰,۳۹,۳۸]. روش سنتی تقسیم بندی شبکه‌ای بدین صورت است که تصویر امضاء را با قاب شطرنجی M در N ای به بلاک‌هایی مساوی تقسیم بندی می‌کنیم. تقسیم بندی بصورت سنتی در شکل (۳-۵) نشان داده شده است.



شکل ۳-۵ تقسیم بندی امضاء به روش ساده

در روش‌های سنتی، تقسیم بندی دیگری علاوه بر روش تقسیم بندی ساده وجود دارد و آن تقسیم بندی دایره ایست. بدینسان که بر اساس مرکز ثقل امضاء، برش‌ها بصورت قطرهای دایره در نظر گرفته می‌شوند. نمونه‌ای از تقسیم بندی تصویر امضاء به روش دایره‌ای در شکل (۴-۵) آمده است.



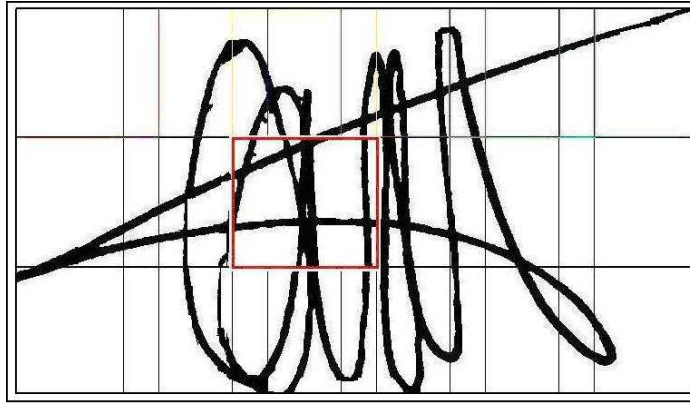
شکل ۴-۵ تقسیم بندی سنتی

در سال ۲۰۱۰، تقسیم بندی شبکه‌ای به شکلی غیر سنتی ارائه شد. رکیک^۱ و همکارانش تصویر امضاء را طوری تقسیم بندی کردند که قطاع هایی که در یک ردیف هستند با هم همپوشانی داشته باشند [۴۲]. در همین سال همزمان، سوان پل^۲ و همکارانش نیز از همپوشانی قطاع ها استفاده کردند ولی اینبار همپوشانی را در هر دو جهت انجام دادند و نشان دادند که به دلیل اینکه امضاءهای متفاوت یک فرد نیز کاملا با هم یکسان نیست، همپوشانی سلول‌های شبکه با سلول‌های همسایه‌اش، نتیجه بهتری نسبت به حالت سنتی دارد و این بسبب اطلاعات اضافی است که در این حالت از هر سلول به دست می‌آید [۴۳]. سوان پل این روش را به دلیل آنکه مساحت همپوشانی را در آن می‌توان تغییر داد، شبکه بندی قابل انعطاف^۳ نامید. ناحیه بندی امضا بکمک روش‌های رکیک و سوان پل در شکل های (۵-۵) و (۶-۵) نشان داده شده است.

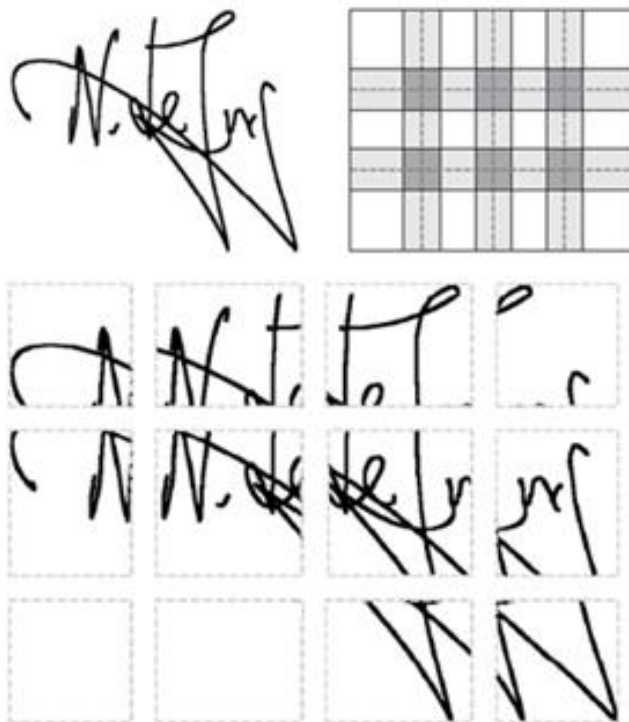
^۱ kReki

^۲ Swanepoel

^۳ Flexible Grid Base



شکل ۵-۵ ناحیه بندی امضاء بکمک روش پیشنهادی رکیک

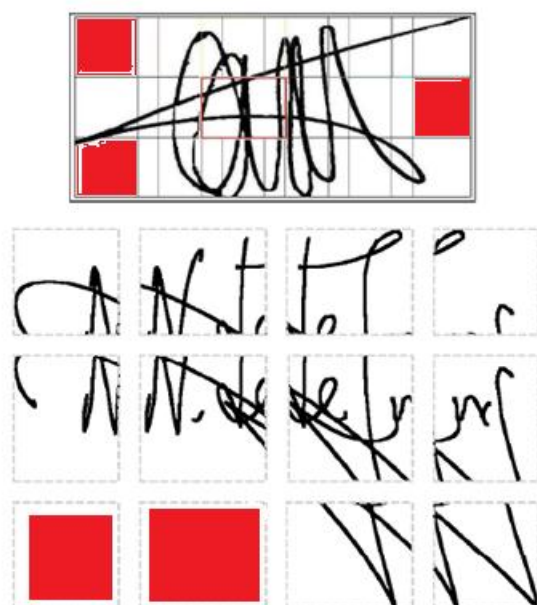


شکل ۵-۶ ناحیه بندی امضاء بکم روش پیشنهادی سوان پل

با مطرح شدن بحث اطلاعات هر سلول، این فکر که تقسیم بندی را به شکل دیگری انجام دهیم، بطوریکه اطلاعات مفیدتری از آنها استخراج شود، بوجود آمد و در این پایان نامه روش زیر را برای شبکه بندی مطرح نمودیم.

در تقسیم بندی سنتی و همچنین شبکه‌ای انعطاف پذیر، برش ها بدون توجه به شکل امضاء انجام می‌شود و تنها از روی مستطیلی که قاب تصویر را تشکیل می‌دهد، مکان‌های برش تعیین می‌شود. بنابراین در اکثر مواقع این حالت پیش می‌آید که چندین سلول خالی داشته باشیم. سلول

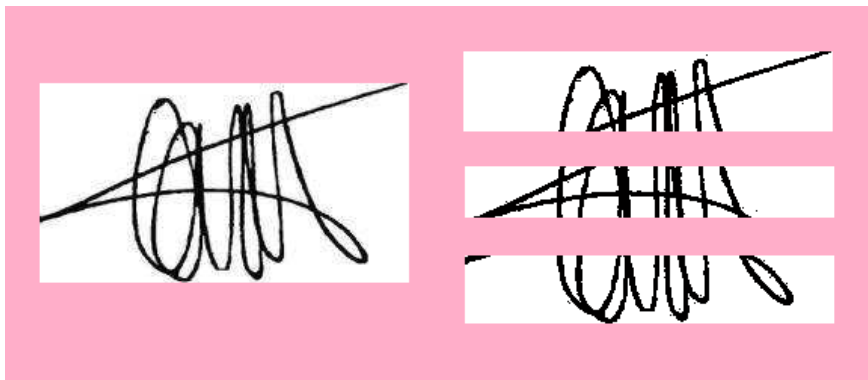
خالی یعنی اطلاعات صفر و در هر حله بعدی که استخراج ویژگی انجام می‌شود، اگر از هر سلول S ویژگی استخراج کنیم و اگر n سلول خالی داشتیم در بردار ویژگی $n*S$ صفر بوجود می‌آید. در شکل (۷-۵)، سلول‌های خالی برای نمونه‌ای از امضا که به روش سوان پل ناحیه بندی شده است، با رنگ قرمز مشخص شده اند تا مفهوم اطلاعات غیر مفیدی که این روش به ما می‌دهد، بهتر درک شود.



شکل ۷-۵ نمایش سلول‌های خالی در روش ناحیه بندی سنتی

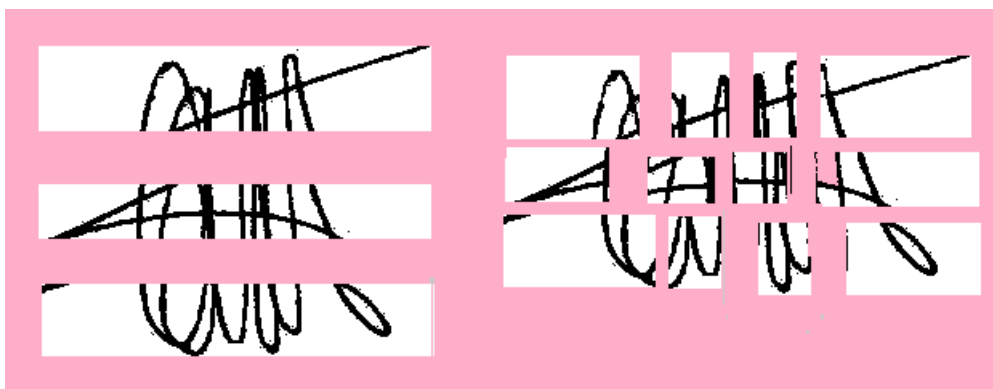
نتیجه می‌گیریم عدم توجه به شکل امضاء برای انتخاب محل قطاع‌ها، اطلاعات اضافی‌ای تولید می‌کند که مفید نیست و تنها حجم زیادی از حافظه پردازنده را اشغال می‌کند. پس روشی را پیشنهاد کردیم که برای به دست آوردن شبکه M در N ، ابتدا امضاء را بطور افقی به N ناحیه

تقسیم کنیم بطوری که تعداد پیکسل‌های سیاه هر کدام از این نواحی با هم برابر باشند. شکل (۵-۸) این تقسیم بندی را نشان می‌دهد.



شکل ۵-۸ تقسیم بندی امضاء در راستای افق به کمک روش پیشنهادی

سپس به همین شکل هر ناحیه را به M قسمت دیگر تقسیم می‌کنیم، طوری که تعداد پیکسل‌های هر کدام از این M قسمت نیز با هم برابر شوند. در این روش مطمئن هستیم هیچ سلول خالی وجود ندارد و اطلاعات سلول‌ها نیز نسبتاً، با هم برابر می‌باشند. ما این روش را تقسیم بندی شبکه‌ای افقی می‌نامیم. زیرا قطعاتها بر اساس شکل امضاء تغییر می‌کنند. در شکل (۵-۹) می‌توانید ناحیه بندی به روش پیشنهادی را بر روی دو نمونه امضا مشاهده کنید.





شکل ۵-۹ تقسیم بندی امضاء به روش پیشنهادی

۵-۴- استخراج ویژگی

همانطور که در بخش پیش ذکر شد، در این پایان نامه، هم از ویژگی‌های محلی و هم از ویژگی‌های سراسری استفاده می‌کنیم. در بخش کارهای انجام شده، به انواع ویژگی‌های استفاده شده در زمینه تأیید صحت امضاء اشاره کردیم. با توجه به تحقیقات انجام شده، ما ویژگی‌های زیر را برای هر امضاء استخراج کردیم. در این قسمت ویژگی‌های متفاوتی را آزمایش کردیم که از آن میان ویژگی‌های زیر کارایی بهتری نسبت به بقیه ویژگی‌ها داشته‌اند.

۵-۴-۱- ویژگی‌های محلی

جدول (۵-۱) ویژگی‌هایی را نشان می‌دهد که از هر سلول استخراج کردیم.

جدول ۵-۱ ویژگی‌های محلی استخراج شده از هر سلول

ردیف	توضیحات
۱	نسبت مجموع سطح خاکستری پیکسل‌های سیاه قرار گرفته در هر سلول به مجموع سطح خاکستری پیکسل‌های سیاه امضاء

<ul style="list-style-type: none"> تعداد پیکسل‌های سیاه قرار گرفته در سلول نسبت به کل پیکسل‌های سیاه امضاء (برای حالت تقسیم بندی سنتی) نسبت طول به عرض هر سلول (برای حالت تقسیم بندی وفقی) 	۲
طول مرکز ثقل هر سلول به طول امضا	۳
عرض مرکز ثقل هر سلول به عرض امضا	۴
فاصله مرکز ثقل هر سلول با مرکز ثقل اصلی تصویر	۵

دلایل انتخاب ویژگی‌های فوق

ویژگی فشار در امضای هر فرد از جمله ویژگی‌هاییست که قدرت تمایز بالایی دارد. این ویژگی در حالت درون خط به کمک قلم نوری و صفحه الکترونیکی، می‌تواند بصورت پارامتر متغیر با زمان استخراج شود. در حالت برون خط نیز اثر این ویژگی بصورت پر رنگ و کمرنگ شدن منحنی امضاء در تصویر اسکن شده، نمایان می‌شود. با توجه به اینکه پایگاه داده‌های استفاده شده در این تحقیق، تصاویر امضاء در مقیاس خاکستری هستند، می‌توانیم با معرفی ویژگی اول این جدول، اثری متناسب با پارامتر فشار را منعکس نماییم.

ویژگی بعدی استفاده شده، از نوع ویژگی‌های آماری می‌باشد و معمولاً در حالت تقسیم بندی شبکه‌ای از این ویژگی استفاده می‌شود. البته برای روش پیشنهادی مطرح شده، این ویژگی هیچ کاربردی ندارد زیرا تقسیم بندی سلول‌ها، اساساً بر مبنای تساوی پیکسل‌های سیاه در هر سلول می‌باشد و این ویژگی را تنها برای حالت سنتی استخراج می‌کنیم. اما معادل این ویژگی می‌تواند نسبت طول به عرض هر سلول در حالت تقسیم بندی وفقی باشد. زیرا بنابر شرط تقسیم بندی این نسبت بر خلاف حالت سنتی، برای هر سلول عددی متفاوت است.

در تحقیقات انجام شده در زمینه تأیید صحت امضاء، همواره توجه خاصی به مرکز ثقل تصویر امضاء شده است و همانطور که در فصل دوم هم به آن اشاره کردیم، در مقالات متعددی از انشعاب دادن مراکز ثقل برای استخراج ویژگی استفاده شده است. به دست آوردن مرکز ثقل برای هر سلول، نیز می‌تواند، نوعی دیگری از این انشعاب دادن ها باشد. پس ما برای هر سلول، طول و عرض مرکز ثقل آن سلول و همچنین فاصله آنرا با مرکز ثقل اصلی تصویر امضاء، به عنوان ویژگی در نظر می‌گیریم.

تا اینجا اگر تعداد سلول‌ها $M*N$ باشد تعداد ویژگی‌های محلی استخراج شده برابر با $5*M*N$ خواهد بود.

۵-۴-۲- ویژگی‌های سراسری

از بین ویژگی‌هایی که می‌توان از روی تصویر امضاء بدون ناحیه بندی استخراج کرد، ما شش ویژگی را برگزیدیم. جدول (۵-۲) ویژگی‌های سراسری استخراج شده از هر نمونه امضاء را نشان می‌دهد.

جدول ۵-۲ ویژگی‌های سراسری

توضیحات	ردیف
ویژگی به دست آمده از روی مراکز ثقل سلول‌ها	$1 + (5*M*N)$
ویژگی به دست آمده از روی مراکز ثقل سلول‌ها	$2 + (5*M*N)$
نسبت تعداد کل پیکسل‌های سیاه به تعداد کل پیکسل‌های تصویر	$3 + (5*M*N)$

امضا	
سطح خاکستری متوسط تصویر	$4 + (5 * M * N)$
طول مرکز ثقل تصویر به مجموع طول و عرض تصویر امضا	$5 + (5 * M * N)$
عرض مرکز ثقل تصویر به مجموع طول و عرض تصویر امضا	$6 + (5 * M * N)$

دو ویژگی اول از روی مراکز ثقل سلول‌ها به دست می‌آیند.

اگر طول مرکز ثقل همه‌ی سلول‌ها را در آرایه‌ی $GL(1)$ و عرض آنها را در آرایه‌ی $GL(2)$

ذخیره نماییم، ویژگی اول از رابطه (۱-۵) و ویژگی دوم از رابطه (۲-۵) به دست می‌آید.

$$(Max (GL(1)) - min GL(1)) / h(1) \quad (1-5)$$

$$(Max (GL(2)) - min GL(2)) / h(2) \quad (2-5)$$

که در آن $h(1)$ طول مرکز ثقل امضاء و $h(2)$ عرض مرکز ثقل امضاء می‌باشد.

۵-۵- فرایند آموزش و آزمایش با استفاده از شبکه عصبی

پایگاه داده GPDS300 GRAY که برای این تحقیق استفاده شده است شامل امضاء ۳۰۰ نفر

می‌باشد که برای هر کاربر ۵۴ امضاء موجود است. از این تعداد، ۲۴ امضاء اصلی و ۳۰ امضاء جعلی

از نوع جعل حرفه‌ای، وجود دارد.

در این قسمت ما از یک شبکه عصبی با ساختار پرسپترون^۱ چند لایه^۲ بمنظور آموزش شبکه

عصبی و آزمایش سیستم تأیید امضاء برون خط استفاده می‌کنیم [۳۶].

^۱ Perceptron

^۲ Multilayer

برای آموزش شبکه عصبی، بردار ویژگی‌های استخراج شده از چند نمونه امضای اصلی و چند نمونه امضای جعلی (جعل حرفه ای) هر کاربر به شبکه عصبی داده می‌شود و شبکه عصبی آموزش می‌بیند. سپس در مرحله آزمایش، از شبکه عصبی آموزش دیده هر کاربر، به عنوان یک طبقه بند دو کلاسه استفاده می‌شود، تا امضاءهای اصلی را از جعلی جدا کند. پس به تعداد نفرات شبکه عصبی خواهیم داشت.

۵-۵-۱- آموزش شبکه عصبی

در این تحقیق ما از شبکه عصبی پس انتشار^۱ (BNN) برای آموزش و تعیین وزن های شبکه عصبی استفاده می‌کنیم. این شبکه بدین صورت عمل می‌کند که در مرحله اول، وزن ها و بایاس ها مقداردهی اولیه می‌شوند و در مرحله بعد این ضرایب، در طی فرآیند تکرار با معیار مینیمم کردن میانگین مربعات خطا^۲ (MSE) بین خروجی شبکه و خروجی های مطلوب در فرایند آموزش، اصلاح و ذخیره می‌شوند. توضیح کامل در مورد شبکه عصبی پس انتشار و چگونگی کارکردش در پیوست (الف) آمده است.

خروجی مطلوب برای امضاءهای اصلی و جعلی یک کاربر را به ترتیب ۱ و ۱- در نظر گرفتیم.

