

بازشناسی برخط حروف فارسی بر پایه مدل مخفی مارکوف

مسعود فرکی و مازیار پالهنگ

آزمایشگاه هوش مصنوعی- دانشکده مهندسی برق و کامپیوتر- دانشگاه صنعتی اصفهان- اصفهان- ایران

palhang@cc.iut.ac.ir و m.faraki@ec.iut.ac.ir

چکیده: در این مقاله روشی برای بازشناسی برخط حروف فارسی بر پایه مدل مخفی مارکوف ارائه می‌شود. ابتدا با به دست آوردن تعداد بخش‌های حرف نوشته شده و بازشناسایی اجزای کوچک آن از تعداد حروف نامزد کاسته می‌شود. سپس بدنه و اجزای کوچک حرف ناشناخته پیش پردازش می‌شوند. مراحل پیش پردازش شامل یکسان‌سازی اندازه و دوباره نمونه‌برداری است. در این صورت بازشناسی نسبت به تغییر اندازه و جابجایی مقاوم و همچنین استخراج ویژگی با دقت بالاتری انجام می‌شود. ویژگی‌های استخراج شده شامل هر دو نوع ویژگی‌های محلی و ساختاری هستند. ویژگی‌های محلی، زوایای بین بردارهای برازش شده به نقاط مهم حرف ناشناخته با محور افقی و ویژگی‌های ساختاری، بنا به تعریف، شامل نقاط اوج، دره-راست و دره-چپ هستند. آموزش مدل نیز با استفاده از الگوریتم Baum-Welch، با پس‌پردازشی روی آن انجام شده است. استفاده از مراحل ذکر شده باعث شده است که بازشناسی صورت گرفته بسیار موفق و کاملاً مستقل از نویسنده باشد. گزارش‌های به دست آمده از نتایج آزمایش‌ها، بیان‌گر دقت بازشناسی ۹۷/۲۲٪ در بخش آموزش و ۹۴/۹٪ در بخش آزمایش سیستم در روش پیشنهادی است.

واژه‌های کلیدی: بازشناسی برخط دست نوشته، مدل مخفی مارکوف.

Online handwritten recognition based on hidden Markov mode

Masoud Faraki and Maziar Palhang

Electrical and Computer Engineering Department, Isfahan University of Technology, Isfahan, Iran.

Abstract: In this paper an approach based on hidden Markov model for recognizing online Farsi characters is presented. At first by obtaining the number of parts of a written character and recognizing its delayed strokes, the number of candidates is decreased and then the body and delayed strokes of unrecognized character are preprocessed. The stage of preprocessing is consisting of size normalization and resampling. Thus the recognition process will be robust to transition and scaling and the extracted features will be done more precisely. The extracted features are both local and structural features. The local features are consisting of the angles between the fitted vectors to some important points of unrecognized character and the structural features are consisting of cusp, left-hump and right-hump points. The training process of models is done by Baum-Welch algorithm with a post process on it. Using the mentioned stages has the advantage of doing the recognition process in an unconstrained and writer independent manner. The obtained results show the 97.22% precision in training and 94.9% precision in testing experiments.

Keywords: Online handwritten recognition, hidden Markov model.

۱- مقدمه

برای مدت‌های طولانی تعامل بین انسان و کامپیوتر توسط ابزارها و وسایلی مثل صفحه کلید، موشواره، صفحه نمایش و وسایلی از این قبیل صورت می‌گرفته است. اما در دهه‌های اخیر رشد و گسترش وسایلی مانند کامپیوترهای جیبی، صفحات رقمی‌کننده و انواع گوشی‌های همراه، نیاز به تعامل آسان‌تر و طبیعی‌تر را با آن‌ها فراهم آورده‌اند. از نمونه‌های این تعاملات طبیعی می‌توان به صحبت و دست نوشته اشاره کرد. بازشناسی برخط^۱ حروف دست‌نویس فارسی موضوع این مقاله است. سیستم‌های بازشناسی نوشتار با توجه به نحوه معرفی داده به آن‌ها به دو نوع برخط و برون‌خط^۲ تقسیم می‌شوند. در نوع برون‌خط از تصویر نوشتار به عنوان داده ورودی استفاده می‌شود ولی در بازشناسی برخط مختصات نقاط مسیر حرکت قلم و سایر اطلاعات پویای نوشتاری (مانند زمان، هنگام نمونه‌برداری و میزان فشار قلم) ورودی سیستم است [۱]. بازشناسی برخط مخصوص ابزارها و وسایلی معرفی شده است.

اختلاف بیش از اندازه بین دست خط‌های افراد مختلف، باعث بروز مشکلات فراوانی در بازشناسی دست نوشته شده است. حتی شیوه‌های نوشتاری یک فرد نیز در مواقع و حالت‌های مختلف، با یکدیگر تفاوت عمده دارند. به عنوان مثال یک فرد ممکن است با توجه به محتوای متن، وسیله مورد استفاده، موقعیت جغرافیایی قرار گرفته و یا حالت‌های مختلف درونی‌اش شیوه نوشتاری خاصی را استفاده کند. بنابراین نیاز به بازشناسی صحیح و با دقت بسیار بالا برای نوشتار امری ضروری است.

در زمینه بازشناسی برخط نوشتار انگلیسی تحقیقات گسترده‌ای انجام شده است. در [۲] از یک شبکه عصبی تاخیر زمانی^۳ برای بازشناسی استفاده شده است. اما این سیستم بازشناسی مشابه سیستم‌های برون‌خط است، زیرا جهت بازشناسی، حرف نوشته شده را به صورت لایه لایه به شبکه می‌دهند و چندان از اطلاعات و جزئیات برخط استفاده نشده است.

در [۳] یک عبارت خطای جدید برای الگوریتم آموزش انتشار به عقب^۴ تعریف و از آن برای بازشناسی برخط اعداد انگلیسی در یک شبکه چند لایه استفاده شده است. تعداد گام‌ها به منظور رسیدن به حداقل خطای طبقه‌بندی به میزان قابل توجهی کاهش یافته است.