پس برای هر کاربر، یک شبکه عصبی مصنوعی در نظر می‌گیریم و به کمک روش پس انتشار، با دادن نمونه‌های آموزشی برای هر کاربر، ضرایب و بایاس های شبکه را تعیین می‌کنیم و شبکه را آموزش می‌دهیم. تعداد لایه های پنهان و تعداد نرون های هر لایه، در سرعت آموزش و کاهش خطای آموزش موثر می‌باشد. در این پایان نامه ما از یک لایه پنهان استفاده کردیم و برای تعیین مقدار بهینه برای تعداد نرون ها، تعداد نرون های مختلف را برای رسیدن به کمترین میزان خطای سیستم آزمایش کردیم که نتایج آنها را در فصل بعد خواهیم آورد.

^۱ Backpropagation Neural Network

^۲ Mean Square Error

فرآیند آموزش برای هر کاربر بصورت جداگانه صورت می‌گیرد. شرط خاتمه فرآیند آموزش را مینیمم خطای ۱۰ میکرو در نظر گرفتیم. همچنین تعداد دفعات ماکزیمم تکرار را ۳۰۰ تعیین کردیم تا بعد از گذشت این تعداد چرخه حتی اگر خطای شبکه به کمتر از ۱۰ میکرو نرسید، آموزش را تمام کند.

۵-۵-۲- آزمایش شبکه‌های عصبی

هنگامیکه آموزش کامل شد، شبکه باید آزمایش شود. از باقیمانده امضاءهای هر فرد برای مرحله آزمایش استفاده می‌کنیم. یعنی ویژگی‌های ۱۲ امضاء اصلی و ۱۵ امضاء جعلی را استخراج و به عنوان ورودی به شبکه عصبی مربوطه می‌دهیم.

سپس باید خروجی شبکه را به ازای آن زیرمجموعه از ورودی‌ها که در مرحله آموزش استفاده نشده‌اند، بدست آوریم.

در این قسمت به ازای امضاهای آزمایشی هر فرد، شبکه عصبی مربوطه فراخوانی می‌شود و شبکه برای آن، یک مقدار را در خروجی مشخص می‌کند. این مقدار معرف جعلی یا اصلی بودن امضای مربوطه است.

چون ما از جعلی بودن و اصلی بودن این امضاءها اطلاع داریم، می‌توانیم کارایی سیستم پیشنهادی را به راحتی ارزیابی نماییم.

۵-۵-۳- تصمیم‌گیری با استفاده از شبکه عصبی

انتظار داریم که وقتی بردار ویژگی‌های امضاءهای تست را به شبکه آموزش دیده شده می‌دهیم، شبکه مقادیر ۱ و ۰-۱ را برای خروجی بدهد. اما بدلیل ساختار پیچیده شبکه عصبی، خروجی مقادیر دیگری خواهد بود. مثلاً می‌تواند ۰,۳ یا ۱,۲ باشد. ما این مشکل را با تعیین یک مقدار

آستانه^۱ بر طرف کردیم. پس از آن مقادیر مثبت نسبت به این آستانه را معادل با همان ۱ و مقادیر منفی را معادل با ۱- در نظر می‌گیریم. معمولاً مقدار این حد آستانه صفر می‌باشد ولی اگر مقدار آستانه عدد دیگری باشد، ابتدا آن را از خروجی کم می‌کنیم. سپس فاصله عدد حاصل را با اعداد ۱ و ۱- محاسبه می‌کنیم. اگر فاصله عدد حاصل تا عدد ۱ کمتر از فاصله‌اش تا ۱- باشد، امضای متناظر با آنرا اصلی می‌شناسیم و در غیر اینصورت آنرا جعلی معرفی می‌کنیم.

نتایج مربوط به شبیه سازی الگوریتم پیشنهادی برای پایگاه داده GPDS۳۰۰GRAY با استفاده از شبکه عصبی در فصل بعد آورده می‌شود.

۵-۶- معیار میزان خطای سیستم

همانطور که گفته شد، ما در این پایان نامه از شبکه عصبی برای کلاسه بندی امضاءها استفاده می‌کنیم. در این بخش نحوه محاسبه میزان خطا برای این کلاسه‌بند را توضیح می‌دهیم.

در بخش قبل اشاره کردیم که خروجی شبکه عصبی آموزش دیده شده در مرحله ارزیابی، هر عددی می‌تواند باشد و لزوماً ۱ یا ۱- نیست. پس وجود یک آستانه برای تصمیم‌گیری ضروریست. چون معمولاً این آستانه را صفر در نظر می‌گیرند ما نیز این عدد را برای حد تصمیم‌گیری بر می‌گزینیم.

ابتدا برای تمامی امضاءهای مرحله آزمایش با توجه به این آستانه، جعلی بودن و یا اصلی بودن را به کمک شبکه عصبی تعیین می‌کنیم. این در حالیست که در حقیقت ما از جعلی بودن امضاءهای مرحله آزمایش با خبریم. پس براحتی می‌توانیم خطا را محاسبه کنیم.

^۱ Treshold

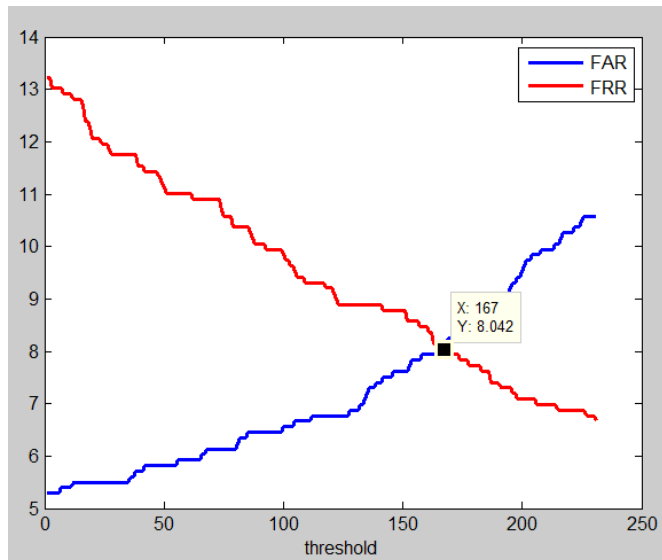
معیار هایی که در فصل ۲ معرفی شد، FAR و FRR و EER بودند که معیار هایی استاندارد در زمینه تأیید صحت امضاء می‌باشند و در بیشتر کارهای انجام شده از آنها برای محاسبه خطا استفاده می‌شود.

برای به دست آوردن نرخ خطای FAR همانطور که از نامش پیداست، کفایت امضاءهای جعلی‌ای که صحیح تشخیص داده شده‌اند را بشماریم و بر کل امضاءهای آزمایش تقسیم کنیم. به همین شکل با شمارش امضاءهای اصلی که جعلی معرفی شدند و تقسیم عدد حاصل به تعداد کل امضاءها، FRR به دست می‌آید.

همانطور که در فصل دوم ذکر شد، FAR و FRR به آسانی با تغییر آستانه تصمیم‌گیری برای تأیید امضاء، تغییر می‌کنند. پس معیار دقیقی برای سنجش خطا و مقایسه سیستم‌های مختلف نمی‌باشند.

اگر آستانه عوض شود می‌تواند یکی از آنها کمتر شود. کاهش یکی افزایش دیگری را در بر می‌گیرد. اگر تغییر آستانه را بیشتر ادامه دهیم می‌توانیم به نقطه‌ای دست یابیم که این دو نرخ با هم برابر می‌شوند و این نقطه در یک سیستم همواره مقدار ثابتی است. پس این عدد می‌تواند نرخ خطای مناسبی برای سنجش سیستم‌ها باشد و اکثراً در تحقیقات مختلف، به عنوان نرخ خطای نهایی مقایسه می‌شود. برای به دست آوردن آن کفایت نمودار تغییرات FAR و FRR را بر اساس آستانه تصمیم‌گیری در یک نمودار رسم کنیم. سپس محل تقاطع دو منحنی را پیدا کنیم. این همان EER است. همچنین در تحقیقات انجام شده، عدد EER-۱۰۰ را تحت عنوان دقت^۱ سیستم (Acc) برای ارزیابی کارایی، محاسبه می‌کنند. ما نیز در این پایان نامه این مقدار را محاسبه کرده و بعنوان ابزاری برای مقایسه بکار می‌گیریم.

^۱ Accuracy



شکل ۵-۱۰ نمودار چگونگی به دست آوردن EER

در شکل (۵-۱۰)، چگونگی به دست آوردن خطای EER نشان داده شده است. در اینجا EER برابر با $8,042$ و Acc برابر با $91,058$ می‌باشد.

۵-۷- انتخاب ویژگی

در فصل سوم به لزوم استفاده از الگوریتم‌های انتخاب ویژگی در این پروژه پی بردیم. به دلیل توجه بسیاری که در سال‌های اخیر به الگوریتم رقابت استعماری شده است و مزیت‌های آن نسبت به الگوریتم‌های انتخاب ویژگی دیگر و به دلیل آنکه ما به دنبال آزمایش روش‌های جدیدتر بودیم، از این الگوریتم برای انتخاب ویژگی‌های بهینه استفاده می‌کنیم. در این بخش به این می‌پردازیم که چگونه الگوریتم رقابت استعماری را به عنوان یک الگوریتم انتخاب ویژگی در این پروژه به کار ببریم. این مستلزم آن است که ابتدا به صورت جزئی‌تر به بررسی قسمت‌های مختلف این الگوریتم و نحوه پیاده‌سازی آنها بپردازیم.

هنگامی که مسئله‌ای را حل می‌کنیم، معمولاً دنبال آن هستیم که به بهترین راه حل و یا به بیان دیگر به یک حل بهینه از بین حل‌های ممکن برای مسئله برسیم. به محدوده‌ای که جواب‌های مسئله قابل قبول می‌باشند، بطوریکه جواب بهینه نیز یکی از زیرمجموعه‌های این محدوده

است، فضای جستجو^۱ گفته می‌شود [۴۴]. هر نقطه از فضای جستجو نشان‌دهنده یکی از روش‌های حل مسئله می‌باشد. هر راه حل نیز بصورت عدد خروجی که از آن راه حل به دست می‌آید، مشخص می‌شود، که اصطلاحاً به این مشخصه و راه حل مربوط به آن هزینه^۲ می‌گویند.

در واقع جستجو برای پیدا کردن یک راه حل برای مسئله معادل آن است که برای به دست آوردن مقادیر اکسترمم مربوط به مسئله در محدوده فضای جستجو، مشغول جستجو هستیم. در اینجا، ما بوسیله الگوریتم رقابت استعماری به دنبال راه حل بهینه، از بین کلیه راه حل‌های ممکن مشخص شده در محدوده فضای جستجو می‌باشیم.

ه-۷-۱ - استفاده از الگوریتم رقابت استعماری بعنوان الگوریتم انتخاب

ویژگی

همانطور که گفته شد موقعیت‌های کشورها، بصورت آرایه‌ای به تعداد متغیرهای تابعی که می‌خواهیم بهینه شود، تعریف می‌شود. مثلاً برای یک تابع ۵ متغیره داریم:

$$\text{Country}_1 = [7 \ 2 \ 9 \ 4 \ 5];$$

$$\text{Country}_2 = [6 \ 4 \ 5 \ 3 \ 2];$$

برای اینکه بتوانیم از الگوریتم‌های بهینه‌سازی برای انتخاب ویژگی استفاده کنیم، یک راه این است که به تعداد ویژگی‌های اولیه، متغیر تعریف کنیم. بطوری که دامنه این متغیرها تنها اعداد {۰، ۱} باشند. عدد ۱ برای انتخاب آن ویژگی و عدد ۰ برای عدم انتخاب آن. در این روش کشورها یا همان کروموزوم‌ها در الگوریتم ژنتیک، رشته‌ای شامل این متغیرها می‌باشند. به طور مثال اگر ۵ ویژگی داشته باشیم که بخواهیم در موردشان تصمیم‌گیری کنیم رشته تولید شده برابر خواهد بود با:

^۱ Search Space

$$\text{Countries}=[x_1 x_2 x_3 x_4 x_5]$$

در اینجا ما ۵ ویژگی داریم که انتخاب و عدم انتخابشان را بررسی می‌کنیم. پس تعداد کل کشور های ممکن یا دقیقتر تعداد کشورهای فضای جستجو برابر $2^5=32$ خواهد بود. در زیر دو نمونه از این کشور ها را ملاحظه می‌کنید:

$$\text{Country}_1 = [0 \ 1 \ 1 \ 0 \ 1];$$

$$\text{Country}_2 = [0 \ 0 \ 1 \ 1 \ 1];$$

کشور اول معرف حالتی می‌باشد که ویژگی‌های ۲، ۳ و ۵ انتخاب شده است و ویژگی‌های ۱ و ۴ خیر.

در این پایان نامه، ما از همین روش، برای توصیف فضای ویژگی برای الگوریتم رقابت استعماری استفاده می‌کنیم. تا اینجا در توضیح الگوریتم رقابت استعماری، ما دامنه متغیرهای تابع برازش یا همان تابع هزینه را پیوسته فرض کردیم. در حالی که برای روشی که برای حل مسئله انتخاب ویژگی برگزیدیم، دامنه متغیرهایمان یا همان فضای جستجو، گسسته است. پس قبل از شروع کار باید تغییراتی در الگوریتم رقابت استعماری ایجاد کنیم تا آماده حل مسئله انتخاب ویژگی شود.

۵-۷-۲- الگوریتم رقابت استعماری برای حل مسائل انتخاب ویژگی

با اینکه الگوریتم رقابت استعماری، در نسخه های اولیه آن، برای حل مسائل بهینه‌سازی پیوسته معرفی شد، اما از همان ابتدای معرفی، در کارهای پژوهشی متنوعی در حل مسائل گسسته نیز به کار گرفته شده است. در این بخش می‌خواهیم نحوه ایجاد تغییرات در الگوریتم رقابت استعماری پیوسته را جهت تبدیل آن به روشی برای حل مسائل انتخاب ویژگی، بررسی کنیم [۴۵].

در الگوریتم رقابت استعماری پارامتر ها را می‌توان به دو بخش تقسیم بندی نمود.

• **عملگرهای ساختاری و مستقل از مسئله:** عملگرهایی هستند که به نوع کد کردن مسئله وابسته نیستند و ساختار تقریباً معین و ثابتی دارند. مثلاً بخش رقابت استعماری، کل تصمیم‌گیری را از روی هزینه کشورها و امپراطوری‌ها انجام می‌دهد و نوع کد شدن و شیوه نمایش کشورها را در فرآیند خود دخالت نمی‌دهد.

عملگرهای وابسته به مسئله: این عملگرها، باید وابسته به مسئله طراحی شوند. مثلاً عملگر جذب^۱ را در نظر بگیرید. جذب در مسائل پیوسته به این صورت است که کشور مستعمره در راستای برداری از خود به سمت استعمارگر، با روند خاصی (+)، حرکت می‌کند. این حرکت باعث نزدیکی اقلیدسی مستعمره به استعمارگر می‌شود. اما در یک مسئله گسسته، نزدیکی اقلیدسی، لزوماً معنی نزدیکی و نزدیک شدن را نمی‌دهد و حتی در بعضی موارد، نزدیکی اقلیدسی فاقد هرگونه معنی خاصی می‌باشد. مثلاً نزدیک شدن شهر شماره ۱۲ به شهر شماره ۱۷ و مثلاً رسیدن به شهر شماره ۱۴,۷۵ معنی ندارد.

در استفاده از الگوریتم رقابت استعماری در حل مسائل گسسته، تنها بخش‌هایی که باید تغییر داده شوند، عملگرها و بخش‌های وابسته به مسئله هستند. این بخش‌ها عبارتند از:

- ایجاد جمعیت اولیه
- عملگر جذب
- عملگر انقلاب

^۱ Assimilation

۵-۷-۲-۱ - ایجاد جمعیت اولیه

در غالب اوقات، کافی است که متغیر بهینه‌سازی را تنها در مرحله ارسال به تابع هزینه، گسسته‌سازی کنیم. بنابراین کدها را دستکاری نمی‌کنیم و فرض می‌کنیم با یک مسئله پیوسته سرو کار داریم و تمام عملیات، برای مسئله پیوسته انجام می‌شوند. تنها کاری که می‌کنیم این است که یک تابع واسط میان تابع هزینه اصلی و برنامه الگوریتم رقابت استعماری قرار می‌دهیم. این تابع متغیر پیوسته را در هنگام ارزیابی میزان هزینه، گسسته کرده و به تابع هزینه مسئله بهینه‌سازی، ارسال نموده و میزان هزینه آن را دریافت می‌کند. سپس این مقدار هزینه را به الگوریتم، جهت ادامه عملیات برمی‌گرداند. الگوریتم بدون اینکه متوجه شود که مشغول حل یک مسئله گسسته است، کار خود را ادامه داده و به پایان می‌رسد. در نهایت مقادیر بهینه ای را که الگوریتم داده است و پیوسته هستند، با همان تابع، گسسته می‌کنیم و به عنوان جواب اصلی مسئله گسسته بر می‌گردانیم. اما ما در اینجا به این شکل عمل می‌کنیم که جمعیت اولیه از همان ابتدا بصورت رشته های تصادفی از صفر ها و یک ها تولید شوند. در این قسمت می‌توانستیم رشته های صفر و یک را طوری ایجاد کنیم تا تعداد یک ها همواره عدد ثابتی باشند. به این معنا که در انتخاب ویژگی، همواره تعداد مشخصی ویژگی انتخاب شوند، مثلاً ۶۴ ویژگی از مجموع ۱۲۰ یا اما با این کار ممکن بود زیرمجموعه ویژگی‌های به دست آمده کاملاً بهینه نباشد و از این کار صرف نظر کردیم. پس برای تابع تولید جمعیت اولیه از یک تابع تولید استفاده کردیم که رشته های از ۰ و ۱ را با احتمال یکنواخت ایجاد می‌کرد.

۵-۷-۲-۲ - عملگر جذب

با در نظر گرفتن ماهیت مسئله و شیوه مورد استفاده برای کد کردن آن، باید روش مناسبی برای نزدیک کردن مستعمره در نظر گرفته شود. این روش باید به گونه ای روی مستعمره عمل کند که مستعمره جدید، حالتی میانی بین مستعمره قبلی و استعمارگر داشته باشد. از دیدگاه

اطلاعاتی، موقعیت جدید باید هم اطلاعات قبلی خود را داشته باشد هم بخشی از اطلاعات استعمارگر را. شاید ساده ترین کار در این بخش، استفاده از عملگر تقاطع^۱ از الگوریتم ژنتیک باشد. بدین ترتیب در یک الگوریتم مرکب "استعماری - ژنتیک"، ساختار کل مسئله و سایر عملگرهای الگوریتم رقابت استعماری حفظ می شود. ولی در بخش جذب، عملگر تقاطع از الگوریتم ژنتیک استفاده می شود. این مورد در برخی از مقالات، با موفقیت مورد استفاده قرار گرفته است. ما نیز از همین روش در انجام این پایان نامه استفاده کردیم. البته کارهای بسیار متنوعی را در این بخش می توان انجام داده و به نسخه های متفاوتی از الگوریتم رسید.

۵-۷-۲-۳- عملگر تقاطع

تقاطع در الگوریتم ژنتیک، عملگر اصلی ایجاد نسل جدید در مرحله تکثیر است. همانند کروموزوم ها در طبیعت، فرزندان حاصل از این عمل، هر یک بخشی از اطلاعات روی کروموزوم های والدین را دارند [۴۴]. ساده ترین نوع تقاطع، تقاطع یک نقطه ای است. نوع دیگر عملگر تقاطع، تقاطع چند نقطه ای می باشد که در آن تعدادی نقطه روی کروموزوم انتخاب شده و آن را به چند قسمت تقسیم می کند. سپس قسمت های مشابه از کروموزوم ها یکی در میان با هم عوض می شوند. اولین قسمت هر کروموزوم نگه داشته می شود. در شکل (۵-۱۱) این شیوه نشان داده شده است.

$$\left\{ \begin{array}{l} ۱۱۱۰.۱۰۱۱۰۰.۱.۱۰۰۰۱۰۱ \\ ۰.۱۰۰.۱۱۱۰۰۰.۰.۱۰۱۱۰۰ \end{array} \right\} \Rightarrow \left\{ \begin{array}{l} ۱۱۱۰.۱۱۱۰۰۰.۱۰۰۰۱۰۱ \\ ۰.۱۰۰.۱۰۱۱۰۰.۱.۰۱۰۱۱۰۰ \end{array} \right\}$$

شکل ۵-۱۱ تقاطع چند نقطه ای

به کمک تقاطع چند نقطه ای، فضای مسئله بهتر مورد جستجو قرار می گیرد و این امر باعث ثبات الگوریتم می گردد. بنابراین در تقاطع یک نقطه ای و چند نقطه ای تعدادی نقطه تعریف

^۱ Crossover

می‌شود که کروموزوم در آن شکسته شده و اطلاعات آن با کروموزوم دیگری تعویض می‌گردد. در تقاطع یکنواخت این حالت گسترش یافته است. یک رشته تصادفی از ارقام یک و صفر به طول کروموزوم ایجاد می‌شود. حال خانه های مشابه این سه کروموزوم را در نظر می‌گیریم. اگر در رشته تصادفی جدید عدد یک وجود داشت تغییری به وجود نمی‌آید. اگر عدد رشته تصادفی صفر باشد، آنگاه ارقام خانه های دو کروموزوم با هم عوض می‌شوند. شکل زیر این حالت را نشان می‌دهد.

$$\begin{Bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \end{Bmatrix} * \begin{Bmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \end{Bmatrix}$$

شکل ۵-۱۲ تقاطع یکنواخت

و اما روشی که ما برای اصلاح عملگر جذب از روی تقاطع الگوریتم ژنتیک انتخاب می‌کنیم اینگونه است که ابتدا یک رشته تصادفی از صفر ها و یک ها به طول تعداد کل ویژگی‌ها ایجاد می‌کنیم. البته در این رشته تصادفی همواره تعداد ۱ ها را تعدادی معین در نظر می‌گیریم، برابر با نصف تعداد کل ویژگی‌ها. سپس آن متغیر از کشور مستعمره که به ازای آن رشته ی ایجاد شده ۱ بود را مقدارش را عوض کرده و برابر با مقدار معادل آن در موقعیت امپراتوری قرار می‌دهیم. مثلاً اگر تعداد کل ویژگی‌ها ۸ باشد، رشته ایجاد شده می‌تواند بصورت زیر باشد.

$$1 \ 0 \ 1 \ 0 \ 0 \ 1 \ 1 \ 0$$

واگر داشته باشیم:

$$1 \ 0 \ 0 \ 1 \ 1 \ 0 \ 1 \ 1 \quad \text{موقعیت امپراتوری}$$

$$0 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0 \quad \text{موقعیت مستعمره}$$

بیت هایی که با رنگ زرد مشخص شده‌اند همان ۱ ها در رشته تصادفی هستند که مقادیرشان در موقعیت مستعمره با مقدار متناظر موقعیت امپراتوری عوض می‌شود.

پس از اعمال عملگر جذب خواهیم داشت:

موقعیت امپراتوری ۱ ۱ ۰ ۰ ۱ ۰ ۰

موقعیت مستعمره ۱ ۰ ۰ ۱ ۰ ۰ ۱ ۰

۵-۷-۲-۴ - عملگر انقلاب

عملگر انقلاب، نیاز به توضیحات زیادی ندارد. اگر بخش "ایجاد جمعیت اولیه" به خوبی پیاده‌سازی شده باشد، این عملگر، بعضی از کشورهای تصادفی را انتخاب می‌کند و آنها را بصورت تصادفی با موقعیت جدید، جایگزین می‌نماید. جهت ایجاد موقعیت جدید، از همان تابع ایجاد کننده جمعیت اولیه استفاده می‌شود. بنابراین اگر مقصود استفاده از همان عملگر انقلاب موجود در نسخه استاندارد الگوریتم باشد، عموماً نیازی به انجام تغییرات عمده در تابع مربوطه نیست.

۵-۷-۳ - تعیین تابع هزینه مناسب برای مرحله انتخاب ویژگی

در هر مسئله بهینه‌سازی، طراحی تابع هزینه مناسب، یک چالش بزرگ است. ما در این تحقیق از دو تابع هزینه متفاوت برای الگوریتم رقابت استعماری استفاده می‌کنیم، تا با مشاهده نتایج مربوط به آنها بتوانیم در شرایط مختلف، تابع هزینه مناسب را انتخاب کنیم. در یک مرتبه، انتخاب ویژگی را با تابع هزینه‌ای بر اساس خطای شبکه عصبی انجام می‌دهیم و بار دیگر این عمل بر اساس تابع هزینه‌ای با معیار فیشر (Fisher) صورت می‌گیرد. مطابق آنچه در فصل دوم آورده شد، یکی از این توابع با معیار خطای طبقه بندی و دیگری با معیاری مبتنی بر فاصله می‌باشد.

در بخش ۲-۷، پایگاه داده‌ی GPDS۳۰۰ GRAY معرفی شد و اشاره شد که در نسخه استفاده شده‌ی آن برای این پایان نامه، تعداد افراد امضاء کننده ۳۰۰ نفر می‌باشد، که برای هر فرد ۳۰ امضاء صحیح و ۲۴ امضاء جعلی، موجود است.

در این پایان نامه برای مرحله انتخاب ویژگی، از امضاءهای ۵۰ نفر اول این پایگاه داده استفاده می‌کنیم و پس از انتخاب زیرمجموعه‌ی بهینه از ویژگی‌ها توسط الگوریتم رقابت استعماری، این ویژگی‌ها را برای ۲۵۰ نفر باقیمانده در پایگاه داده، بکار می‌بریم.

۵-۷-۴ - تابع هزینه بر اساس خطای شبکه عصبی

همانطور که در بخش ۵-۶ گفتیم، برای آموزش شبکه عصبی متناظر برای هر فرد، از ۱۲ امضاء صحیح و ۱۵ امضاء جعلی استفاده می‌کنیم. باقیمانده امضاءها در مرحله آزمون بکار برده می‌شود و ارزیابی با معیارهای خطای معرفی شده در بخش ۵-۷ صورت می‌گیرد.

در اینجا برای هر زیرمجموعه از ویژگی‌ها، هزینه بدین صورت تعریف می‌گردد که ابتدا برای آموزش شبکه عصبی، این ویژگی‌ها بکار برده می‌شوند و در مرحله ارزیابی خطای EER محاسبه می‌شود. در نهایت، این مقدار خطای EER متناظر با هر زیرمجموعه ویژگی به عنوان هزینه آن زیرمجموعه، معرفی می‌شود.

۵-۷-۵ - تابع هزینه بر اساس معیار Fisher

در این قسمت تابع هزینه را بر اساس تابع معیار Fisher تعریف می‌کنیم. برای این کار، اینبار وارد مرحله ارزیابی نمی‌شویم و هزینه را تنها از داده‌هایی که برای فاز آموزش در نظر گرفته ایم، به دست می‌آوریم.

به دست آوردن هزینه بر اساس این معیار، بکمک ماتریس‌های کواریانس که پراکندگی داده‌ها را در فضای ویژگی‌ها محاسبه می‌کنند، انجام می‌گیرد [۴۶].

مطابق معیار Fisher، اطلاعات کلاس‌ها در دو ماتریس پراکندگی که در روابط (۳-۵) و (۴-۵) معرفی می‌شوند، لحاظ می‌شود.

$$\Sigma_W = \sum_i P(w_i) * \Sigma_i \quad (3-5)$$

$$\Sigma_B = \sum_i P(w_i) * \Delta_i * \Delta_i^T \quad (4-5)$$

Σ_i ماتریس کواریانس کلاس i ام می‌باشد که شامل اطلاعات تغییرات نمونه‌های هر کلاس حول مقدار میانگین آن یعنی μ_i است و $P(w_i)$ احتمال پیشین هر کلاس را نشان می‌دهد. Δ_i نیز بردار $(m_i - m.)$ است که بردار میانگین کلاس i ام و $m.$ بردار میانگین همه‌ی نمونه‌ها بدون در نظر گرفتن اینکه متعلق به کدام کلاس هستند، می‌باشد.

Σ_W ، بدلیل اینکه تفاوت بین نمونه‌های موجود در هر کلاس را محاسبه می‌کند، ماتریس پراکندگی درون کلاسی و Σ_B که فاصله بین بردارهای میانگین را محاسبه می‌کند، ماتریس پراکندگی بین کلاسی، نامیده می‌شود.

برای یک I منحصر بفرد که معرف یک زیرمجموعه از ویژگی‌ها می‌باشد، شاخص تفکیک مطابق رابطه (5-5) تعریف می‌شود.

$$J(I) = \text{trace}(\Sigma_W^{-1} * \Sigma_B) \quad (5-5)$$

عملگر trace استفاده شده در رابطه (5-5)، دلالت بر مجموع درایه های قطر اصلی یک ماتریس، دارد.

مقدار بالای این شاخص نشان می‌دهد که میانگین کلاس‌ها به اندازه کافی از هم فاصله دارند و نیز نمونه‌های هر کلاس با فاصله مناسبی حول میانگین پخش شده اند.

و اما برای مسئله تأیید امضاء، برای هر نفر، دو کلاس وجود دارد (کلاس امضاءهای اصلی و کلاس امضاءهای جعلی). چون برای هر فرد ۱۲ امضاء جعلی و ۱۵ امضاء اصلی برای مرحله آموزش در نظر گرفته شده است، احتمال پیشین برای یک کلاس، $\frac{12}{27}$ و برای کلاس دیگر $\frac{15}{27}$ می‌باشد.

در این مورد ابتدا برای هر فرد $J(I)$ را بکمک روابط (۵-۳) تا (۵-۵) محاسبه می‌کنیم و سپس مقدار هزینه هر زیرمجموعه از ویژگی‌ها را بصورت مجموع $J(I)$ های تمامی افراد، باز می‌گردانیم.

۵-۸- نتیجه گیری

در این فصل الگوریتم پیشنهادی برای تشخیص صحت امضاء بصورت برون خط را بصورت کامل توضیح دادیم. همانطور که گفته شد پس از مرحله پیش پردازش امضاءها، آنها را بصورت شبکه‌ای وفقی تقسیم‌بندی کردیم تا استخراج ویژگی انجام شود. سپس به کمک الگوریتم رقابت استعماری ویژگی‌های بهینه را انتخاب نمودیم. برای طبقه بندی نیز از شبکه‌های عصبی استفاده کردیم که نتایج مربوط به شبیه سازی این الگوریتم به کمک این کلاسه‌بند، در فصل بعد آورده می‌شود.

پیاده‌سازی الگوریتم پیشنهادی و ارزیابی نتایج

۶-۱- مقدمه

در این فصل نتایج پیاده‌سازی الگوریتم تشریح شده در فصل چهارم بر روی پایگاه داده GPDS۳۰۰GRAY آورده می‌شود و نتایج آن با کارهای مشابه مقایسه می‌گردد. ابتدا برای بررسی سیستم پیشنهادی تأیید امضاء، نرخ خطای تأیید آن را با دیگر روش‌ها مقایسه می‌کنیم. سپس با توجه به نتایج به دست آمده و مقایسه‌ای که با دیگر کارهای مرتبط صورت گرفت، نتیجه‌گیری نهایی انجام می‌شود و پیشنهاداتی برای بهبود کارایی الگوریتم ارائه می‌کنیم.

لازم به ذکر است که روش مورد ارزیابی در محیط نرم افزاری MATLAB Version Core (TM)۲ Duo CPU T۹۵۵۰ @ ۲٫۶۶GHz (R۲۰۱۰a) ، ۷،۱۰،۰،۴۹۹ با پردازنده اینتل به مشخصات RAM=۴،۰۰ GB و ۲،۶۷GHz پیاده‌سازی شده‌اند.

۲-۶- نتایج شبیه سازی الگوریتم پیشنهادی برای پایگاه داده

GPDS۳۰۰GRAY

در این قسمت الگوریتم شرح داده شده در فصل قبل را برای دو حالت استخراج ویژگی شبکه‌ای سنتی و شبکه‌ای وفقی، با طبقه بند شبکه عصبی، بر روی پایگاه داده‌های GPDS۳۰۰GRAY آزمایش می‌کنیم و نتایج آنها را در جدول های مربوطه ارائه می‌کنیم. در پایان نیز نتایج به دست آمده را با هم مقایسه می‌کنیم.

۱-۲-۶- نتایج شبیه سازی برای استخراج ویژگی بصورت سنتی

در فصل گذشته گفته شد که برای هر امضاء چه ویژگی‌هایی را استخراج می‌کنیم. با تقسیم بندی M در N تعداد ویژگی‌های استخراجی برابر با $(M*N*5) + 6$ خواهد بود. پارامتر اول تعداد ویژگی‌های محلی را نشان می‌دهد. عدد ۶ نیز تعداد ویژگی‌های سراسری استخراج شده است.

ما در اینجا نتایج را برای استخراج ویژگی بصورت شبکه بندی سنتی، برای شبکه با اندازه های گوناگون و همچنین تقسیم بندی دایره ای در جدول های متناظر می‌آوریم.

به دلیل اینکه شبکه عصبی مصنوعی، مقداردهی اولیه‌اش بصورت تصادفی می‌باشد پس در هر بار آموزش، ضرایب و بایاس های شبکه به یک مقدار همگرا می‌شوند و این باعث تفاوت جواب ها در هر بار اجرا در مرحله آزمون می‌شود. به همین علت ما برای به دست آوردن نتایج، ۱۰ بار برنامه را اجرا می‌کنیم و سپس از جواب ها میانگین می‌گیریم و آنرا بعنوان جواب نهایی در جدول مربوطه قرار می‌دهیم.

همچنین تعداد نرون های لایه ی پنهان را برای به دست آوردن کمترین میزان خطای سیستم تغییر می‌دهیم. جدول های (۱-۶) تا (۶-۶) خطا بازای تغییر این نرون ها و برای تقسیم بندی

های مختلف را نشان می‌دهد. اعداد مندرج در جدول‌ها بر اساس درصد می‌باشند. در هر جدول بهترین جواب بازای یک تعداد نرون معین، با ضخامت بیشتری نشان داده شده است.

جدول ۱-۶ نتایج مربوط به تقسیم بندی شبکه ۱*۲ سنتی

تعداد نرون	۴	۶	۸	۱۰	۱۲	۱۴	۱۶	۱۸	۲۰
FAR	۱۱,۲۶	۱۱,۴۰	۱۱,۴۸	۱۰,۸۹	۱۱,۳۴	۱۱,۳۵	۱۰,۹۲	۱۰,۷۱	۱۰,۸۷
FRR	۷,۷۷	۷,۹۶	۸,۰۱	۷,۵۳	۷,۱۴	۸,۰۴	۷,۳۵	۷,۶۴	۷,۴۶
EER	۱۱,۱۲	۱۱,۴۵	۱۱,۵۰	۱۰,۹۱	۱۱,۳۳	۱۱,۴۱	۱۰,۹۳	۱۰,۷۱	۱۰,۸۸

جدول ۲-۶ نتایج مربوط به تقسیم بندی شبکه ۳*۳ سنتی

تعداد نرون	۴	۶	۸	۱۰	۱۲	۱۴	۱۶	۱۸	۲۰
FAR	۱۲,۹۸	۱۲,۵۹	۱۱,۵۰	۱۱,۲۴	۱۱,۱۶	۱۰,۲۹	۱۱,۴۸	۹,۷۸	۹,۸۴
FRR	۷,۷۲	۷,۳۸	۷,۰۸	۶,۴۵	۶,۴۲	۶,۴۰	۶,۳۲	۶,۱۹	۶,۰۰
EER	۱۲,۹۴	۱۲,۶۳	۱۱,۵۳	۱۱,۱۵	۱۱,۰۷	۱۰,۳۴	۱۱,۴۱	۹,۸۸	۹,۹۳

جدول ۳-۶ نتایج مربوط به تقسیم بندی شبکه ۳*۴ سنتی

تعداد نرون	۴	۶	۸	۱۰	۱۲	۱۴	۱۶	۱۸	۲۰
FAR	۱۴,۰۲	۱۲,۱۴	۱۳,۰۱	۱۱,۹۰	۱۲,۰۳	۱۱,۳۷	۱۱,۲۹	۱۰,۸۴	۱۱,۸۷
FRR	۸,۵۴	۷,۱۶	۷,۱۴	۶,۷۱	۶,۴۸	۶,۳۷	۶,۱۳	۶,۰۳	۶,۰۰
EER	۱۴,۰۲	۱۲,۱۸	۱۲,۹۳	۱۱,۹۴	۱۱,۹۳	۱۱,۳۸	۱۱,۱۹	۱۰,۷۱	۱۱,۷۴

جدول ۴-۶ نتایج مربوط به تقسیم بندی شبکه ۴*۴ سنتی

تعداد نرون	۴	۶	۸	۱۰	۱۲	۱۴	۱۶	۱۸	۲۰
FAR	۱۴,۷۰	۱۳,۱۷	۱۳,۷۰	۱۳,۸۳	۱۳,۲۵	۱۲,۴۸	۱۲,۵۳	۱۲,۷۲	۱۲,۴۰
FRR	۸,۲۸	۷,۶۴	۶,۸۵	۷,۰۱	۶,۱۶	۶,۶۶	۵,۶۳	۵,۶۰	۵,۲۶
EER	۱۴,۸۶	۱۳,۰۵	۱۳,۵۴	۱۳,۹۵	۱۳,۲۶	۱۲,۴۷	۱۲,۷۱	۱۲,۵۳	۱۲,۳۵

جدول ۵-۶ نتایج مربوط به تقسیم بندی شبکه ۵*۵ سنتی

تعداد نرون	۴	۶	۸	۱۰	۱۲	۱۴	۱۶	۱۸	۲۰
FAR	۱۵,۶۶	۱۴,۷۶	۱۵,۸۷	۱۴,۳۳	۱۲,۸۰	۱۳,۴۹	۱۳,۷۵	۱۴,۵۵	۱۵,۳۴
FRR	۱۱,۳۷	۹,۶۲	۷,۹۸	۷,۹۷	۷,۵۶	۷,۱۹	۸,۰۴	۷,۸۳	۷,۳۰
EER	۱۵,۷۵	۱۴,۷۶	۱۵,۷۴	۱۴,۳۶	۱۲,۷۲	۱۳,۳۰	۱۳,۷۵	۱۴,۳۱	۱۵,۴۲

جدول ۶-۶ نتایج مربوط به تقسیم بندی شبکه دایره ای ۸ قطاعی

تعداد نرون	۴	۶	۸	۱۰	۱۲	۱۴	۱۶	۱۸	۲۰
FAR	۸,۰۹	۹,۶۲	۱۰,۰۵	۹,۳۱	۱۰,۳۷	۹,۶۸	۹,۴۱	۹,۵۲	۸,۸۳
FRR	۷,۲۴	۷,۴۲	۶,۶۱	۶,۹۳	۶,۶۶	۷,۳۵	۶,۵۶	۵,۸۷	۶,۴۵
EER	۸,۱۴	۹,۶۰	۱۰,۱۸	۹,۱۷	۱۰,۴۷	۹,۵۲	۹,۴۴	۹,۵۲	۸,۹۴

۶-۲-۲- نتایج شبیه سازی برای استخراج ویژگی با روش پیشنهادی

در این بخش برای استخراج ویژگی از روش پیشنهادی ارائه شده در فصل قبل استفاده

می کنیم.