الگوریتم‌های پیچش زمانی پویا^۵ یک از ابزارهای بازشناسی برخط دست نوشته هستند [۴-۷]. پیچش زمانی پویا روشی است که بهترین انطباق را بین دو سری زمانی، در صورتی که یکی از آن‌ها به صورت غیرخطی گسترش یا کاهش یافته باشد، پیدا می‌کند [۸]. در این روش در زمان طراحی، برای کلیه طبقه‌های موجود از نمونه‌های حاضر در پایگاه داده آموزشی یک نمونه الگو ساخته می‌شود. نمونه الگوی یک طبقه به عنوان نماینده آن طبقه معرفی می‌شود. به منظور طبقه‌بندی

یک نمونه، ابتدا با دنباله نقاط ورودی سیستم یک سری زمانی ساخته می‌شود و سپس بهترین انطباق را بین این سری زمانی با نمونه الگوهای موجود، با استفاده از پیچش زمانی پویا یافته و طبقه مناسب انتخاب می‌شود. مشکل این روش در محاسبات بسیار بالای هنگام طبقه‌بندی، مراحل طراحی پیچیده و ساخت نمونه الگوها است. با توجه به این که در این روش مرحله استخراج ویژگی نیز صورت نمی‌گیرد، در صورتی که از پیش پردازش مناسب استفاده نشود، بازشناسی نسبت به تغییر اندازه مقاوم نخواهد بود.

مدل مخفی مارکوف^۶ نیز در مسائلی که ذاتاً زمانی هستند، مانند بازشناسی گفتار و نوشتار کاربرد فراوانی دارد [۹]. در [۱۰-۱۲] از مدل مخفی مارکوف گسسته برای بازشناسی برخط حروف و کلمات دست نوشته انگلیسی استفاده شده است. با استفاده از مرحله پیش پردازش و همچنین استخراج ویژگی‌های مناسب، نتایج بازشناسی بر روی پایگاه داده UNIPEN، با دقت بسیار مطلوبی گزارش شده‌اند. بازشناسی انجام شده در محدوده کلمات موجود در فرهنگ لغت ساخته شده است. به منظور محدود کردن فضای فرضیه به سطح قابل قبول، از محدودیت‌های دستور زبانی نیز استفاده شده است.

در زمینه بازشناسی برخط حروف یا کلمات عربی و فارسی تحقیقات چندانی صورت نگرفته است. در [۱۳] حروف مجزای فارسی بر اساس نقاط و علائم بالا یا پایین بدنه اصلی به ۱۲ گروه تقسیم شده‌اند. نخست نقاط و علائم بالا یا پایین هر حرف با یک شبکه عصبی تشخیص داده می‌شود و سپس با توجه به آن، گروهی که حرف ناشناخته در آن قرار می‌گیرد، تعیین می‌گردد. در صورتی که در گروه مربوطه فقط یک حرف وجود داشته باشد، آن حرف به حرف ناشناخته نسبت داده می‌شود و گرنه بدنه حرف ناشناخته با بدنه‌های حروفی از پایگاه داده که با حرف ناشناخته هم‌گروه هستند مقایسه می‌شود و با طبقه‌بند کمترین فاصله، بازشناسی انجام می‌گیرد. نرخ بازشناسی از ۶۳٪ تا ۹۳٪ گزارش شده است.

[۱۴ و ۱۵] روشی را برای بازشناسی برخط زیر کلمات فارسی (بخش‌های یک حرف که حروف آن به هم می‌چسبند) ارائه می‌کنند. ابتدا نقاط و علائم بالا یا پایین بدنه اصلی زیر کلمه ناشناخته تشخیص داده می‌شوند و براساس موقعیت مکانی مرتب می‌شوند. سپس با توجه به آن یک زیرمجموعه از مجموعه زیر کلمات فارسی انتخاب می‌شود. در نهایت، زیرکلمه‌ای که کمترین فاصله را داشته باشد، انتخاب می‌شود. مجموعه زیر کلمات پوشش داده شده برابر با ۷۳۱۷ زیر کلمه هستند که به ۸۹۴ گروه تقسیم شده‌اند.

[۱۶] روشی را برای بازشناسی برخط کلمات فارسی ارائه می‌کند. ابتدا زیرکلمات موجود در کلمه نوشته شده استخراج می‌شوند. سپس علائم هر زیرکلمه و مکان نسبی آن‌ها تشخیص داده می‌شوند و با توجه به

(مستقل از نویسنده^۸) حاکی از دقت بسیار بالای بازشناسی است. ساختار مقاله به این ترتیب است که بخش دوم مقاله چارچوب کلی سیستم بازشناسی را معرفی می‌کند. بخش‌های سوم و چهارم، دو نوع غربال کردن را بیان می‌کنند. بخش پنجم، مربوط به پیش‌پردازش و دوباره نمونه‌برداری است. بخش ششم، ویژگی‌های محلی و ساختاری استخراج شده را معرفی می‌کند. بخش هفتم، تنظیمات در نظر گرفته شده برای آموزش و بازشناسی و بخش هشتم، نتایج به دست آمده از آزمایش‌ها و دقت سیستم بازشناسی را گزارش می‌دهند. بخش آخر مقاله نیز به جمع‌بندی و کارهای آینده در این زمینه اختصاص داده شده است.

۲- چارچوب بازشناسی

از آنجایی که هدف بازشناسی برخط حروف فارسی است، در این پروژه فرض بر این است که بدنه حروف و هر کدام از اجزای کوچک حرف، در یک حرکت قلم نوشته شوند. منظور از اجزای کوچک علائمی مثل نقطه، دونقطه و سه‌نقطه هستند که بعد از کامل نوشتن بدنه حروف گذاشته می‌شوند. برای دونقطه دو شکل دونقطه پیوسته و دو عدد نقطه نزدیک به هم و برای سه‌نقطه نیز دو شکل سه‌نقطه پیوسته و سه عدد نقطه نزدیک به هم در نظر گرفته شده است.