تقسیم بندی ها متناظر با بخش قبل صورت گرفته و تعداد ویژگی های استخراج شده برای مقایسه با روش تقسیم بندی سنتی، همان $(M*N*5)+6$ برای یک شبکه M در N می باشد. همانند بخش قبل به ازای نرون های متفاوت، سیستم را ارزیابی می کنیم. نتایج مربوط به تقسیم بندی های گوناگون برای مرحله استخراج ویژگی را می توانید در جداول (۶-۷) تا (۶-۱۰) مشاهده کنید.

جدول ۶-۷ نتایج مربوط به تقسیم بندی شبکه ۱*۲ و فقی

تعداد نرون	۴	۶	۸	۱۰	۱۲	۱۴	۱۶	۱۸	۲۰
FAR	۱۴,۸۱	۱۲,۸۵	۱۲,۱۶	۱۰,۵۸	۱۱,۱۶	۱۱,۰۵	۱۱,۰۵	۱۱,۱۱	۱۰,۹۵
FAR	۱۱,۹۵	۱۱,۱۱	۱۰,۰۵	۸,۶۷	۹,۳۱	۷,۶۷	۸,۶۷	۸,۶۷	۸,۲۰
FAR	۱۴,۸۵	۱۱,۵۶	۱۳,۷۲	۱۱,۷۸	۱۱,۰۹	۱۱,۳۲	۱۱,۲۸	۱۰,۴۰	۱۱,۱۸

جدول ۶-۸ نتایج مربوط به تقسیم بندی شبکه ۳*۳ و فقی

تعداد نرون	۴	۶	۸	۱۰	۱۲	۱۴	۱۶	۱۸	۲۰
FAR	۱۰,۶۱	۱۱,۸۸	۱۳,۱۹	۱۴,۷۷	۱۴,۶۷	۱۶,۲۲	۱۸,۱۶	۱۸,۴۴	۱۸,۰۲
FAR	۶,۶۳	۵,۹۹	۵,۱۴	۴,۷۲	۵,۰۷	۴,۱۶	۴,۵۸	۴,۳۰	۴,۲۳
FAR	۸,۵۵	۸,۵۱	۸,۳۵	۸,۵۳	۸,۹۴	۸,۵۵	۹,۳۴	۸,۸۳	۹,۰۶

جدول ۶-۹ نتایج مربوط به تقسیم بندی شبکه ۳*۴ و فقی

تعداد نرون	۴	۶	۸	۱۰	۱۲	۱۴	۱۶	۱۸	۲۰
FAR	۷,۰۳	۷,۳۲	۷,۷۲	۷,۷۲	۷,۸۵	۷,۵۷	۸,۲۰	۸,۷۵	۸,۸۶
FAR	۶,۶۹	۶,۱۳	۵,۵۲	۴,۸۶	۵,۰۲	۴,۶۸	۴,۶۲	۸,۷۵	۸,۸۶
FAR	۷,۵۰	۷,۳۴	۷,۸۱	۷,۸۷	۸,۰۱	۸,۵۴	۸,۱۳	۸,۷۵	۸,۸۸

جدول ۱۰-۶ نتایج مربوط به تقسیم بندی شبکه ۴*۴ و فقی

تعداد نرون	۴	۶	۸	۱۰	۱۲	۱۴	۱۶	۱۸	۲۰
FAR	۷,۳۷	۸,۶۷	۸,۱۸	۷,۹۷	۱۰,۲۶	۹,۷۷	۱۰,۰۸	۹,۹۸	۹,۹۱
FAR	۶,۰۰	۵,۹۲	۵,۱۸	۴,۲۶	۲,۷۹	۴,۴۰	۵,۲۵	۴,۶۹	۴,۹۷
FAR	۷,۴۹	۸,۶۷	۸,۱۶	۷,۷۹	۱۰,۳۱	۱۰,۰۱	۱۰,۱۰	۹,۷۵	۹,۹۲

۳-۲-۶ - مقایسه روش پیشنهادی برای تقسیم بندی امضاء با روش سنتی

تقسیم بندی برای مرحله استخراج ویژگی

در این بخش برای مقایسه و نتیجه گیری، بهترین جواب های هر قسمت را که در بخش های

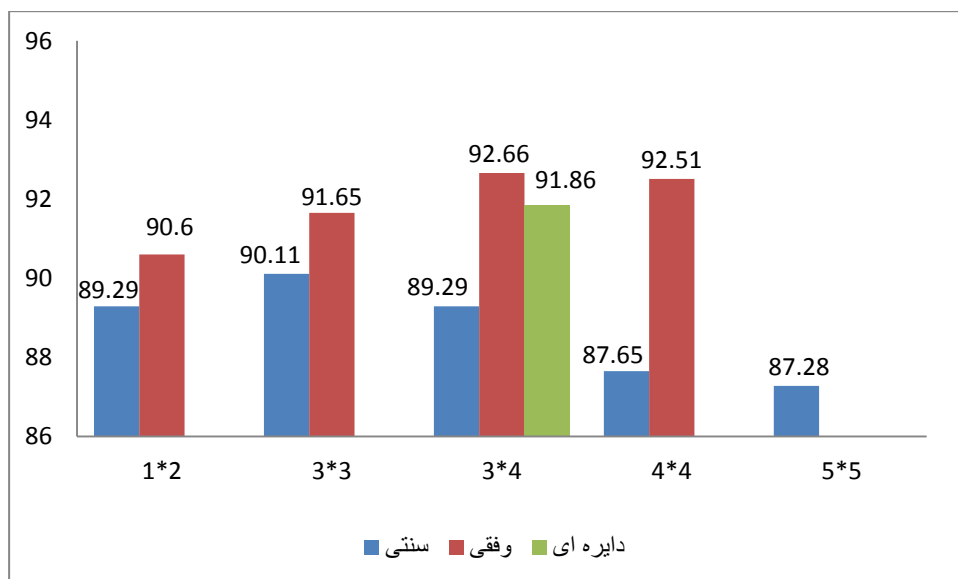
پیش مشخص شده بودند، در کنار هم در جدول (۱۱-۶) قرار می دهیم.

جدول ۱۱-۶ نتایج خطای EER بازای تقسیم بندی های گوناگون

نوع تقسیم بندی شبکه	EER (%)	تعداد نرون
شبکه ۱*۲ سنتی	۱۰,۷۱	۱۸
شبکه ۳*۳ سنتی	۹,۸۸	۱۸
شبکه ۳*۴ سنتی	۱۰,۷۱	۱۸
شبکه ۴*۴ سنتی	۱۲,۳۵	۲۰
شبکه ۵*۵ سنتی	۱۲,۷۲	۱۲
شبکه دایره ای	۸,۱۴	۴
شبکه ۱*۲ سنتی	۱۰,۴	۱۸
شبکه ۳*۳ سنتی	۸,۳۵	۱۰
شبکه ۳*۴ سنتی	۷,۳۴	۶
شبکه ۴*۴ سنتی	۷,۴۹	۴

حال برای اینکه دید بهتری برای مقایسه داشته باشیم، Acc روش‌های گفته شده را در شکل

زیر نمایش می‌دهیم.



شکل ۶-۱ مقایسه روش تقسیم بندی پیشنهادی با روش‌های تقسیم بندی سنتی و دایره ای

همانطور که در شکل (۶-۱) مشخص است، بازای شبکه با اندازه های مساوی، همواره تقسیم بندی به روش پیشنهادی نتیجه بهتری نسبت به حالت سنتی دارد. همچنین تقسیم بندی به روش دایره‌ای با تقسیم بندی ۸ تایی نیز توانسته نتیجه بالاتری از کلیه حالات سنتی، داشته باشد. در نهایت می‌توانیم بصراحت ادعا کنیم که با روش پیشنهادی خطای EER را حدود ۱,۰۱ درصد نسبت به روش سنتی و حدود ۰,۸ درصد نسبت به روش دایره ای کم کردیم.

همانطور که مشاهده می‌کنید در تمامی تقسیم بندی ها به روش سنتی، تعداد نرون لایه مخفی برای رسیدن به کمترین میزان خطای EER بیشتر از ۱۰ می‌باشد. ولی در روش دایره ای تعداد نرون بهینه، ۴ است که عددی بسیار کمتر از مرحله قبل است. همچنین برای روش تقسیم بندی پیشنهادی نیز می‌توان دید، تقسیم بندی هایی که خطای کمتری دارند، تعداد نرون هایشان نیز عددی کمتر از ۱۰ است. هرچه نرون ها بیشتر باشد، آموزش و آزمون شبکه عصبی به زمان بیشتری نیاز دارد. پس این نتیجه خوبی است که با رسیدن به خطای کمتر، همزمان تعداد نرون های کمتری، مورد نیاز است. زیرا که زمان آموزش و آزمون را کاهش می‌دهد.

حال ما الگوریتمی که بهترین نتیجه را در این قسمت نشان می‌دهد، بر می‌گزینیم و از ویژگی‌های استخراج شده به کمک آن، در مرحله ارزیابی نتایج انتخاب ویژگی به کمک الگوریتم رقابت استعماری، استفاده می‌کنیم.

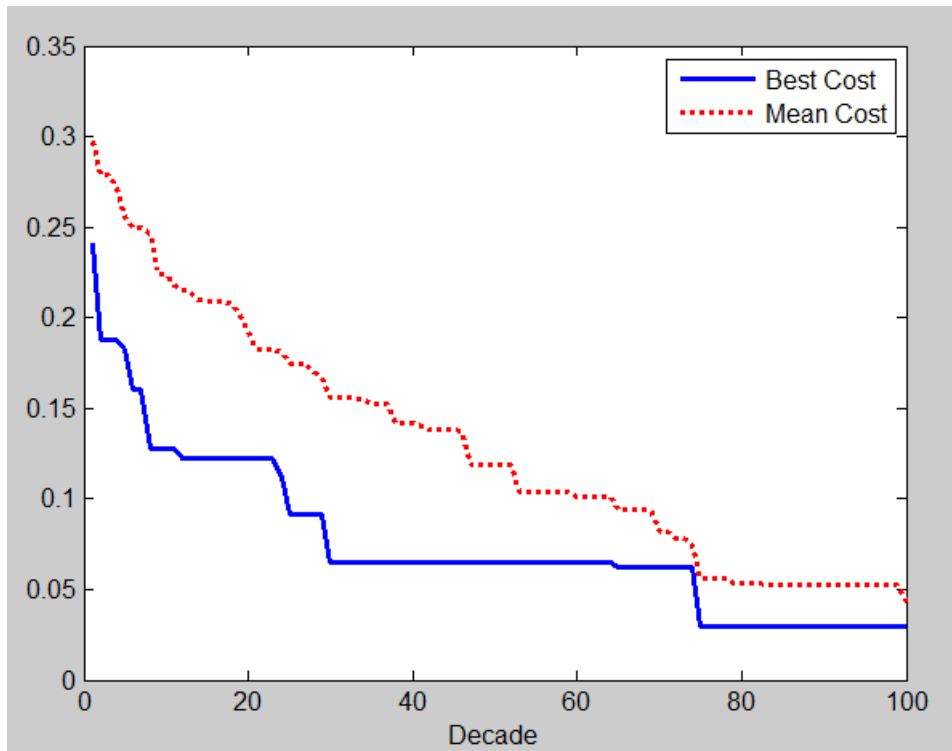
۶-۳- نتایج شبیه سازی بازای ویژگی‌های انتخاب شده با الگوریتم رقابت

استعماری

در بخش قبل تقسیم بندی ۳ در ۴ به روش پیشنهادی، بهترین نتیجه را به دست آورد. پس ما نیز ابتدا به کمک آن روش، ویژگی‌ها را استخراج می‌کنیم و سپس به کمک الگوریتم رقابت استعماری از میان آن ویژگی‌ها، زیرمجموعه بهینه را انتخاب می‌نماییم. همانطور که در فصل گذشته گفته شد، برای تابع ارزیابی یکبار از خطای شبکه عصبی و یکبار از معیار Fisher استفاده می‌کنیم و نتایج آنها را در جدول های مربوطه ارائه می‌کنیم.

در الگوریتم رقابت استعماری استفاده شده در این تحقیق، تعداد کل کشورها را ۱۰۰ و تعداد امپراتوری‌ها را ۱۰ در نظر گرفتیم. همچنین بنابر مشاهدات به دست آمده از روی تحقیقات گوناگون، بهترین مقدار برای ضریب مشارکت مستعمرات در هزینه کل امپراتوری‌ها ۰,۱ می‌باشد.

شکل (۶-۲) روند به دست آوردن بهترین زیرمجموعه از ویژگی‌ها را برای رسیدن به هزینه کمتر نشان می‌دهد.



شکل ۶-۲ روند بهینه‌سازی به کمک الگوریتم رقابت استعماری با گذشت دهه‌ها

۶-۳-۱ نتایج شبیه‌سازی با ویژگی‌های انتخاب شده‌ی الگوریتم رقابت

استعماری با تابع معیار خطای شبکه عصبی

برای پارامتر نرخ انقلاب، اکثر تحقیقات مقدار ۰٫۲ را پیشنهاد می‌دهند. ما نیز در این بخش به دلیل اینکه انتخاب خطای طبقه بندی بعنوان تابع برازش الگوریتم‌های انتخاب ویژگی، زمان زیادی برای اجرا می‌خواهد (در حدود چند روز)، تنها این مقدار را برای این قسمت در نظر گرفتیم و مقادیر دیگر را امتحان نمی‌کنیم. اما در بخش بعد که تابع برازش، معیار Fisher می‌باشد، سه مقدار را برای نرخ انقلاب در نظر می‌گیریم تا اثر تغییر این پارامتر در نتایج مشخص شود.

قبل از اجرای الگوریتم، تعداد ویژگی‌های مورد استفاده $66 = 6 + 5 * 4 * 3$ عدد بود. پس از اجرای الگوریتم رقابت استعماری با این تابع معیار، الگوریتم رشته‌ای را پیشنهاد داد که بر اساس آن ۳۰ ویژگی به عنوان ویژگی‌های بهینه انتخاب شدند که نتایج حاصل از استفاده از این زیرمجموعه ویژگی‌ها، در جدول (۶-۱۲) آورده شده است.

جدول ۱۲-۶ نتایج استفاده از ویژگی‌های انتخاب شده به کمک ICA با تابع معیار خطای شبکه عصبی

تعداد نرون	۴	۶	۸	۱۰	۱۲	۱۴	۱۶	۱۸	۲۰
FAR (%)	۱۵,۹۷	۱۶,۹۳	۱۸,۶۲	۱۸,۸۳	۲۰,۹۵	۲۱,۲۶	۲۳,۲۸	۲۱,۴۸	۲۰,۶۳
FAR (%)	۶,۰۳	۳,۸۰	۳,۴۹	۳,۷۰	۳,۱۷	۴,۵۵	۳,۴۹	۳,۷۰	۲,۵۳
FAR (%)	۷,۳۵	۶,۹۳	۷,۵۶	۷,۴۰	۷,۶۱	۸,۱۴	۷,۷۸	۷,۵۱	۷,۰۳

زمان اجرای الگوریتم رقابت استعماری با تابع برازش خطای شبکه عصبی در حدود ۷۱۶۴۹,۶۶۵۱۸ ثانیه می‌باشد.

۲-۳-۶- نتایج شبیه سازی با ویژگی‌های انتخاب شده الگوریتم رقابت

استعماری با تابع معیار Fisher

در این قسمت همانطور که در فصل قبل توضیح داده شد، از معیار Fisher برای تابع هزینه استفاده کردیم و الگوریتم را برای ۳ مقدار ۰,۱ و ۰,۲ و ۰,۳ برای پارامتر نرخ انقلاب اجرا کردیم. در هر ۳ مورد تعداد ویژگی‌های انتخاب شده ۲۵ مورد بودند.

ما ویژگی‌های پیشنهادی الگوریتم را برای هر سه مقدار به طور جداگانه به سیستم اعمال کردیم. نتایج حاصل را در جدول‌های (۱۳-۶)، (۱۴-۶) و (۱۵-۶) مشاهده می‌کنید.

جدول ۱۳-۶ نتایج استفاده از ویژگی‌های انتخاب شده به کمک ICA با تابع معیار Fisher و نرخ انقلاب ۰,۱

تعداد نرون	۴	۶	۸	۱۰	۱۲	۱۴	۱۶	۱۸	۲۰
FAR	۱۶,۵۰	۱۷,۲۴	۱۸,۷۳	۱۹,۸۹	۲۱,۲۶	۲۰,۷۴	۲۱,۴۸	۲۳,۳۸	۲۱,۷۹
FAR	۶,۰۳	۵,۵۰	۴,۷۶	۵,۱۸	۳,۹۱	۴,۳۳	۴,۸۶	۵,۵۰	۴,۱۲
FAR	۸,۴۸	۸,۰۹	۸,۲۰	۸,۶۲	۸,۲۵	۸,۵۱	۸,۹۴	۹,۶۲	۹,۲۵

جدول ۱۴-۶ نتایج استفاده از ویژگی‌های انتخاب شده به کمک ICA با تابع معیار fisher و نرخ انقلاب ۰,۲

تعداد نرون	۴	۶	۸	۱۰	۱۲	۱۴	۱۶	۱۸	۲۰
FAR	۱۵,۵۵	۱۸,۳۰	۲۰,۴۲	۲۰,۱۰	۲۱,۳۷	۲۱,۷۹	۲۰,۷۴	۲۲,۴۳	۲۳,۰۶
FAR	۵,۶۰	۷,۰۸	۴,۳۳	۶,۷۷	۵,۲۹	۶,۴۵	۴,۴۴	۴,۹۷	۴,۱۲
FAR	۷,۷۷	۹,۱۰	۸,۲۰	۹,۳۱	۹,۴۷	۹,۷۳	۹,۵۲	۱۰,۲۱	۷,۸۱

جدول ۱۵-۶ نتایج استفاده از ویژگی‌های انتخاب شده به کمک ICA با تابع معیار fisher و نرخ انقلاب ۰,۳

تعداد نرون	۴	۶	۸	۱۰	۱۲	۱۴	۱۶	۱۸	۲۰
FAR	۱۶,۵۰	۱۸,۹۴	۲۰,۱۰	۱۹,۳۶	۲۰,۵۲	۲۱,۵۸	۲۱,۴۸	۲۲,۰۱	۲۲,۷۵
FAR	۷,۰۷	۶,۳۴	۴,۸۶	۵,۸۲	۴,۲۳	۴,۲۳	۴,۴۴	۳,۷۰	۴,۳۳
FAR	۸,۲۰	۹,۶۸	۸,۴۱	۸,۶۲	۸,۳۵	۸,۳۵	۸,۴۱	۸,۳۰	۹,۵۲

زمان اجرای الگوریتم رقابت استعماری با تابع برازش معیار Fisher در حدود ۱۱۳۴,۸۷۵ ثانیه

می‌باشد.

۳-۳-۶ - مقایسه نتایج استفاده از ویژگی‌های انتخاب شده توسط الگوریتم

رقابت استعماری بازای تابع هزینه های متفاوت

برای نشان دادن اثر تابع هزینه برای الگوریتم رقابت استعماری بعنوان الگوریتم انتخاب ویژگی،

خطای حاصل از استفاده‌ی ویژگی‌های پیشنهادی متناظر با توابع هزینه به کار برده شده در

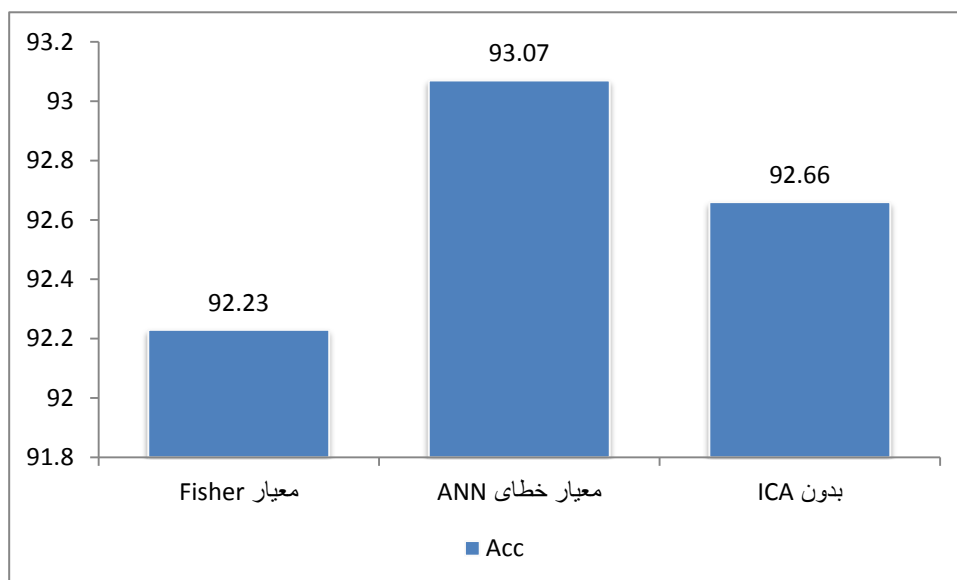
بخش‌های قبل را در کنار هم در جدول (۱۶-۶) نمایش می‌دهیم.

جدول ۱۶-۶ زمان و خطای EER حاصل از استفاده ویژگی‌های انتخاب شده به کمک ICA بازای نرخ انقلاب و

توابع هزینه گوناگون

نوع تابع بهینه	خطای شبکه عصبی	معیار fisher نرخ انقلاب ۰,۱	معیار fisher نرخ انقلاب ۰,۲	معیار fisher نرخ انقلاب ۰,۳
EER	۶,۹۳	۸,۰۹	۷,۷۷	۸,۲۰
زمان لازم برای انتخاب ویژگی (ثانیه)	۷۱۶۴۹,۶۶۵	۱۱۳۴,۸۷۵		

حال باید ببینیم انتخاب ویژگی در بهبود جواب سیستم چه تاثیری دارد. چون برای مرحله ارزیابی از داده‌هایی استفاده می‌کنیم که در مرحله انتخاب ویژگی خطای طبقه بندیشان استفاده نشده است، پس لزوماً نباید انتظار داشته باشیم همواره با استفاده از ویژگی‌های انتخاب شده خطا کاهش بیابد. با این حال استفاده از الگوریتم رقابت استعماری هنگامی که تابع هزینه اش را خطای شبکه عصبی در نظر گرفته بودیم، خطای EER را حدود ۰,۴۱ درصد کاهش داد. بر عکس در حالی که تابع هزینه معیار Fisher بود، خطا بمیزان ۰,۴۳ درصد زیاد شد. در شکل (۳-۶) کارایی سیستم را قبل و بعد از استفاده از الگوریتم رقابت استعماری با هر دو تابع هزینه نشان می‌دهیم.



شکل ۳-۶ مقایسه کارایی سیستم با و بدون استفاده از الگوریتم رقابت استعماری و بازای توابع هزینه‌های

گوناگون

درست همانگونه که در فصل دوم در مورد الگوریتم‌های انتخاب ویژگی گفته شد، دقت الگوریتم‌های انتخاب ویژگی با خطای طبقه بندی بسیار بالاست. در حالیکه پیچیدگی محاسباتی و زمان محاسبه بالایی دارند. اما اگر از معیار وابستگی برای هزینه استفاده شود، پیچیدگی محاسباتی و زمان محاسبه پایین می‌آید، که این به قیمت کاهش دقت سیستم می‌شود.

حال برای انتخاب سیستم مناسب، کارایی و زمان لازم جهت آموزش سیستم‌های فوق الذکر را بطور همزمان در جدول (۶-۱۷) نشان دهیم.

جدول ۶-۱۷ مقایسه سیستم در حالت های با و بدون استفاده از الگوریتم رقابت استعماری از نظر زمانی و دقت سیستم

نوع سیستم	زمان استخراج ویژگی از داده‌های آموزش (ثانیه)	زمان لازم برای انتخاب ویژگی	Acc
بدون انتخاب ویژگی	۳۷۵,۶	۰	۹۲,۶۶
با انتخاب ویژگی با معیار خطای ANN	۲۷۳,۳	۷۱۶۴۹,۶۶۵	۹۳,۰۷
با انتخاب ویژگی با معیار Fisher	۱۸۵,۵	۱۱۳۴,۸۷۵	۹۲,۲۳

نتایج کلی نشان می‌دهد، در جاییکه می‌خواهیم خطا به کمترین مقدار برسد و از نظر زمانی محدودیتی نداریم، بهتر است از الگوریتم رقابت استعماری با تابع هزینه خطای شبکه عصبی استفاده کنیم. در مواقعی که محدودیت زمانی داریم و افزایش ۰,۴۴ درصدی EER را به کاهش زمان ترجیح می‌دهیم از الگوریتم انتخاب ویژگی استفاده نمی‌کنیم. همچنین مواردی وجود دارد که پایگاه داده بسیار بزرگ است و حافظه و زمان کمی در اختیار داریم. در اینجا می‌توانیم با انتخاب ویژگی به کمک معیار Fisher زیرمجموعه کوچکی از ویژگی‌ها را ذخیره کنیم ولی در قبال این کاهش حجمی باز هم خطا افزایش می‌یابد. پس بسته به حساسیت موضوع می‌توانیم از هر یک از سه سیستم فوق استفاده کنیم.

۶-۴ - مقایسه الگوریتم پیشنهادی با دیگر کارهای انجام شده

در تحقیقاتی که در زمینه تأیید امضاء بصورت برون خط انجام گرفته، افراد معمولاً از پایگاه داده‌های متفاوت استفاده کرده اند و عدم وجود پایگاه داده استاندارد، سبب می‌شود تا امر مقایسه امکان پذیر نباشد. بعلاوه انتخاب تعداد امضاء برای مرحله آموزش و آزمون نیز در کارهای مختلف، متفاوت است و مقایسه را بمراتب پیچیده تر می‌کند.

ما در این تحقیق به این علت که قصد استخراج یک دسته از ویژگی‌ها بر اساس تصویر امضاء با مقیاس خاکستری را داشتیم، از پایگاه داده GPDS۳۰۰GRAY استفاده کردیم.

متأسفانه در تحقیقات انجام شده در این زمینه علاقه کمتری به بررسی تصویر خاکستری امضاء نشان داده شده است و در بیشتر تحقیقات از نمونه سیاه و سفید مشابه این پایگاه داده یعنی GPDS استفاده می‌شود و نیز همانطور که در فصل دوم گفتیم گردآورنده این پایگاه داده‌ها بمرور زمان پایگاه داده موجود را کامل تر می‌کند و کمتر از گذشت یک فصل، نسخه‌های موجود تغییر می‌کند. همین امر سبب شده که در کارهای انجام شده بر روی این پایگاه داده، ورژن‌های متفاوتی مورد استفاده قرار گیرد. لذا با توجه به موارد فوق به طور دقیق و قطعی نمی‌توانیم نتایج کارهای گوناگون را با هم مقایسه کنیم. با این حال ما در این قسمت تنها برای اینکه دید کلی نسبت به جایگاه الگوریتم پیشنهادی در بین کارهای دیگر داشته باشیم، در جدول (۶-۱۸) بهترین جوابی را که در این پایان نامه به آن دست یافتیم با نتایج ارائه شده در تحقیقاتی که در آنها از پایگاه داده GPDS و نه لزوماً GPDS۳۰۰GRAY استفاده شده است، مقایسه می‌کنیم.

جدول ۶-۱۸ مقایسه الگوریتم پیشنهادی با کارهای مشابه

	[۴۸]	[۴۲]		[۴۷]				الگوریتم پیشنهادی
		A	B	A	B	C	D	
نام پایگاه داده	GPDS-۱۰۰	GPDS-۱۰۴		GPDS-۱۰۰		GPDS-۷۵۰		GPDS۳۰۰ GRAY
تعداد افراد امضاء کننده	۱۰۰	۳۰۰		۱۰۰		۷۵۰		۲۵۰
تعداد امضای اصلی	۲۴	۲۴		۲۴		۲۴		۲۴
تعداد امضای جعلی	۲۴	۲۴		۲۴		۳۰		۳۰
خطای EER (%)	۶,۲۰	۲۰	۲۳,۷۵	۱۵,۲	۱۴,۲۲	۱۷,۳۶	۱۵,۲۵	۶,۹۳

۶-۵ - مقایسه الگوریتم پیشنهادی با سیستم‌های ارائه شده در مسابقه

۴NSigComp۲۰۱۰

در این پایان نامه بازای تقسیم بندی وفقی ۴*۳ برای مرحله استخراج ویژگی و همچنین استفاده از ویژگی‌های منتخب به کمک الگوریتم انتخاب ویژگی رقابت استعماری با تابع هزینه خطای شبکه عصبی، کمترین مقدار خطا $EER=6,96$ به دست آمد. حال سیستم متناظر با این خطای بهینه را برای پایگاه داده مسابقه ۴NSigComp۲۰۱۰ استفاده کرده و نتیجه حاصل را با نتایج افراد شرکت کننده در مسابقه مقایسه می‌کنیم. برای اطلاع از چگونگی انجام مسابقه می‌توانید به پیوست ب مراجعه نمایید. پایگاه داده GPDS۹۶۰ شامل امضاءها بصورت سیاه و سفید است. این امر را بر این داشت تا برای ارزیابی سیستم در مرحله استخراج ویژگی، از آن دسته از ویژگی‌هایی که از تصویر با سطوح خاکستری به دست می‌آید، صرف نظر کنیم.

در پیوست ب قوانین مسابقه آورده شده است. متأسفانه امکان دسترسی مجموعه امضاء با جعل تصادفی وجود نداشت. پس ما تنها برای ارزیابی سیستم از معیار FAR استفاده کردیم. بعلاوه سیستم پیشنهادی ما برای مرحله آموزش نیاز به امضاءهای جعلی هم داشت. در حالیکه در این مسابقه در مرحله آموزش تنها از امضاءهای اصلی استفاده می‌شود. این بدان معناست که بطور دقیق نمی‌توانیم قوانین مسابقه را رعایت کنیم. با این حال مقایسه نتیجه الگوریتم پیشنهادی با نتایج تیم های شرکت کننده می‌تواند یک دید کلی نسبت به عملکرد سیستم پیشنهادی بدهد. در جدول (۶-۱۹) می‌توانید نتایج مربوط به این مسابقه را مشاهده نمایید. چون بر خلاف مسابقه، معیار ارزیابی در اینجا معیار خطای FRR است. در این جدول رتبه بندی سیستم‌ها را بر اساس همین معیار مشخص کردیم.

جدول ۶-۱۹ جدول نرخ خطای FRRS برای سیستم‌های شرکت داده شده در مسابقه ۲۰۱۰ NSigComp و سیستم ارائه شده در این پایان نامه

رتبه در مسابقه	FRRS	شناسه سیستم
۱	۷,۸۱	۶
	۱۱,۸۷	سیستم پیشنهادی این پایان نامه
۳	۲۲,۸۴	۹
۲	۲۵,۳۸	۳
۸	۲۸,۳۶	۴
۷	۲۸,۴	۳
۹	۲۹,۳۵	۵
۱۰	۲۹,۷۳	۷
۴	۳۹,۱۸	۱
۶	۴۶,۲۷	۲
۵	۴۷,۸	۱۰

روش پیشنهادی ما در بین تیم های شرکت کننده در این مسابقه در مرتبه ۲ قرار گرفت.

۶-۶- نتیجه گیری

هدف ما در این پایان نامه ارائه یک الگوریتم مناسب برای سیستم تأیید امضاء برون خط می باشد. برای انجام این کار ما در مرحله استخراج ویژگی برای اینکه ویژگی های محلی را استخراج کنیم، روش های تقسیم بندی سنتی را کنار گذاشتیم و روشی جدید برای شبکه بندی تصویر امضاء ارائه کردیم. نتایج انجام شده بر روی پایگاه داده GPDS300 GRAY نشان داد که این نوع تقسیم بندی همواره خطای کمتری نسبت به هم نوع سنتی خود دارد. در مرحله بعد از الگوریتم رقابت استعماری برای انتخاب ویژگی های مناسب از امضاء استفاده کردیم که منجر به استخراج 30 ویژگی برای هر امضاء شد. با انتخاب این 30 ویژگی توانستیم خطای سیستم را باز پایین بیاوریم. بطوریکه در نهایت به نرخ خطای $EER=6,96$ دست یافتیم. مزایای استفاده از این الگوریتم پیشنهادی نسبت به الگوریتم های مشابه، دقت بالاتر الگوریتم است، علت این امر به دلیل نوع جدید تقسیم بندی تصویر امضاء است که منجر به استخراج اطلاعات مفید تر از هر سلول شبکه می شود و دلیل دیگر نیز استفاده از الگوریتم رقابت استعماری برای انتخاب ویژگی های بهینه می باشد.

۶-۷- پیشنهادات

در این تحقیق برای آنکه بتوانیم روش تقسیم بندی پیشنهادی را با روش‌های سنتی مقایسه کنیم، از همپوشانی سلول‌ها با سلول‌های همسایه شان خودداری کردیم. در حالیکه می‌توانست باعث بهبود خطای سیستم شود. پس در ادامه تحقیق می‌توانیم این نوع تقسیم بندی را نیز آزمایش کنیم. همچنین پیشنهاد می‌کنیم تا از این روش برای حالت تقسیم بندی دایره ای نیز استفاده شود.

در این پایان نامه ما از شبکه عصبی مصنوعی برای طبقه بندی استفاده کردیم. استفاده از کلاسه بند های دیگر نظیر ماشین بردار پشتیبان یا کلاسه بند های جدیدتر و همچنین استفاده از نتیجه چندین کلاسه بندی برای تصمیم گیری نهایی نیز می‌تواند پیشنهاد مفیدی باشد.

و در نهایت می‌توان گفت، بدلیل اینکه دانش شناسایی آماری الگو، روز به روز در حال ارائه الگوریتم‌های جدید تر است، مطالعه و به روز بودن می‌تواند ایده های جدیدی چه در مرحله استخراج ویژگی و چه در مرحله طبقه بندی به ما بدهد.

ضمیمه أ - شبکه‌های عصبی

مصنوعی چند لایه

أ-۱ - تاریخچه شبکه‌های عصبی مصنوعی

از قرن نوزدهم به طور همزمان اما جداگانه، از سوی نروفیزیولوژیست‌ها سعی کردند سیستم یادگیری و تجزیه و تحلیل مغز را کشف کنند و از سوی دیگر ریاضیدانان تلاش کردند تا مدل ریاضی بسازند که قابلیت فراگیری و تجزیه و تحلیل عمومی مسائل را دارا باشد. اولین کوشش‌ها در شبیه‌سازی با استفاده از یک مدل منطقی توسط مک کلوک^۱ و پیتز^۲ انجام شد، که امروزه بلوک اصلی سازنده اکثر شبکه‌های عصبی مصنوعی است. این مدل فرضیه‌هایی در مورد عملکرد نرون‌ها ارائه می‌کند. عملکرد این مدل مبتنی بر جمع ورودی‌ها و ایجاد خروجی است. چنانچه حاصل جمع ورودی‌ها از مقدار آستانه بیشتر باشد، اصطلاحاً نرون برانگیخته می‌شود. نتیجه این مدل اجرای توابع ساده مثل AND و OR بود [۴۹]. در همان زمان دانشمندان دیگری مثل دونالد هب^۳ نیز بر روی قوانین تطبیقی در سیستم‌های نرونی کار می‌کردند. در سال ۱۹۴۹، دونالد هب یک قانون یادگیری برای تطبیق ارتباطات بین نرون‌های مصنوعی ارائه نمود.

^۱ McCulloch

^۲ Pitts

^۳ Donald Hebb

نه تنها نروفیزیولوژیست‌ها، بلکه روان‌شناسان و مهندسان نیز در پیشرفت شبیه‌سازی شبکه‌های عصبی تاثیر داشتند. در سال ۱۹۵۸ شبکه پرسپترون^۱ توسط روزنبلات^۲ معرفی گردید. وی تئوری تفکیک آماری را بر مبنای آن توسعه داد.

سیستم دیگر، مدل خطی تطبیقی نرون می‌باشد که در سال ۱۹۶۰ توسط ویدرو^۳ و هاف^۴ از دانشگاه استنفورد به وجود آمد که اولین شبکه‌های عصبی به کار گرفته شده در مسائل واقعی بودند. این طرح موسوم به آدالاین^۵ بود. آدالاین یک دستگاه الکترونیکی بود که از اجزای ساده‌ای تشکیل می‌یافت. روشی که برای آموزش استفاده می‌شد با پرسپترون فرق داشت. در سال ۱۹۷۱، وربوس^۶ یک الگوریتم پس انتشار را در رساله دکتری خود منتشر کرد و در نهایت روزنبلات این تکنیک را در سال ۱۹۸۶ مجدداً کشف نمود [۵۰].

۲-۱- اساس شبکه‌های عصبی مصنوعی

مبحث شبکه‌های عصبی مربوط به شبیه سازی قوه یادگیری در انسان و پیاده سازی آن به صورت الگوریتم‌های کامپیوتری می‌باشد. قوه یادگیری در انسان عبارت است از فراگیری الگوهای اطراف خود بکمک تکرار. مغز به عنوان یک سیستم پردازش اطلاعات با ساختار موازی، از میلیونها نرون^۷ تشکیل شده است. بافت‌هایی که عصب نامیده می‌شوند اجتماعی از نرون‌ها هستند که اطلاعات و پیامها را از یک قسمت بدن به قسمت دیگر منتقل می‌کنند [۵۱].