فرایند بازشناسی پس از گذشت یک ثانیه بعد از نوشتن حرف ناشناخته، آغاز می‌شود. ابتدا با استفاده از مرحله غربال کردن حروف نامزد توسط بخش‌های حرف، مجموعه‌ای از حروف نامزد انتخاب می‌شوند. سپس در صورتی که تعداد بخش‌های کلمه بیش از یک باشد، مرحله غربال کردن حروف نامزد توسط اجزای کوچک به کار می‌افتد و با بازشناسی اجزاء کوچک حرف، تعداد حروف نامزد را کاهش و این مجموعه حروف نامزد را به مرحله بازشناسی انتقال می‌دهد. از طرف دیگر بدنه حرف ناشناخته، پیش‌پردازش می‌شود. همان طور که در بخش‌های آینده ذکر می‌شود، این مرحله پیش‌پردازش به منظور بارز کردن ویژگی‌های حرف و مقاوم کردن بازشناسی نسبت به انتقال و تغییر اندازه نوشتن حرف، انجام می‌شود. بعد از پیش‌پردازش مرحله استخراج ویژگی و ساخت بردار ویژگی آغاز می‌شود. ویژگی‌های استخراج شده شامل هر دو نوع ویژگی محلی و ساختاری است. پس از استخراج این دو نوع ویژگی، بردار ویژگی به نحوی که مناسب برای بازشناسی باشد، ساخته می‌شود. در نهایت هم بردار ویژگی استخراج شده به مرحله بازشناسی انتقال می‌یابد و بازشناسی انجام می‌شود. ذکر این نکته اهمیت دارد که مراحل پیش‌پردازش، استخراج ویژگی و بازشناسی جزء کوچک نیز مشابه مراحل بدنه حرف انجام می‌شوند. شکل (۱) چارچوب بازشناسی را به همراه بخش‌های مختلف آن نشان می‌دهد.

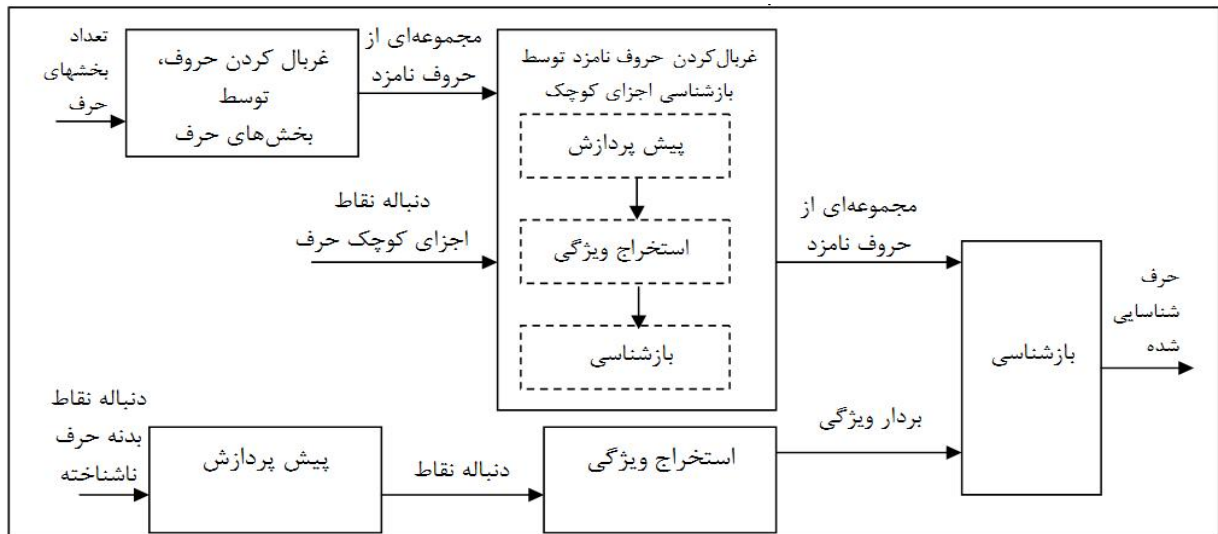
آن گروهی از کلمات، که از نظر تعداد زیرکلمات و علامت‌های موجود در هر زیرکلمه با کلمه ورودی یکسان هستند، از واژگانی با ۳۰۰۰۰ کلمه، انتخاب می‌شوند. هر کدام از زیرکلمات موجود در کلمه ورودی، با توجه به گروه انتخاب شده، با تعدادی از زیرکلمات مقایسه می‌شود و براساس فاصله‌ای که با آن زیرکلمات دارد امتیازی به هر کدام از آنان داده می‌شود. برای هر یک از کلمات گروه انتخابی امتیازی بر اساس امتیازات زیرکلمات آن محاسبه می‌شود و کلمه‌ای که بیشترین امتیاز را آورده باشد به کلمه ورودی نسبت داده می‌شود. روش پیشنهادی روی یک متن انتخابی، که توسط یک نویسنده نوشته شده، آزمایش شده است. میزان بازشناسی صحیح برابر با ۹۴،۹٪ برای متن نمونه به دست آمده است.

در [۱۷] از مدل مخفی مارکوف برای بازشناسی برخط ارقام فارسی استفاده شده است. میزان موفقیت بازشناسی برابر با ۱۴ . ۹۳٪ گزارش شده است. در [۱۸] نیز از مدل مخفی مارکوف گسسته برای بازشناسی برخط کلمات عربی استفاده شده است. مراحل پیش‌پردازش و استخراج ویژگی به ترتیب اعمال می‌شوند. بازشناسی انجام شده به مجموعه لغات حاضر در فرهنگ لغت ساخته شده، محدود است. دقت بازشناسی در حدود ۹۵٪ گزارش شده است.

در [۱۹] به منظور بازشناسی برخط دست‌نوشته از منطق فازی استفاده شده است. در روش پیشنهادی نمایش پارامترهای نوشتاری توسط متغیرهای زبانی فازی انجام شده است. در این کار این نتیجه حاصل شده است که مدل‌سازی نوشتار بر اساس متغیرهای زبانی فازی، امکان مقاوم بودن سیستم را نسبت به تغییرات دست‌خط فراهم می‌کند. روش پیشنهادی روی مجموعه‌ای از کاراکترهای فارسی آزموده شده است.

در کار قبلی [۲۰]، سیستمی را معرفی کردیم که بر اساس آن، کاربر با استفاده از یک لیزر دستی قادر بود مطالبی را روی دیوار یا پرده بنویسد. قابلیت بازشناسی حروف فارسی را به عنوان یک کاربرد مناسب به سیستم مذکور افزودیم. بازشناسی انجام شده با استفاده از درخت تصمیم‌گیری و آموزش آن با استفاده از الگوریتم C4.5 و تصدیق متقاطع ۱۰ تایی انجام می‌گرفت. در این سیستم تنها آزمایش وابسته به نویسنده^۷ و آن هم برای نمونه دست‌خط‌های فرد آموزش دهنده انجام پذیرفت.

در این مقاله روشی برای بازشناسی برخط حروف دست‌نوشته فارسی بر پایه مدل مخفی مارکوف گسسته پیشنهاد می‌شود. استفاده از مراحل غربال کردن، پیش‌پردازش و همچنین استخراج ویژگی‌های ابتکاری ساختاری باعث شده است تا بازشناسی کاملاً موفق بوده و همچنین نسبت به دست‌خط‌های مختلف، مقاوم باشد. نتایج به دست آمده از آزمایش‌های انجام شده در هر دو بخش آموزش و آزمایش



شکل (۱): چارچوب سیستم بازشناسی

اعمال غریبال کردن توسط بخش‌های حرف در سیستم‌های بیرون خط که از تصویر نوشتار به عنوان ورودی استفاده می‌کنند، از پیچیدگی محاسباتی بالایی برخوردار است. زیرا در این گونه سیستم‌ها نیاز به حداقل یک بار پوشش کامل تصویر و برچسب‌زنی اجزای متصل است. همچنین در صورتی که اجزای کوچک با یکدیگر و یا بدنه حروف همپوشانی داشته باشند، این نوع غریبال کردن نتایج مناسبی نخواهد داشت. از نمونه‌های همپوشانی بین جزء کوچک و بدنه می‌توان به دسته روی حرف "ط" اشاره کرد.

۴ - غریبال کردن حروف نامزد توسط اجزای کوچک

ورودی این بخش دنباله نقاط اجزای کوچک حرف و یکی از چهار مجموعه حروف نامزد که در بخش قبل تولید شده‌اند و خروجی آن مجموعه‌ای از حروف نامزد که اجزای کوچک یکسان دارند، است. بنابراین در صورتی که تعداد بخش‌های کلمه، یک باشد، مجموعه حروف نامزد ورودی و خروجی برابرند و در غیر این صورت ابتدا جزء کوچک پیش‌پردازش شده سپس ویژگی‌های آن استخراج و در نهایت، بازشناسی انجام می‌شود. سه مرحله پیش‌پردازش، استخراج ویژگی و بازشناسی در این مرحله، مشابه این مراحل برای بدنه حرف هستند که در ادامه توضیح داده خواهند شد. با بازشناسی صحیح جزء کوچک و غریبال کردن حروف با جزء کوچکشان به میزان قابل توجهی از تعداد نامزدها در مجموعه حروفی که دو بخشی هستند کاسته می‌شود. جدول (۲) تقسیم‌بندی حروف را با توجه به جزء کوچکشان نشان می‌دهد (برای دونقطه و سه‌نقطه دو حالت در نظر گرفته شده است).

۳- غریبال کردن حروف نامزد توسط بخش‌های حرف

این مرحله به منظور کاهش تعداد حروف نامزد برای بازشناسی انجام می‌شود. تقریباً همه فارسی زبانان اجزای کوچک یک حرف را بعد از کامل نوشتن بدنه حرف می‌نویسند. بنابراین در کاربردهای برخط به سادگی می‌توان تعداد بخش‌های حرف نوشته شده، که همان تعداد برخورد‌های قلم با صفحه است را با استفاده از یک شمارنده و رویدادهای قلم به دست آورد. با استفاده از این ویژگی برخط و غریبال کردن، به گونه قابل توجهی از پیچیدگی و جستجو در مرحله بازشناسی کاسته می‌شود. همان طور که قبلاً هم اشاره شد برای اجزای کوچک دونقطه دو حالت (دو نقطه پیوسته و دو عدد نقطه) و برای سه‌نقطه دو حالت (سه نقطه پیوسته و سه عدد نقطه) در نظر گرفته شده است. نقطه نیز باید کمتر از هشت پیکسل را به خود اختصاص دهد. جدول (۱) تعداد بخش‌های ممکن حروف فارسی را با توجه به نحوه نوشتن آنها نشان می‌دهد.

جدول (۱): تعداد بخش‌های ممکن حروف فارسی و حروف آنها

| تعداد بخش‌های حروف | ۱ | ۲ | ۳ | ۴ | |
|--------------------|---|---|---------------|------------------|--|
| حروف نامزد | ا، ح، د، ر، س، ص ع، ل، م و، ه، ی | آ، ب، پ، ت، ث، ج، چ، خ، ذ، ز، ژ، ش، ض، غ، ط، ک، ف، ق، ن | ت، ظ، گ، ق | پ، ث، چ، ژ، ش | |

جدول (۲): تقسیم‌بندی حروف با توجه به اجزای کوچک آنها

| ۴ | ۳ | | | ۲ | | | | | ۱ | تعداد بخش‌های حروف مجموعه |
|---------------------------|------------------------|-------------------|----------------|---------------------------|------|------|---------------------|---------------------------------------|---|------------------------------------|
| پ، ث، ج، ژ، ش | گ | ظ | ت، ق | پ، ث، ج، ژ، ش | ط | ک | ت، ق | ب، ج، خ، ذ، ز، ض، غ، ف، ن | ا، ح، د، ر، س، ص، ع، ل، م، و، ه، ی | حروف مجموعه |
| سه عدد نقطه | دو عدد سرکش نقطه | دسته و نقطه | دو عدد نقطه | سه نقطه (پیوسته) | دسته | سرکش | دو نقطه (پیوسته) | مد نقطه | - | نام جزء (های) کوچک |

۵ - پیش‌پردازش

PXA27X ARM920T و 56.62MB، فراهم شده‌اند. در هنگام نمونه‌برداری نیز از ذخیره کردن نقاط یکسان متوالی خودداری می‌شود.