^۱ Perceptron

^۲ Rosenblatt

^۳ Widrow

^۴ Hoff

^۵ Adaline

^۶ Werbos

^۷ Neuron

أ-٢-١- نرون بیولوژیکی

نرون های بیولوژیکی از چهار قسمت اساسی تشکیل شده اند:

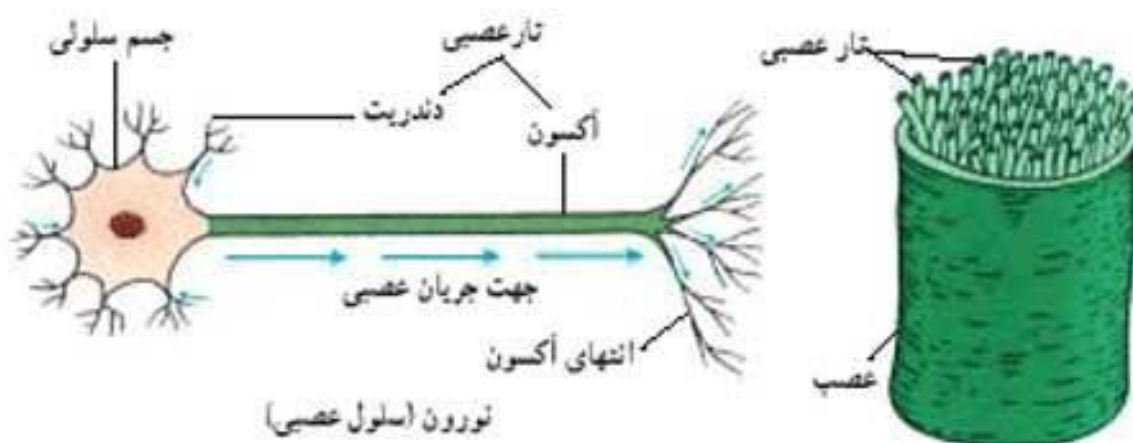
دندریت^١: اطلاعات را دریافت و به هسته سلول می رساند.

بدنه سلول: اطلاعات دریافتی را پردازش می کند.

اکسون^٢: اطلاعات هسته سلول را به نرون های دیگر منتقل می کند.

سیناپس^٣: محل تلاقی یک اکسون از یک نرون به دندریتهای نرون دیگر را سیناپس میگویند.

در شکل (أ-١) می توانید ساختار یک نرون بیولوژیکی را مشاهده نمایید.



شکل الف- ١ نرون بیولوژیکی

یک نرون مصنوعی، در حقیقت مدلی محاسباتی است که از نرون های عصبی واقعی انسان، الهام گرفته شده است. نرون های طبیعی، ورودی خود را از طریق سیناپس دریافت می کنند. این سیناپس ها بر روی دندریت ها یا غشاء عصب قرار دارند. در یک عصب واقعی، دندریت ها، دامنه پالس های دریافتی را تغییر می دهند که نوع این تغییر در طول زمان یکسان نمی ماند و در اصطلاح،

^١ Dendrite

^٢ Axon

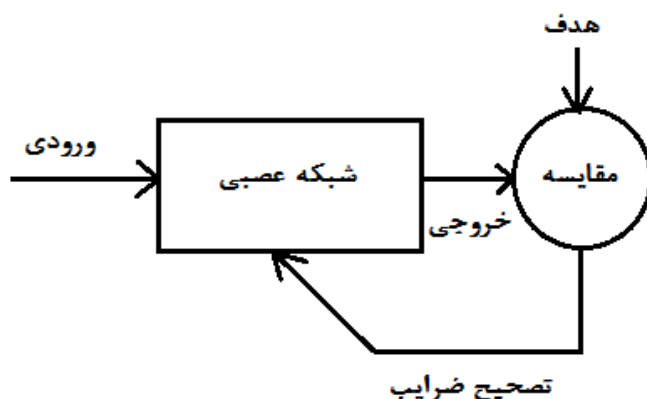
^٣ Sinaps

توسط عصب یاد گرفته می شود. اگر سیگنال دریافتی به حد کافی قوی باشد (از یک مقدار آستانه بیشتر شود)، عصب فعال شده و سیگنالی را در طول اکسون منتشر می کند. این سیگنال نیز به نوبه خود می تواند به یک سیناپس دیگر وارد شده و سایر اعصاب را تحریک کند [۵۲].

گمان می رود که مغز انسان از تعداد ^{۱۱} ۱۰ نرون تشکیل شده باشد که هر نرون با تقریباً ^۴ ۱۰ نرون دیگر در ارتباط است. سرعت سوئیچینگ نرونها در حدود ^۳ ۱۰ ثانیه است که در مقایسه با کامپیوترها ^{۱۰} (۱۰ ثانیه) بسیار ناچیز می نماید. با این وجود آدمی قادر است در ۰,۱ ثانیه، تصویر یک انسان را بازشناسائی نماید. این قدرت فوق العاده باید از پردازش موازی توزیع شده در تعدادی زیادی از نرونها حاصل شده باشد.

۳-۱ - چگونگی کارکرد شبکه های عصبی

شبکه های عصبی از عناصر عملیاتی ساده ای بصورت موازی ساخته می شوند. این عناصر از سیستم های عصبی زیستی الهام گرفته شده اند. در طبیعت ساختار شبکه های عصبی از طریق نحوه اتصال بین اجزاء تعیین می شود. بنابراین ما می توانیم یک ساختار مصنوعی به تبعیت از شبکه های طبیعی بسازیم و با تنظیم مقادیر هر اتصال، تحت عنوان وزن اتصال، نحوه ی ارتباط بین اجزای آن را تعیین نماییم.



شکل الف- ۲ بلوک دیاگرام یک شبکه عصبی

پس از تنظیم یا همان آموزش شبکه عصبی، اعمال یک ورودی خاص به آن منجر به دریافت پاسخ خاص می‌شود. همانطور که در شکل (الف-۲) دیده می‌شود، شبکه بر مبنای تطابق و همسنجی بین ورودی و هدف، سازگار می‌شود تا اینکه خروجی شبکه و هدف بر هم منطبق گردند. عموماً تعداد زیادی از این زوج‌های ورودی و خروجی به کار گرفته می‌شوند، تا در این روند که از آن تحت عنوان یادگیری نظارت شده یاد می‌شود، شبکه آموزش داده شود.

آموزش دسته‌ای شبکه، به معنی تغییر وزن‌ها و یا بایاس‌های شبکه بر مبنای مجموعه‌ی تمام بردارهای ورودی می‌باشد. در حالیکه آموزش گام به گام، وزن‌ها و بایاس‌های شبکه را پس از اعمال تک‌تک بردارهای ورودی تنظیم می‌کند. از آموزش گام به گام تحت عنوان آموزش در خط^۱ یا انطباقی یاد می‌شود.

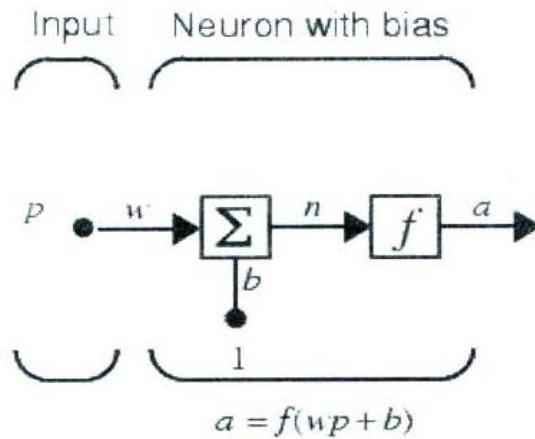
عموماً برای آموزش شبکه‌های عصبی از روش‌های نظارت شده، استفاده می‌شود. اما می‌توان شبکه‌ها را با روش آموزش غیر نظارتی که کاربرد آنها عموماً در تشخیص گروه‌های داده‌ای می‌باشد، نیز آموزش داد. گونه‌های خاصی از شبکه‌های عصبی مانند هاپفیلد^۲ و شبکه‌های خطی، بصورت مستقیم توسط طراح، طراحی می‌شوند. بطور خلاصه می‌توان گفت روش‌های مختلفی برای طراحی تکنیک‌های یادگیری موجود می‌باشد.

أ-۴- معماری شبکه عصبی مصنوعی

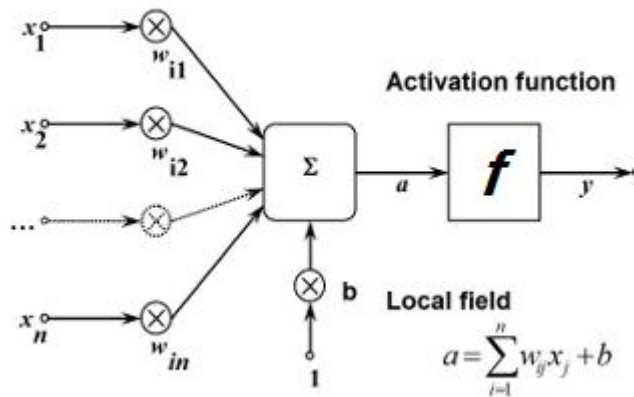
نرون مصنوعی، کوچکترین واحد پردازشگر اطلاعات است. ورودی‌های نرون توسط یک نوع ارتباط به نام وزن، به نرون وارد می‌شوند. در شکل (الف-۳) شمای کلی یک نرون تک لایه با یک ورودی نشان داده شده است. همچنین نمونه‌ای از یک نرون تک لایه با چند ورودی را در شکل (الف-۴) نمایش می‌دهیم.

^۱ Online

^۲ Hopfield



شکل الف- ۳ نرون تک لایه با یک ورودی

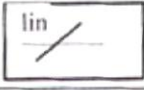
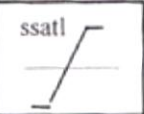
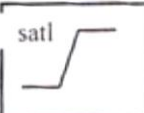


شکل الف- ۴ نرون تک لایه با چند ورودی

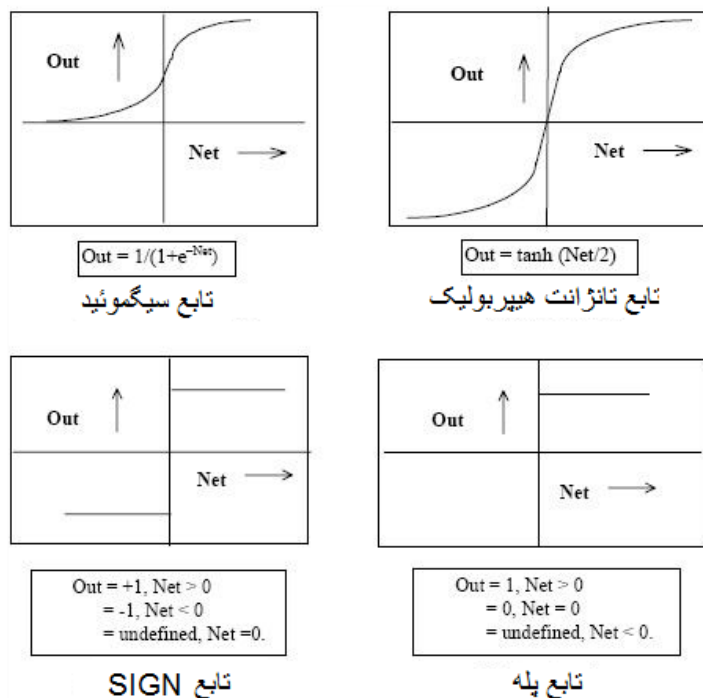
در شکل (الف-۴) تعداد ورودی‌ها برابر با n می‌باشد. w ها وزن ها و F تابع فعالساز خروجی می‌باشد. مقدار خروجی از رابطه‌ی (الف-۱) به دست می‌آید.

$$A = f(w * x + b) \quad \text{(الف - ۱)}$$

هر تابع مشتق‌پذیر و پیوسته‌ای که بطور یکنواخت افزایش یابد، می‌تواند بعنوان تابع محرک استفاده شود. معمول‌ترین توابع، توابع سیگموئید (تانژانت هیپربولیک) و خطی می‌باشند. در شکل (الف-۵) چند نمونه از توابع فعالسازی خطی و در شکل (الف-۶) چند نمونه از توابع فعالسازی غیر خطی نشان داده شده است.

خطی	$a = n$	خطی
		
	$a = -1, n < -1$ $a = n, -1 \leq n \leq 1$ $a = 1, n > 1$	آستانه‌ای خطی متقارن
	$a = 0, n < 0$ $a = n, 0 \leq n \leq 1$ $a = 1, n > 1$	آستانه‌ای خطی

شکل الف-۵ نمونه‌ای از توابع فعالسازی برای شبکه عصبی



شکل الف-۶ رایج‌ترین توابع غیر خطی مورد استفاده به عنوان تابع فعالساز شبکه عصبی

انتخاب نوع تابع محرک، به هدف مسئله بستگی دارد. چنانچه شبکه یک مسئله دسته‌بندی را حل کند، بهتر است از تابع محرک غیر خطی مانند سیگموئیدی برای نرون های لایه پنهان و از تابع خطی برای خروجی استفاده شود.

أ-۵- شبکه‌های عصبی چند لایه

یک شبکه عصبی چند لایه دارای لایه های زیر می‌باشد:

لایه ورودی : برای دریافت اطلاعات خامی که به شبکه تغذیه شده است.

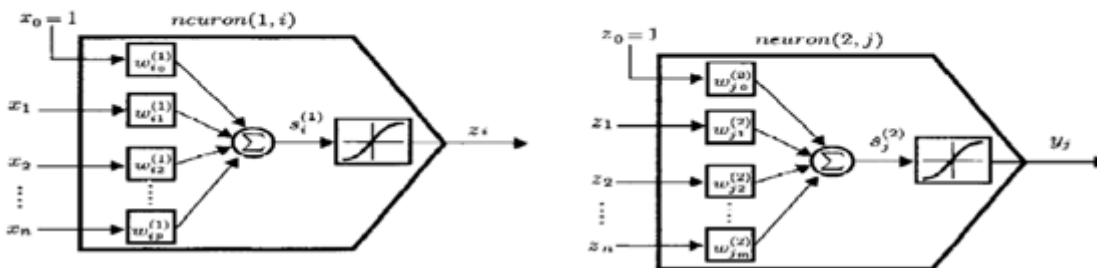
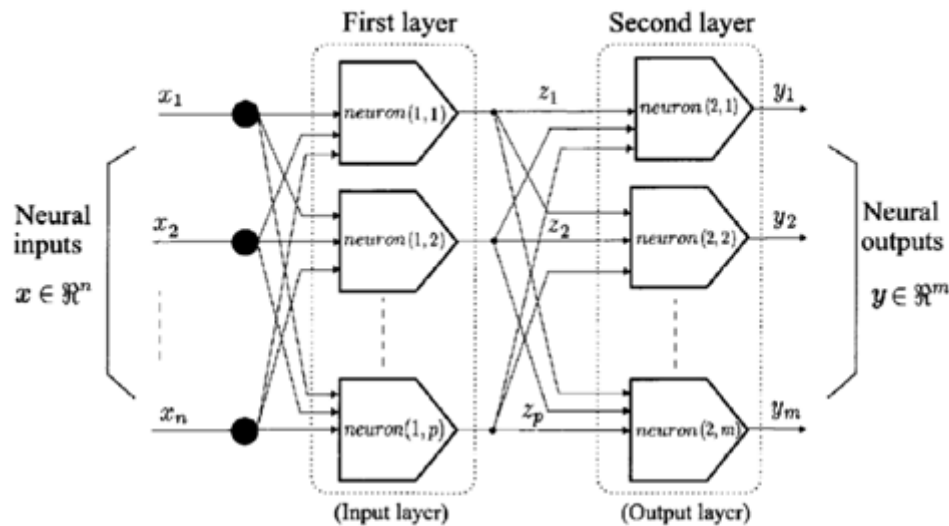
لایه‌های پنهان : عملکرد این لایه‌ها به وسیله ورودی‌ها و وزن ارتباط بین آنها و لایه‌های پنهان

تعیین می‌شود. وزن‌های بین واحدهای ورودی و پنهان، تعیین می‌کند که چه وقت یک واحد

پنهان باید فعال شود.

لایه خروجی : عملکرد واحد خروجی، بسته به فعالیت واحد پنهان و وزن ارتباط بین واحد پنهان و

خروجی می‌باشد.



شکل الف-۷ شبکه عصبی دو لایه

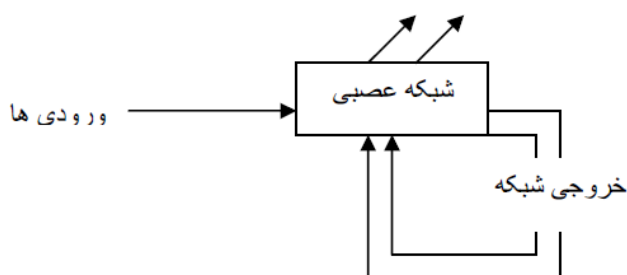
در شکل (الف-۷) یک شبکه عصبی دو لایه نمایش داده شده است.

۶-۱- آموزش شبکه‌های عصبی

الگوریتم‌های آموزش شبکه‌های عصبی به دو گروه اصلی تقسیم می‌شوند [۵۱].

۶-۱-۱ یادگیری نظارت نشده^۱ یا بدون ناظر

در این نوع یادگیری هیچ سیگنالی که اطلاعات را در مورد مطلوبیت جواب شبکه به خود شبکه وارد نماید، وجود ندارد. به عبارت دیگر به شبکه گفته نمی‌شود که خروجی هدف چه است و یا اینکه جواب شبکه چقدر مطلوب است. در این حالت شبکه باید با دریافت اطلاعات ورودی و شاخص‌های موجود در ورودی‌ها، ارتباط موجود بین الگوهای ورودی را پیدا کند و در خروجی‌ها کند. بلوک دیاگرام یادگیری یک شبکه عصبی به روش نظارت نشده را می‌توانید در شکل (الف-۸) مشاهده نمایید.

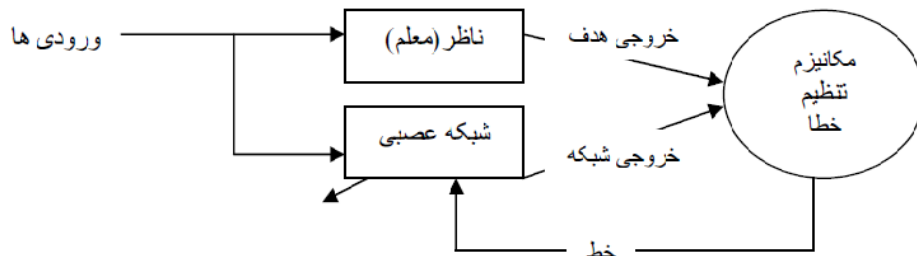


شکل الف-۸ شکل یادگیری نظارت نشده

^۱ Unsupervised Learning

أ- ۶-۲- یادگیری نظارت شده^۱ یا یادگیری با ناظر

در این نوع آموزش به الگوریتم یادگیری، مجموعه‌ای از زوج داده، که به داده‌های یادگیری موسوم هستند، تزریق می‌شود. هر داده یادگیری شامل ورودی به شبکه و خروجی هدف است. پس از اعمال ورودی به شبکه، خروجی شبکه با خروجی هدف مقایسه می‌گردد و سپس خطای یادگیری محاسبه شده و از آن جهت تنظیم پارامترهای شبکه (وزن‌ها) استفاده می‌گردد، به گونه‌ای که اگر دفعه بعد به شبکه همان ورودی را دادیم، خروجی شبکه به خروجی هدف نزدیک گردد. شکل (الف-۹) بلوک دیگرآم یادگیری یک شبکه عصبی به روش نظارت شده را نشان می‌دهد.



شکل

الف- ۹ یادگیری نظارت شده با ناظر

انواع گوناگونی از شبکه‌های عصبی وجود دارند که از آنها می‌توان به شبکه عصبی احتمالی^۲ (PNN)، شبکه عصبی رگرسیون عمومی^۳ (GRNN)، شبکه عصبی پس انتشار^۴ (BNN) و شبکه‌های عصبی شعاعی^۵ (RNN) اشاره کرد.

به علت استفاده از شبکه‌های عصبی پس انتشار در این پایان نامه، در بخش بعدی، این شبکه

بررسی می‌گردد.

^۱ Supervised Learning

^۲ Probabilistic Neural Network

^۳ General Regression Neural Network

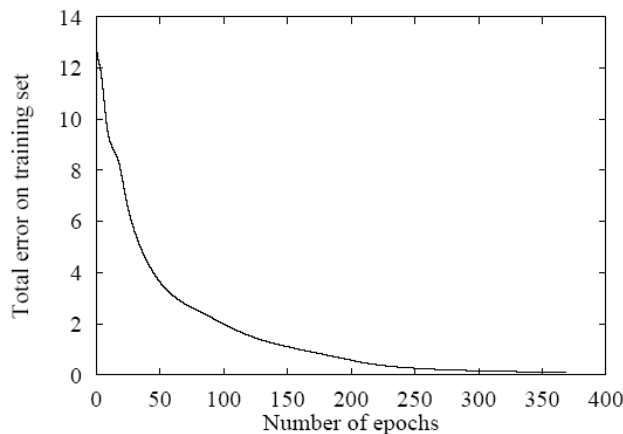
^۴ workBackpropagation Neural Net

^۵ Radial Basis Neural Network

۷-۱- شبکه عصبی پس انتشار (BNN)

این الگوریتم که در سال ۱۹۸۶ توسط روملهارت^۱ و مک‌کلیلاند^۲ پیشنهاد گردید، در شبکه‌های عصبی پیشرو مورد استفاده قرار می‌گیرد. پیشرو بودن به این معناست که نرون‌های مصنوعی در لایه متوالی قرار گرفته‌اند و خروجی سیگنال خود را رو به جلو می‌فرستند. واژه پس انتشار نیز به معنای این است که خطاها به سمت عقب در شبکه تغذیه می‌شوند تا وزن‌ها را اصلاح کنند و پس از آن، مجدداً ورودی، مسیر پیشروی خود تا خروجی را تکرار کند. روش پس انتشار خطا از روش‌های باسرپرست است. به این مفهوم که نمونه‌های ورودی برچسب خورده‌اند و خروجی مورد انتظار هر یک از آنها، از پیش معلوم است. لذا خروجی شبکه با این خروجی‌های ایده آل مقایسه شده و خطای شبکه محاسبه می‌گردد. در این الگوریتم، ابتدا فرض بر این است که وزن‌های شبکه به طور تصادفی انتخاب شده‌اند. در هر گام خروجی شبکه محاسبه شده و بر حسب میزان اختلاف آن با خروجی مطلوب، وزن‌ها تصحیح می‌گردند تا در نهایت این خطا مینیمم شود. در الگوریتم پس انتشار، تابع تحریک هر عصب بصورت جمع وزن دار ورودی‌های مربوط به آن عصب، در نظر گرفته می‌شود.

در شکل (الف-) میانگین مربعات خطا برای ۳۷۰ یادگیری، نشان داده شده است.



شکل الف- ۱۰ میانگین مربعات خطا بر حسب تعداد تکرار برای آموزش

^۱ Rumelhart

^۲ McClelland

مراحل الگوریتم پس انتشار را می توان به صورت زیر بیان کرد:

- شبکه یک نمونه آموزشی را دریافت می کند و با استفاده از وزن های موجود در شبکه، که در ابتدا بصورت تصادفی مقداردهی می شوند، خروجی های شبکه محاسبه می شود.
- خطا یعنی اختلاف بین نتیجه محاسبه شده و مقدار مورد انتظار محاسبه می شود.
- خطا، درون شبکه منتشر می شود و وزن ها برای حداقل نمودن خطا، از نو تنظیم می شوند.

اکنون مراحل انجام الگوریتم پس انتشار را به صورت ریاضی بیان می کنیم. خطای خروجی سلول j در لایه خروجی در تکرار n ام مطابق رابطه (الف-۲) به دست می آید.

$$e_j(n) = d_j(n) - y_j(n) \quad (\text{الف} - ۲)$$

که $d_j(n)$ پاسخ مطلوب و $y_j(n)$ خروجی سلول j است. اگر انرژی خطا برای سلول j ام را به صورت $\frac{1}{2}e_j(n)$ تعریف کنیم، انرژی خطای کل برای تمام سلول ها را می توان از رابطه (الف-۳) بدست آورد.

$$E(n) = \frac{1}{2} \sum_{j=1}^c e_j^2(n) \quad (\text{الف} - ۳)$$

که در آن c ، تعداد سلول های لایه خروجی است. فرض کنید تعداد نمونه ها در داده های آموزشی، برابر با N باشد. میانگین مربعات انرژی خطا را با جمع کردن $E(n)$ ها در نمونه n ام و نرمال کردن آن به تعداد نمونه ها می توانیم مطابق رابطه (الف-۴) بدست آوریم.

$$E_{av} = \frac{1}{N} \sum_{n=1}^N E(n) \quad (\text{الف} - ۴)$$

E_{av} تابعی از تمام پارامترهای آزاد شبکه مانند وزن های اتصال و بایاس ها می باشد. برای یک مجموعه آموزشی E_{av} معرف تابع هزینه (تابع خطا) است. هدف فرآیند یادگیری، حداقل نمودن E_{av} است.

روش ساده آموزش، به روز کردن وزن‌ها به صورت نمونه به نمونه، در طی یک دوره^۱ آموزش است. هر دوره عبارت است از یکبار ارائه کامل همه مجموعه نمونه‌های آموزشی به شبکه.

الگوریتم پس انتشار، مقدار تصحیح وزن معادل $\Delta\omega_{ji}(n)$ را به وزن قبلی $\omega_{ji}(n)$ اضافه می‌کند.

این مقدار تصحیح وزن متناسب است با مشتق جزئی $E(n)$ نسبت به $\omega_{ji}(n)$ ، یعنی $\frac{\partial E(n)}{\partial \omega_{ji}(n)}$.

مشتق جزئی مذکور، جهت جستجو را در فضای وزن‌ها^۲، برای یافتن $\omega_{ji}(n)$ مناسب، تعیین می‌کند.

مقدار تصحیح وزن را می‌توان از روی رابطه (الف-۵) به دست آورد.

$$\Delta\omega_{ji}(n) = -\eta \frac{\partial E(n)}{\partial \omega_{ji}(n)} \quad (\text{الف} - ۵)$$

η را ضریب یادگیری گویند. می‌توان ثابت کرد که مقدار تصحیح وزن در نهایت از رابطه (الف-۶) به دست می‌آید.

$$\Delta\omega_{ji}(n) = \eta \delta_j(n) y_j(n) \quad (\text{الف} - ۶)$$

که در آن $\delta_j(n)$ گرادیان محلی نامیده می‌شود و برای سلول خروجی j ام بکمک رابطه (الف-۷) محاسبه می‌شود.

$$\delta_j(n) = e_j(n) \phi_j(y_j(n)) \quad (\text{الف} - ۷)$$

^۱ Epoch

^۲ Weight Space

در این رابطه $y_j(n)$ مجموع وزن دار سلول j ام است و $\varphi_j(y_j(n))$ تابع انتقال یا محرک به کار رفته می‌باشد. رابطه گرادیان محلی برای سلول‌های خروجی و سلول‌های میانی (لایه‌های پنهان) متفاوت است. رابطه (الف-۷) برای سلول‌های خروجی صادق است. برای سلول j ام طبق رابطه (الف-۸) خواهیم داشت:

$$\delta_j(n) = \varphi_j(y_j(n)) \sum_k \delta_k(n) \omega_{kj}(n) \quad (\text{الف} - ۸)$$

به طور خلاصه، مقدار تصحیح وزن $\Delta\omega_{kj}(n)$ که بر روی اتصال سلول i به j اعمال می‌شود را می‌توان مطابق رابطه (الف-۶) به صورت زیر تعریف کرد.

(ورودی سلول j) . (گرادیان محلی) . (ضریب یادگیری) = (مقدار تصحیح وزن)

گرادیان محلی بسته به اینکه سلول j در لایه خروجی باشد یا لایه‌های میانی به ترتیب از روابط (الف-۷) و (الف-۸) محاسبه می‌شود. محاسبه هر دو این روابط، مستلزم مشتق‌گیری از تابع انتقال است. برای مشتق پذیر بودن تابع انتقال φ لازم است این تابع پیوسته باشد [۵۳].

۸-۱ آموزش شبکه پس انتشار

آموزش شبکه امری وقت گیر می‌باشد. شبکه با چندین تکرار آموزش داده می‌شود و بعد از رسیدن به یک تکرار ماکزیمم، متوقف می‌گردد. تعداد تکرارها به بزرگی شبکه بستگی دارد. بنابراین یک شبکه بزرگ نسبت به یک شبکه کوچکتر، احتیاج به وقت بیشتری برای آموزش دارد. از خطای مجاز مینیمم نیز به عنوان معیار توقف استفاده می‌شود. هرگاه اختلاف بین مقدار خروجی شبکه و مقدار خروجی مطلوب، کمتر از یک حد خاصی شود، یادگیری متوقف می‌شود. در طی آموزش ممکن است شبکه بیش از اندازه تعلیم ببیند. این مشکل به عنوان بیش برآزش^۱، ذکر می‌گردد. بیش برآزش یک مشکل بحرانی در بیشتر ساختارهای شبکه‌های عصبی متعارف است.

^۱ Over Fitting

یک راه حل، توقف سریع آموزش شبکه می‌باشد. استفاده از معیار توقف مناسب، روش دیگری برای جلوگیری از بیش برآزش است.

فرآیند آموزش برای هر کاربر بصورت جداگانه صورت می‌گیرد و شبکه مربوط به هر کاربر پس از آموزش ذخیره می‌گردد. هنگامی که آموزش کامل شد، شبکه برای قابلیت‌های تعمیم یافتن و یادگیری، مورد آزمایش قرار می‌گیرد. آزمایش شبکه توسط آن دسته از ورودی‌ها که در مرحله آموزش استفاده نشده‌اند، انجام می‌شود تا از بیش از حد آموزش دیده شدن شبکه جلوگیری شود [۵۳].

ضمیمه ب - مسابقه تأیید

صحت امضای

برون خط ۲۰۱۰ NSigComp ۴

هدف از این مسابقه اثبات کارایی سیستم‌های تأیید امضای خودکار در ارزیابی توسعه پیشرفت های اخیر تکنولوژی در حوزه تحلیل اسناد و یادگیری ماشین بود [۵۴].

بعد از اعلام مسابقه ۱۵ تیم علاقه خود را برای شرکت در مسابقه اعلام کردند که از آن میان ۷ تیم برنامه هایشان را ارائه دادند (۳ تیم از موسسات علمی و یک تیم از موسسات دولتی و الباقی از مراکز صنعتی). ۲ گروه چندین برنامه ارائه نمودند. بنابراین در این مسابقه شاهد ارزیابی ۱۰ سیستم مختلف بودیم. تیم ها از ۷ کشور هند، کانادا، ترکیه، ایالات متحده آمریکا، استرالیا، اسپانیا و فرانسه بودند. جدول (ب-۱) این گروه ها را همراه با شناسه ای برای برنامه های ارائه شده شان، نشان می دهد.

جدول ب-۱ شرکت کنندگان مسابقه ۲۰۱۰ NSigComp ۴

Institution	Country	Team coordinator	System id
Indian Statistical Institute	India	Rajesh Kumar	1
École de technologie supérieure, Montreal	Canada	Luana Bezerra	2
Sabancı University	Turkey	Berrin Yanikoglu	3,4,5
Parascript LTD	USA	Tim Strunkov	6
Anonymous	Austria	Anonymous	7
Universidad Autónoma de Madrid	Spain	Fernando Alonso-Fernandez	8
NIFISOFT	France	Ali Hassaine	9,10

ب-۱- معرفی پایگاه داده‌های GPDS۹۶۰

پایگاه داده GPDS۹۶۰ امضای برون‌خط، شامل امضای ۹۶۰ نفر می‌باشد. ۲۴ امضای اصلی برای هر نفر به علاوه ۳۰ امضای جعلی. ۲۴ امضای اصلی در یک روز و در یک جلسه از هر فرد جمع آوری شد و امضاهای جعلی از روی تصویر استاتیکی امضای اصلی گرفته شده است. بدین صورت که هر جاعل اجازه داشت به هر تعداد که می‌خواهد برای جعل تمرین کند. هر جاعل ۳ امضاء برای ۵ نفر را در یک روز و در یک جلسه جعل نمود. امضاهای نشان داده شده به هر جاعل بصورت تصادفی از بین ۲۴ امضای اصلی هر فرد انتخاب شده بود. در نهایت برای هر فرد ۳۰ امضای جعلی داریم که توسط ۱۰ فرد مختلف جعل شده‌اند. لازم به ذکر است هر فرد امضاء کننده اجازه داشت از قلم خود برای امضاء کردن استفاده کند.

برای مسابقه ۲۰۱۰ NSigComp۴ از زیرمجموعه پایگاه داده GPDS۹۶۰ استفاده شد.

یکی از پایگاه داده‌های استفاده شده برای این مسابقه، امضای امضاء کنندگان شماره ۳۰۱ تا ۷۰۰ از مجموعه پایگاه داده GPDSsignature بود، که همانطور که گفته شد پایگاه داده GPDS۹۶۰ شامل ۲۴ امضای اصلی و ۳۰ امضای برای هر نفر می‌باشد. از این میان تنها ۴ امضای اصلی از بین ۲۴ امضاء در اختیار شرکت کنندگان این مسابقه قرار گرفت.

برای مرحله ارزیابی نیز از امضای همین افراد استفاده کردند. ۲۰ امضای اصلی باقیمانده هر فرد برای مرحله آزمون سیستم‌ها به کار برده شد. برای جعل حرفه ای نیز که همانطور که در بالا توضیح داده شد، برای هر فرد ۳۰ امضاء موجود بود. در این مسابقه جعل تصادفی نیز ارزیابی شد و برای امضاء با جعل تصادفی از امضاهای جعلی و اصلی امضاء کنندگان شماره‌ی ۷۰۰ تا ۹۶۰ استفاده کردند. پس برای مرحله ارزیابی امضاهای موجود شامل $۲۰ * ۴۰۰ = ۸۰۰۰$ امضای اصلی و $۳۰ * ۴۰۰ = ۱۲۰۰۰$ امضای جعلی حرفه ای و ۱۰۰۰۰ امضاء برای جعل تصادفی در نظر گرفته شد.

در شکل (ب-۱) چند نمونه از امضاءهای این پایگاه داده نمایش داده شده است.



شکل ب- ۱ نمونه امضاء برون خط پایگاه داده

ب-۲- نتایج مسابقه

معیار های خطای مسابقه FRR و FARR و FARS و OE در نظر گرفته شدند. معیار FRR نرخ خطا برای امضاهای اصلی ای که جعلی شناخته شدند، FRRS نرخ خطا برای اصلی شناخته شدن امضاها با جعل حرفه ای و FARR نرخ خطا برای اصلی شناخته شدن امضاهای با جعل تصادفی را نشان می‌دهد. همچنین OE نیز معرف نرخ خطای پذیرفته شدن هر دو نوع جعل می‌باشد.

جدول (ب-۲) نتایج تیم های شرکت کننده در این مسابقه را بر اساس معیارهای بالا نشان می‌دهد.