۶ - یکسان کردن اندازه

دنباله نقاط نمونه‌برداری شده $((x_1, y_1), \dots, (x_N, y_N))$ ، توسط معادلات پیشنهادی (۱) و (۲) نگاشت و دنباله نقاط $((x'_1, y'_1), \dots, (x'_N, y'_N))$ حاصل می‌شوند.

$$x'_i = \left(\frac{x_i - x_{\min}}{\sigma_x} \right) \times scale_factor \quad (1)$$

$$y'_i = \left(\frac{y_i - y_{\min}}{\sigma_y} \right) \times scale_factor \quad (2)$$

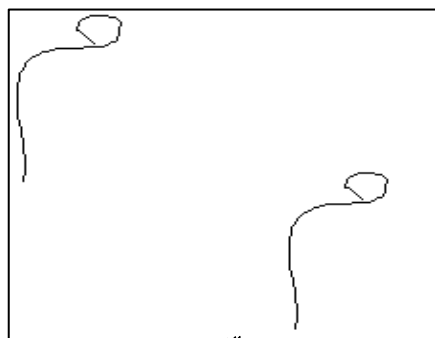
$$\mu_x = \frac{1}{N} \sum_{i=1}^N x_i, \quad \sigma_x = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \mu_x)^2}, \quad \text{که}$$

$$x_{\min} = \min(x_i) \quad \text{و} \quad y_{\min} = \min(y_i) \quad \text{هستند.}$$

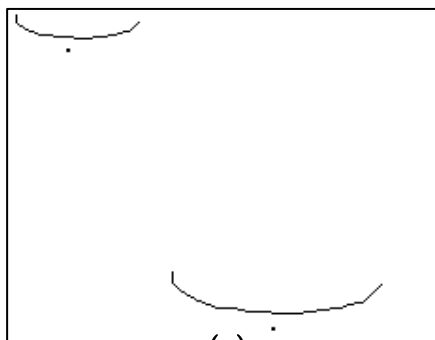
ذکر دو نکته در اینجا اهمیت دارد.

۱- $scale_factor$ یک ضریب اندازه است که با سعی و خطا برای بدنه حروف برابر با ۳۰ و برای اجزای کوچک برابر با ۲۰ در نظر گرفته شده است و اندازه ابعاد حرف پیش‌پردازش شده را نشان می‌دهد. در صورتی که از ضریب اندازه استفاده نشود اکثر نقاط حاصل شده در محدوده کوچکی قرار می‌گیرند و نتایج مطلوب حاصل نمی‌شود.

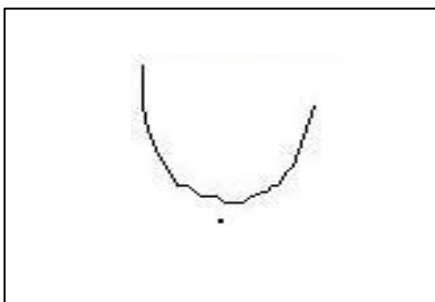
این بخش به منظور مقاوم‌سازی بازشناسی نسبت به تغییرات اندازه، جابجایی و همچنین بارزسازی ویژگی‌های آن‌ها انجام می‌شود. اگر چه با استخراج ویژگی‌های مناسب می‌توان تا حد زیادی به اهداف ذکر شده دست یافت اما انجام یک مرحله پیش‌پردازش روی دنباله نقاط، نیاز به استخراج ویژگی‌های پیچیده را از بین برده و درصد موفقیت بازشناسی را به گونه قابل توجهی بالا می‌برد. منظور از دوباره نمونه‌برداری حذف نقاط زائد در دنباله نقاط یک حرف و توزیع کردن نقاط، به نحو مناسب است. با توجه به استخراج ویژگی‌های ساختاری در این مقاله توزیع کردن نقاط به گونه مناسب در استخراج این گونه ویژگی‌ها کمک فراوانی می‌کند. در مقالات مختلف پیش‌پردازش‌های متفاوتی توصیه شده است. از جمله این پیش‌پردازش‌ها می‌توان به حذف هر n نقطه و حذف سایر نقاط، افزودن n نقطه بین نقاط متوالی، توزیع نقاط به گونه‌ای که فاصله یکسان از یکدیگر پیدا کنند، نگاشت دنباله نقاط به گونه‌ای که در یک ناحیه با طول و عرض از پیش تعیین شده‌ای قرار گیرند و انتقال مرکز جرم نقاط، اشاره کرد. تمامی این روش‌ها اگر به گونه‌ای خطی و به طور یکسان برای همه حروف انجام شوند، باعث از بین رفتن نقاط بارز و قابل توجه می‌شوند. پیش‌پردازش‌های انجام شده در این مقاله شامل یکسان کردن اندازه و دوباره نمونه‌برداری است. لازم به ذکر است که داده‌های مورد استفاده با استفاده از یک کامپیوتر جیبی با اندازه صفحه 240×320 پیکسل، سیستم عامل Windows Mobile 5[®] و همچنین CPU و RAM،



(الف)



(ب)



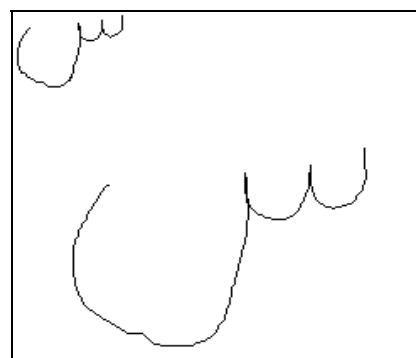
(ج)