جدول ب-۲ نتایج تیم های شرکت کتته در مسابقه ۲۰۱۰ NSigComp۴

ID	Coordinator name's	FRR	FARR	FARS	OE	Rank
1	Rajesh Kumar	10.43	9.18	39.18	17.31	4 th
2	Luana Bezerra	22.06	0.07	46.27	22.62	6 th
3	Berrin Yanikoglu	37.86	0.15	28.4	26.07	7 th
4		40.83	0.09	28.36	27.53	8 th
5		41.84	0.09	29.35	28.28	9 th
6	Tim Strunkov	13.96	0.01	7.81	8.94	1 st
7	Anonymous	46.08	0.24	29.73	30.53	10 th
8	Fernando Alonso-Fernandez	18	4.31	25.38	16.42	2 nd
9	Ali Hassaine	21.49	1.22	22.84	16.76	3 rd
10		11.19	0.96	47.8	17.79	5 th

- [۱] Information Technology—Biometric Data Interchange Formats—Part ۱۱: Signature/Sign Processed Dynamic Data, ISO Standard ISO/IEC ,2011
- [۲] M. Ammar, Y. Yoshida, T. fukumara, “ A new effective approach for offline verification of signature by using pressure feature”, IEEE Trans on system, Man and Cybematic, vol.SMC-۱۶, No.۳,pp.۳۹-۴۷,(۱۹۸۶)
- [۳] R. Sabourin, R. Plamondon, “Preprocessing of handwritten signatures from image gradient analysis”, Proc. ۸th Int. Conf. on Pattern Recognition, Paris, pp. ۵۷۶-۵۷۹, (۱۹۸۶).
- [۴] R. Sabourin, G. Genest, “An extended shadow code-based approach for offline signature verification: Part I. Evaluation of the bar mask definition”, Proc. Int. Conf. on Pattern Recognition, Vol. ۲, pp. ۴۵۰-۴۵۳, (۱۹۹۴).
- [۵] R. Sabourin, G. Genest, “An extended shadow code-based approach for offline signature verification: Part II. Evaluation of several multi-classifier combination strategies”, Proc. Int. Conf. on Document Analysis and Recognition, pp. ۱۹۷-۲۰۱, (۱۹۹۵).
- [۶] K. Han, K. Sethi, “Signature identification via local association of features”, Proc. Int. Conf. on Document Analysis and Recognition, pp. ۱۸۷-۱۹۰, (۱۹۹۵).
- [۷] G.D.D.C. Cavalcanti, R.C. Doria, E.Cde.B.C. Filho. “Featur Selection for Offline Recognition of Different Size Signatures,”Proceedings of the ۱۲th IEEE Workshop on Neural Networks for Signal Processing. Martigny, Switzerland, ۲۰۰۲
- [۸] E. Ozgunduz, T. Senturk, E. Karsligil. “OffLine Signature Verification and Identification by Support Vector Machine”. ۱۱th International Conference on Computer Analysis of Images and Patterns. Versailles, France, ۲۰۰۵
- [۹] J. Coetzer, B.M. Herbst, J.A. du Preez. “Offline Signature Verification Using the Discrete Radon Transform and a Hidden Markov Model”. EURASIP Journal on Applied Signal Processing, ۲۰۰۴(۴):۵۵۹-۵۷۱, ۲۰۰۴
- [۱۰] Z. Mohamadi. “Persian Static Signature Recognition”. Thesis of Bachelor of Engineering, Ferdowsi University of Mashhad, March ۲۰۰۶
- [۱۱] M.H. Sigari, M.R. Pourshahabi. “Static Handwritten Signature Identification and Verification”. Thesis of Bachelor of Engineering, Ferdowsi University of Mashhad, July ۲۰۰۶
- [۱۲] M.H. Sigari, M.R. Pourshahabi, H.R. Pourreza. “Offline Handwritten Signature Identification using Grid Gabor Features and Support Vector Machine”. ۱۶th Iranian Conference on Electrical engineering. Tehran, Iran, ۲۰۰۸
- [۱۳] " صفا بخش محرابیان تشخیص صحت امضاءهای فارسی،" سومین مجموعه مقالات سومین کنفرانس بین المللی سالانه انمجن کامپیوتر ایران، دانشگاه علم و صنعت ایران ، تهران ، صص ۱۹۸-۲۰۵ ، دی ماه ۱۳۷۶
- [۱۴] E.Özgündüz,T.Şentürk ,M. Elif Karsligil ,” offline signature verification and recognition by support vector machine ,” Computer Engineering Department, Yıldız Technical University , Istanbul, Turkey,2009

- [۱۵] M.FAKHLAI, H.POURREZA, “ Off line Signature Recognition Based on Wavelet, Curvelet and Contourlet Transforms,” 11th WSEAS International Conference on SIGNAL PROCESSING, COMPUTATIONAL GEOMETRY and ARTIFICIAL VISION (ISCGAV'08),2008
- [۱۶] Daramola Samuel et. al. “novel feature extraction technique for off-line signature verification system”, / International Journal of Engineering Science and Technology Vol. ۲(۷), ۲۰۱۰, ۳۱۳۷-۳۱۴۳
- [۱۷] Piotr Porwik ,”The compact three stages method of the signature recognition,” Institute of Informatics, University of Silesia, Będzińska ۳۹, ۴۱-۲۰۰ Sosnowiec, Poland, 2007
- [۱۸] D.Kisku, P.Gupta, J.Sing “ Offline Signature Identification by Fusion of Multiple Classifiers using Statistical Learning Theory ” International Journal of Security and Its Applications Vol. ۴, No. ۳, July, ۲۰۱۰
- [۱۹] آ. استاد زاده ، ش. استاد زاده " طراحی یک سیستم چند سطحی تأیید امضای دستنویس ایستا بر پایه تطبیق ککشان با درجه تفکیک پذیری چند گانه " مجله فنی مهندسی دانشگاه آزاد اسلامی مشهد دوره اول شماره اول تابستان ۱۳۸۵.
- [۲۰] Stuart Inglis ,Ian H. Witten, “Compression-based Template Matching”, Proc. IEEE Data Compression Conference, pp. ۱۰۶-۱۱۵, Los Alamitos, CA, ۱۹۹۴.
- [۲۱] Peter ShaoHua Deng, et al, “Wavelet-based off-line handwritten signature verification”, Computer vision and image understanding , Vol.۷۶, Issue ۳, pp. ۱۷۳-۱۹۰, Dec ۱۹۹۹
- [۲۲] Edson J. R. Justino, Abdenaimel Yacoubi, Flaviob Ortolozzi and Roberts Abourin, “An Off-Line Signature erification System Using Hidden Markov Model and Cross-Validation”, in IEEE International Workshop on Neural Networks for Signal Processing, pp. ۸۵۹-۸۶۸, Sydney, Australia, December ۲۰۰۰.
- [۲۳] D. Bhattacharyya, S. Kumar Bandyopadhyay, P. Das, D.Ganguly, S. Mukherjee, “Statistical Approach for Offline Handwritten Signature Verification,” Journal of Computer Science ۴ (۳): ۱۸۱-۱۸۵, ۲۰۰۸ ISSN ۱۵۴۹-۳۶۳۶© ۲۰۰۸ Science Publications.
- [۲۴] A. Pérez-Hernández, A. Sánchez and J.F. Vélez, “Simplified Stroke-based Approach for Off-line Signature Recognition ,” University of Rey Juan Carlos, Spain, ۲۸۹۳۳ Móstoles (Madrid),2005
- [۲۵] V.Nguyen, M.Blumenstein,G.Leedham,” Global Features for the Off-Line Signature Verification Problem”, ۲۰۰۹ ۱۰th International Conference on Document Analysis and Recognition
- [۲۶] D.uppalapti,” integration of offline and online signature verification systems” Indian institute of technology, Kanpur, July ۲۰۰۷
- [۲۷] B.Kovari, A.Horvath, B.Toth, H.Charaf,”Local Feature Based Off-line Signature Verification using Neural Network Classifiers” Proceedings of the ۱۱th WSEAS Int. Conf. on MATHEMATICAL METHODS, COMPUTATIONAL TECHNIQUES AND INTELLIGENT SYSTEMS,2009
- [۲۸] D.uppalapti,” integration of offline and online signature verification systems” Indian institute of technology, Kanpur, July ۲۰۰۷
- [۲۹] M.Bhuyan, K.kumar sarma, H das, “ Signature Recognition using Hybrid Feature and clustered Artificial Neural Network (ANN)s,” Interbational Journal and Information engineering ۴:۴ ۲۰۱۰

[۳۰] Debasish Jena, Banshidhar Majhi, Sanjay Kumar Jena, "Improved Offline Signature Verification Scheme Using Feature Point Extraction Method" Journal of Computer Science ۴ (۲): ۱۱۱-۱۱۶, ۲۰۰۸ ISSN ۱۵۴۹-۳۶۳۶ © ۲۰۰۸ Science Publications

[۳۱] <http://www.gpds.ulpgc.e>

[۳۲] M. Dash, H. Liu, "Feature Selection for Classification," Intelligent Data Analysis ۱:۱۳۱-۱۵۶, ۱۹۹۷.

[۳۳] R. R. Palmer, "A History of the Modern World", New York: Alfred A. Knopf, ۱۹۶۴, ©۱۹۵۶.

[۳۴] C. V. Findley & J. A. M. Rothney, Twentieth century world, ۳rd edition, Boston: Houghton Mifflin Company, ۱۹۸۶

[۳۵] اسماعیل آتش یز گرگری، «توسعه الگوریتم بهینه‌سازی اجتماعی و بررسی کارایی آن»، پایان نامه کارشناسی ارشد دانشکده مهندسی برق و کامپیوتر دانشگاه تهران، ۱۳۸۷

[۳۶] T.A. Osman, M.J. Paulik and N.M. Krishnam, "An Online Signature Veification System based on Multivariate Autoregressive Modeling and DTW Segmentation". IEEE SAFE ۲۰۰۷: Signal Processing Applications for Public Security and Forensics, Washington D.C., April 2007

[۳۷] D. Impedovo and G. Pirlo, Automatic signature verification the state of the art, IEEE transactions on systems, man, and cybernetics partC: applications and reviews, vol. ۳۸, NO.۵, September ۲۰۰۸.

[۳۸] E. Justino, F. Bortolozzi, and R. Sabourin, "Off-line signature verification using HMM for random, simple and skilled forgeries," *ICDAR*, vol. ۱, pp. ۱۰۵-۱۱۰, ۲۰۰۱.

[۳۹] L. Oliveira, E. Justino, C. Freitas, and R. Sabourin, "The graphology applied to signature verification," Conference of the International Graphonomics Society, vol. ۱۲, pp. ۲۸۶-۲۹۰, ۲۰۰۵.

[۴۰] E. Justino, F. Bortolozzi, and R. Sabourin, "A comparison of SVM and HMM classifiers in the off-lin signature verification," *Pattern Recognition Letters*, vol. ۲۶, pp. ۱۳۷۷-۱۳۸۵, ۲۰۰۵.

[۴۱] C. Santos, E. Justino, F. Bortolozzi, and R. Sabourin, "An off-line signature verification method based on the questioned document experts approach and a neural network classifier," *IWFHR*, vol. ۹, pp. ۴۹۸-۵۰۲, ۲۰۰۴.

[۴۲] Y. Rekik, N. Houmani, M.A. El Yacoubi, S. Garcia-Salicetti, and B. Dorizzi, " A Comparison of Feature Extraction Approaches for Offline Signature Verification". ۹۷۸-۱-۶۱۲۸۴-۷۳۲-۰/۱/۱/۲۶,۰۰ ©۲۰۱۰ IEEE

[۴۳] J.P. Swanepoel, J. Coetzer, " Off-line Signature Verification Using Flexible Grid Features and Classifier Fusion". ۲۰۱۰ ۱۲th International Conference on Frontiers in Handwriting Recognition: ۹۷۸-۰۰-۷۶۹۵-۴۲۲۱-۸/۱۰ \$۲۶,۰۰ © ۲۰۱۰ IEEE DOI ۱۰,۱۱۰۹/ICFHR.۲۰۱۰,۵۲

[۴۴] ع. رضائی و س. رنجبران، "الگوریتم ژنتیک و فازی در نرم افزار MATLAB"، اسفند ۱۳۸۸

[۴۵] <http://www.icasite.info>

- [۴۶] L. Cordella, C. De Stefano, F. Fontanella and C. Marrocco, "A Feature Selection Algorithm for Handwritten Character Recognition", ۰۸/۶-۲۱۷۵-۴۲۴۴-۱-۹۷۸/۲۵,۰۰\$ ©2008 IEEE
- [۴۷] J.F. Vargas, C.M. Travieso, J.B. Alonso, M.A. Ferrer, "Off-line Signature Verification Based on Gray Level Information using Wavelet Transform and Texture Features", ۹۷۸-۰۰-۷۶۹۵-۴۲۲۱-۸/۱۰ ۰۲۶,۰۰ © ۲۰۱۰ IEEE DOI ۱۰,۱۱۰۹/ICFHR.۲۰۱۰,۹۶
- [۴۸] Jesus F. Vargas, Miguel A. Ferrer, Carlos M. Travieso, Jesus B. Alonso, "Offline Signature Verification Based on Pseudo-Cepstral Coefficients", 0-978 09/2-3725-7695 25.00\$ © 2009 IEEE DOI 10.1109/ICDAR.2009.68
- [۴۹] http://en.wikipedia.org/wiki/Artificial_neural_network
- [۵۰] <http://saba.kntu.ac.ir/eecd/teshnehlab/nnc/fasle۱.pdf>
- [۵۱] hkamal.persianguig.com/document/mahdian.docx

[۵۲] الف. صیادی، "آشنایی با شبکه‌های عصبی مصنوعی"، [ebook]، اسفند ۱۳۸۷،

(http://ee.sharif.ir/~osayadi/Sayadi_NeuralNetwork_Basic_Tutorial.pdf)

- [۵۳] L. Fausett, "Fundamental og Neural Network, arvhitectures, algorithms, and applications:, Prentice-Hall International, 1994
- [۵۴] M. Blumenstein, Miguel A. Ferrer, J.F.Vargas, "The ۴NSigComp۲۰۱۰ off-line signature verification competition: Scenario ۲", ۲۰۱۰ ۱۲th International Conference on Frontiers in Handwriting Recognition, 10/8-4221-7695-0-978 26.00\$ © 2010 IEEE DOI 10.1109/ICFHR.2010.117

واژه نامه انگلیسی به فارسی

Accuracy	دقت
Adaline	آدلاین
Artificial Neural Network	شبکه عصبی مصنوعی
Assimilation	جذب
Axon	اکسون
Backpropagation Neural Network	شبکه عصبی پس انتشار
Base Line	خط اصلی
bottom-up	پایین به بالا
Chain Code	کد زنجیره‌ای
Classifier Error Rate Measures	معیارهای مبتنی بر خطای کلاسه‌بند
Coefficient	ضریب
Colony	مستعمره
Consistency Measures	معیارهای مبتنی بر سازگاری
Correlation	همبستگی
Criterion Function	تابع معیار
Crossover	تقاطع
Dependence Measures	معیارهای مبتنی بر وابستگی
Dendrite	دندریت
Distance Measures	معیارهای مبتنی بر فاصله

Epoch	دوره
Equal Error Rate	نرخ خطای برابر
Evaluation Function	تابع ارزیابی
Flexible Grid Base	شبکه‌ی قابل انعطاف
Gabor Wavelet Transform	تبدیل ویولت گابور
General regression Neural Network	شبکه عصبی رگرسیون عمومی
Generality	عمومیت
Genetic Algorithm	الگوریتم ژنتیک
Grid Base Feature	ویژگی شبکه‌ای
Heuristic Search	جستجوی مکاشفه‌ای
Hidden Marcov Model	مدل مخفی مارکوف
Horizontal Projection	نگاشت افقی
Imperial	امپراتوری
Imperialism	امپریالیسم
Imperialist Competitive Algorithm	الگوریتم رقابت استعماری
Information Measures	معیارهای مبتنی بر اطلاعات
Minimum mean square error	میانگین مربعات خطا
Multilayer	چند لایه
Neuron	نرون
Offline	برون خط
Online	برخط

Overfitting	بیش برآزش
Particle Swarm Optimization Algorithm	الگوریتم بهینه سازی گروه ذرات
Perceptron	پرسپترون
Principal Component Analysis	آنالیز اجزای اصلی
Probabilistic Neural Network	شبکه عصبی احتمالی
Radial Basis Neural Network	شبکه عصبی شعاعی
Radon Transform	تبدیل رادون
Random Search	جستجوی تصادفی
Search Space	فضای جستجو
Sinaps	سیناپس
Static	ایستا
Supervised Learning	یادگیری نظارت شده
Threshold	آستانه
top-down	بالا به پایین
Unsupervised Learning	یادگیری نظارت نشده
Validation Procedure	تعیین اعتبار
Vertical Projection	نگاشت عمودی
Weight Space	فضای وزن‌ها

واژه نامه فارسی به انگلیسی

Axon	اکسون
Particle Swarm Optimization Algorith	الگوریتم بهینه سازی گروه ذرات
Imperialism Competitive Algorith	الگوریتم رقابت استعماری
Genetic Algorith	الگوریتم ژنتیک
Imperial	امپراتوری
Imperialism	امپریالیسم
Static	ایستا
Adaline	آدلاین
Threshold	آستانه
Principal Component Analysis	آنالیز اجزای اصلی
top-down	بالا به پایین
Online	برخط
Offline	برون خط
Overfitting	بیش برازش
bottom-up	پایین به بالا
Perceptron	پرسپترون
Evaluation Function	تابع ارزیابی
Criterion Function	تابع معیار
Radon Transform	تبدیل رادون
Gabor Wavelet Transform	تبدیل ویولت گابور
Validation Procedure	تعیین اعتبار
Cross over	تقاطع

Assimilation	جذب
Random Search	جستجوی تصادفی
Heuristic Search	جستجوی مکاشفه‌ای
Multilayer	چند لایه
Base line	خط اصلی
Accuracy	دقت
Dendrite	دندریت
Epoch	دوره
Sinaps	سیناپس
Probabilistic Neural Network	شبکه عصبی احتمالی
Backpropagation Neural Network	شبکه عصبی پس انتشار
General Regression Neural Network	شبکه عصبی رگرسیون عمومی
Radial Basis Neural Network	شبکه عصبی شعاعی
Artificial Neural Network	شبکه عصبی مصنوعی
Flexible Grid Base	شبکه‌ی قابل انعطاف
Coefficient	ضریب
Generality	عمومیت
Search Space	فضای جستجو
Weight Space	فضای وزن‌ها
Chain Code	کد زنجیره‌ای
Hidden Marcov Model	مدل مخفی مارکوف
Colony	مستعمره
Information Measures	معیارهای مبتنی بر اطلاعات
Classifier Error Rate Measures	معیارهای مبتنی بر خطای کلاسه‌بند

Consistency Measures	معیارهای مبتنی بر سازگاری
Distance Measures	معیارهای مبتنی بر فاصله
Dependence Measures	معیارهای مبتنی بر وابستگی
Mean Square Error	میانگین مربعات خطا
Equal Error Rate	نرخ خطای برابر
Neuron	نرون
Horizontal Projection	نگاشت افقی
Vertical Projection	نگاشت عمودی
Correlation	همبستگی
Grid Base Feature	ویژگی شبکه‌ای
Supervised Learning	یادگیری نظارت شده
Unsupervised Learning	یادگیری نظارت نشده

Abstract

Nowadays the use of biological characteristics to authenticate individuals is increasing.

The most biological characteristics which used in this field are fingerprint, iris, retina, face, voice and signature. In the administrative System, especially in the field of economic and commercial sectors, such as banks, corporations, organizations are seeking ways to identify people. One of these tools which are used a lot cause of its simplicity and short cost is signature, But individual's signatures is at risk of forge by other people. Therefore an efficient method to recognize genuine signature from forgeries would be useful.

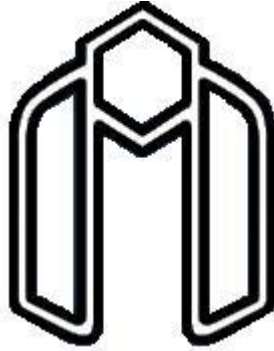
According to the dependency to the kind of receiving the data, signature verification can be divided into two groups, online and offline. In online methods, Signatures information and the information of the dynamic characteristics of signatures such as pressure and speed and ... can be achieved by the use of a tablet and an electric pen which is connected to a computer. While in offline approach, information where collected when signing operation is done, Then at the other time written signature is converted to digital image by a scanner. In this case, the extracted features are called static. Offline systems are most practical in comparison with online systems and are used in a simpler way in all over the world. However because of the lack of the dynamic information, they are supposed to be more complicated against online systems.

In this thesis our work is based on the offline systems. At first in order to preprocessing we cropped the signatures, and then by a special segmentation on signature image, new gride-based local features are extracted. In continue optimal subset of these features are selected by using Imperialist Competitive Algorithm. In classification level, an artificial Neural Network is used for each signer which we consider ۲۷ signatures for training phase. In order to evaluation phase, the ۲۷ remained signatures of each individual are injected to it's corresponding neural network which has been learned in the previous step. Decision about refusing or acceptting of the test signature is made. Finally the results are compared with similar projects. In this thesis we used GPDS۳۰۰GARY database to evaluate proposed algorithm. Equal Error Rate(EER) of the proposed algorithm for this database is ۶,۹۶

Practical results show that the proposed algorithm has an acceptable accuracy in comparison with the other presented methods.

Keywords:

Signature verification, Static characteristics, Feature selection, Imperialist competitive algorithm, Grid based features



Shahrood University of Technology

Faculty of Electrical and Robotic Engineering.

Offline Handwritten Signature Verification

Newsha Navaie

**Supervisor :
Dr. Ali Soleimani**

Date: September ۲۰۱۲