شکل (۳): اعمال یکسان کردن اندازه روی دو حرف مختلف، (الف): نتیجه یکسان کردن اندازه روی حرفی با کشیدگی عمودی زیاد، (ب): نتیجه یکسان کردن اندازه روی حرفی با کشیدگی افقی زیاد (ج): نتیجه استفاده از انحراف معیار عمودی در معادله (۲) و یکسان کردن اندازه شکل اصلی (ب)

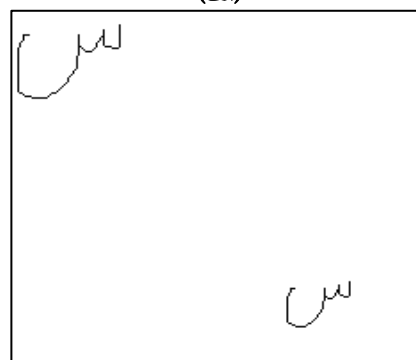
در شکل (الف) نتیجه یکسان شده حرف "م" نشان داده شده است که نسبت به حرف "ب" دارای ارتفاع بیشتری می‌باشد. بنابراین همان طور که در این دو شکل مشاهده می‌شود شکل طبیعی این حروف (که اولی در راستای عمودی و دومی در راستای افقی کشیدگی دارند) حفظ شده و فقط عمل یکسان کردن اندازه روی آن‌ها صورت گرفته است. در صورتی که در راستای عمودی از انحراف معیار عمودی استفاده شود شکل حروفی مانند حرف "ب" کشیدگی عمودی زیادی پیدا خواهند کرد (به علت کوچک شدن مخرج معادله (۲)) و نتیجه یکسان شده اندازه با شکل اصلی حرف، تفاوت زیادی خواهد داشت. این مطلب در شکل (۳) (ج) قابل مشاهده می‌باشد. همچنین نظر به این که اکثر حروف فارسی کشیدگی افقی مناسبی دارند استفاده از انحراف معیار افقی در عمل یکسان کردن اندازه، منطقی می‌باشد. ذکر این نکته

۲- تعداد نقاط در دنباله خروجی برابر با تعداد نقاط دنباله ورودی است، اما نقاط به گونه‌ای انتقال می‌یابند که شکل حرف نوشته شده تقریباً بین دست‌خط‌های مختلف افراد یکسان باشد. تغییر اندازه در شکل حروف نیز تاثیری در نتیجه پیش‌پردازش شده ندارد. از آنجائی که اکثر حروف گسترده‌گیشان در جهت افقی است، از σ_x در مخرج هر دو کسر (هم برای x ها و هم برای y ها) استفاده شده تا از انحرافات نقاط در راستای x تا حد زیادی کاسته شود.

شکل‌های (۲) (الف) و (ب) حاصل خروجی این بخش را روی حرف "س" با دو اندازه مختلف و طرز نوشتاری متفاوت، نشان می‌دهد. نتیجه حاصل از مرحله یکسان کردن اندازه در هر مورد در گوشه بالا و چپ تصویر نشان داده شده است. همان طور که در این شکل مشاهده می‌شود اگر چه دو حرف نوشته شده، اندازه و مکان مختلف دارند اما نتیجه نگاشت شده‌شان تقریباً با یکدیگر انطباق دارند. به بیان دیگر اگر معادلات (۱) و (۲) را بر روی دنباله نقاط شکل‌های (۲) (الف) و (ب) اعمال کنیم مشاهده می‌کنیم که مرحله یکسان کردن اندازه، شکل یک حرف که با دو اندازه مختلف نوشته شده است را به شکل‌هایی تبدیل کرده است که از لحاظ اندازه (تقریباً) یکسان می‌باشند.



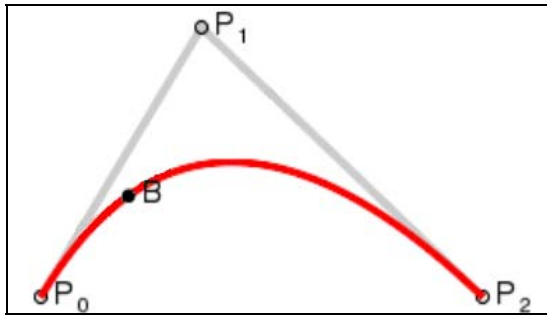
(الف)



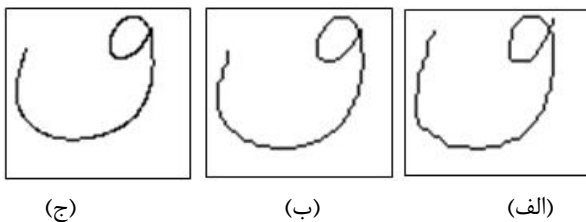
(ب)

شکل (۲): اعمال یکسان کردن اندازه روی یک حرف با اندازه‌های متفاوت، (الف): نتیجه یکسان کردن اندازه روی حرفی با اندازه بزرگ، (ب): نتیجه یکسان کردن اندازه روی حرفی با اندازه کوچک

شکل‌های (۳) (الف) و (ب) نشان می‌دهند که خروجی مرحله پیش‌پردازش برای شکل حروف متفاوت، یکسان نیست.



شکل (۴): یک منحنی Bezier درجه ۲ ساخته شده با گام افزایشی ۰.۲۵ در نهایت نیز در صورتی که اختلاف سطر و ستون دو نقطه متوالی در دنباله نقاط حاصل شده، بیش از یک باشد، نقاطی روی خط مستقیم بین دو نقطه به دنباله نقاط افزوده می‌شوند تا هیچ ناپیوستگی‌ای در دنباله نقاط وجود نداشته باشد. شکل (۵) مراحل مختلف دوباره نمونه‌برداری را روی بدنه یک حرف فارسی نشان می‌دهد. لازم به ذکر است که در شکل‌های (الف) و (ب) نقاط نمونه‌برداری شده به صورت خطی به یکدیگر متصل شده‌اند اما در شکل (ج) این نقاط با استفاده از منحنی مذکور به یکدیگر متصل شده‌اند.



شکل (۵): اعمال مراحل مختلف دوباره نمونه‌برداری روی بدنه حرف "ق" دست‌نویس، (الف): دنباله نقاط ورودی یافته (ب): دنباله نقاط متصل پس از اعمال صافی گوسی (ج): اتصال دنباله نقاط صافی شده با استفاده از منحنی Bezier

۸ - استخراج ویژگی

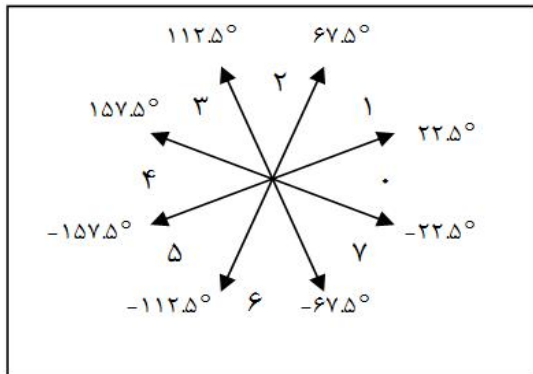
دنباله نقاط حاصل از مرحله پیش‌پردازش به بخش استخراج ویژگی انتقال داده می‌شوند. با توجه به شکل ساختاری حروف فارسی، استخراج بردار ویژگی با طول ثابت به ازای هر حرف منجر به از دست دادن بسیاری از اطلاعات مفید برای حروف با شکل پیچیده و به دست آوردن اطلاعات مازاد برای شکل حروف ساده‌تر می‌شود. اما با به‌کارگیری ابزار بازشناسی مناسب می‌توان به بردار ویژگی با طول متغیر تکیه کرد. در این مقاله از مدل مخفی مارکوف گسسته به عنوان ابزار بازشناسی استفاده شده است و طول بردار ویژگی نیز ثابت نیست. در این بخش دو نوع ویژگی محلی و ساختاری استخراج می‌شوند. ویژگی‌های محلی قابل اطمینان‌تر اما کمتر حاوی اطلاعات مفید می‌باشند و در مقابل، ویژگی‌های ساختاری قدرتمندتر ولی کمتر قابل اطمینان هستند. به بیان دیگر، ویژگی‌های محلی با دقت بالایی قابل

ضروری است که این عمل، در مورد حرف "ا" که انحراف معیار افقی آن از حد آستانه‌ای کوچکتر می‌باشد انجام نمی‌شود.

۷ - دوباره نمونه‌برداری

همان‌طور که می‌دانیم نقاطی که از حرکت قلم روی صفحه حساس کامپیوتر جیبی (یا هر صفحه رقمی کننده دیگر) حاصل می‌شوند یک دنباله نقاط پیوسته نمی‌باشند. اما به این علت که قصد داریم تا کاربر، احساسی مانند آنچه که در هنگام نوشتن روی کاغذ دارد، داشته باشد، در برنامه پیاده‌سازی شده، هنگامی که کاربر قلم را روی صفحه حرکت می‌دهد نقاط حاصل از نمونه‌برداری به یکدیگر با خط مستقیم متصل می‌شوند. این عمل با سرعت بالایی قابل انجام است. اما همان‌طور که در شکل (۵) (الف) مشاهده می‌شود، شکل حرف به دست آمده دارای اعوجاج‌های زیادی است. در این بخش به ارائه راه‌حلهایی برای حل این مسئله می‌پردازیم. ورودی این مرحله، دنباله نقاط حاصل از اعمال یکسان کردن اندازه است. ابتدا به منظور از بین بردن انحرافات که ممکن است توسط دستگاه نمونه‌برداری یا به علت لرزش دست نویسنده صورت گرفته شده باشند، یک صافی گوسی پائین‌گذر روی دنباله نقاط اعمال می‌شود. به علت این که این انحرافات معمولاً با فرکانس بالا ایجاد می‌شوند، این صافی، پائین‌گذر انتخاب شده است. ضرایب این صافی نیز برابر با اعداد چهارمین سطر مثلث خیام-پاسکال در نظر گرفته شده‌اند. سپس مجموعه نقاط جدید به منظور اتصال نقاط متوالی در بین نقاط صافی شده اضافه می‌شوند. برای این منظور ابتدا بین هر دو نقطه متوالی یک نقطه مجازی با مختصات برابر با میانگین دو نقطه مذکور در نظر گرفته و نقاط جدید با استفاده از منحنی Bezier [۲۳] درجه ۲ با گام افزایشی برابر با ۰.۱، تولید می‌شوند (بدون در نظر گرفتن نقاط تکراری). منحنی Bezier توسط Pierre Bezier در سال ۱۹۶۲ معرفی شد و یک منحنی پارامتری مهم در گرافیک کامپیوتری و زمینه‌های مرتبط با آن است. از این منحنی برای هموار مدل کردن انواع منحنی‌هایی که به صورت نامعین تغییر اندازه می‌دهند، استفاده می‌شود. یک منحنی Bezier درجه ۲ (معادله (۳)) مسیری سهمی‌گون است که توسط تابع $B(t)$ با داشتن سه نقطه P_0 ، P_1 و P_2 با تعیین گام افزایشی t (که متغیر از ۰ تا ۱ است)، ساخته می‌شود (شکل (۴)).

عنوان یک ویژگی نهایی (که تضمین می‌کند طول بردار ویژگی کمتر از دو نشود) زاویه بردار بین نقطه مرکز جرم و نقطه نهایی، با محور افقی نیز محاسبه، رقمی شده و در بردار ویژگی قرار می‌گیرد. بنابراین طول بردار ویژگی در این مرحله برابر با n (خروجی الگوریتم Douglas and Peucker) می‌شود. حد آستانه در نظر گرفته شده برای الگوریتم Douglas and Peucker با سعی و خطا برابر با ۲.۵ در نظر گرفته شده است.



شکل (۷): هشت ناحیه در نظر گرفته شده برای رقمی کردن بردارها

ویژگی‌های ساختاری شامل نقاط اوج، دره-راست و دره-چپ هستند. اگر $v_x[i]$ ، $v_y[i]$ و $\text{deg}[i]$ به ترتیب نشان دهنده سرعت افقی در نقطه i ، سرعت عمودی در نقطه i و زاویه بردار رسم شده بین نقطه $i-1$ و i ($-180 \leq \text{deg}[i] \leq 180$) و محورهای مختصات به صورتی که در شکل (۸) نشان داده شده است، باشند، تعریف هر یک از نقاط معرفی شده به صورت زیر است.

نقطه i را نقطه اوج می‌نامیم اگر دارای سه شرط زیر باشد:

$$1- \quad v_y[i-1] < 0, v_y[i] > 0 \text{ و } v_y[i+1] > 0 \text{ باشند. یعنی حرکت}$$

قبل از نقطه i در خلاف جهت محور عمودی و بعد از آن برعکس باشد.

$$2- \quad \text{deg}[i-1] < 0, \text{deg}[i] < 0 \text{ و } \text{deg}[i+1] > 0 \text{ باشند. یعنی تغییر}$$

ناگهانی زاویه را در نقطه i شاهد باشیم.

۳- بین دو اوج متوالی حداقل برای چند نقطه حرکت قلم، هم جهت

با محور عمودی شده باشد.

نقطه i را نقطه دره-راست می‌نامیم اگر دارای چهار شرط زیر باشد:

$$1- \quad v_y[i] < 0 \text{ و } v_y[i+1] < 0 \text{ باشند. یعنی حرکت بعد از نقطه } i \text{ در}$$

خلاف جهت محور عمودی باشد.

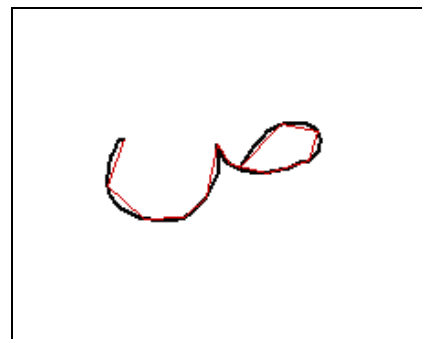
$$2- \quad v_x[i-1] > 0, v_x[i] > 0 \text{ و } v_x[i+1] > 0 \text{ باشند. یعنی حرکت}$$

در نقطه i و قبل و بعد آن نیز در جهت محور افقی باشد.

$$3- \quad \text{deg}[i-1] > 0, \text{deg}[i] > 0, \text{deg}[i+1] < 0 \text{ باشند. یعنی تغییر زاویه}$$

را در نقطه i شاهد باشیم.

محاسبه می‌باشند اما به منظور طبقه‌بندی، کمتر تاثیرگذار هستند اما ویژگی‌های ساختاری تاثیر بیشتری داشته ولی به دست آوردن آن‌ها با خطای بیشتری همراه است. ویژگی‌های محلی با به‌کارگیری الگوریتم Douglas and Peucker [۱۸،۲۱] به دست می‌آیند. ورودی این الگوریتم دنباله‌ای از نقاط و یک عدد حقیقی به عنوان حد آستانه تحمل است. ابتدا پاره‌خطی بین نقطه اول و نقطه آخر در دنباله نقاط ورودی رسم می‌شود. سپس در دنباله نقاط، نقطه‌ای که بیشترین فاصله اقلیدسی را تا پاره‌خط رسم شده دارا باشد، یافت می‌شود. اگر این نقطه را P_{max} بنامیم، در صورتی که فاصله P_{max} تا پاره‌خط، کمتر از مقدار آستانه باشد الگوریتم متوقف می‌شود و در غیر این صورت پاره‌خط اولیه به دو پاره‌خط، یکی بین نقطه اول (در دنباله نقاط) و P_{max} و دیگری بین P_{max} و نقطه آخر (در دنباله نقاط) و دنباله نقاط نیز به همین ترتیب به دو مجموعه تجزیه می‌شوند. سپس الگوریتم روی دو مجموعه نقطه به دست آمده به صورت بازگشتی تکرار می‌شود. بنابراین هنگامی که الگوریتم خاتمه یافت، فاصله تمامی نقاط از پاره‌خط مربوطشان کمتر از مقدار آستانه است. دنباله نقاط ابتدا و انتهای هر کدام از پاره‌خطها خروجی الگوریتم است. شکل (۶) یک حرف فارسی را به همراه پاره‌خطهای متناظر حاصل از اعمال الگوریتم Douglas and Peucker، با آستانه تحمل ۵ نشان می‌دهد.



شکل (۶): پاره‌خطهای رسم شده به دنباله نقاط حاصل از اعمال الگوریتم

Douglas and Peucker

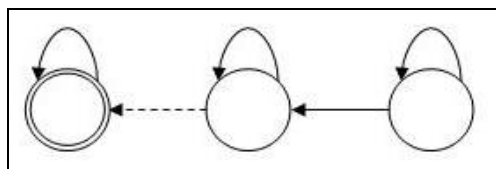
ویژگی‌های محلی این گونه محاسبه می‌شوند که ابتدا الگوریتم Douglas and Peucker به دنباله نقاط ورودی این بخش اعمال شده و مجموعه نقاط حاصل از اعمال الگوریتم به دست می‌آیند. سپس بین هر دو نقطه متوالی به دست آمده یک بردار برآزش و زاویه بین این بردارها و محور افقی محاسبه می‌شوند. بنابراین اگر خروجی الگوریتم n نقطه باشد، $n-1$ زاویه به دست می‌آید. از آنجائی که از مدل مخفی مارکوف گسسته برای بازشناسی استفاده شده است، ورودی به بخش بازشناسی باید دنباله‌ای از مشاهدات گسسته باشد. به همین منظور زوایا در هشت ناحیه متقارن (شکل (۷)) رقمی می‌شوند و در بردار ویژگی اولیه به جای هر زاویه عدد مربوطه (از ۰ تا ۷) قرار می‌گیرد. به

صحیح از صفر تا ده است و این بردار به بخش بازشناسی فرستاده می‌شود.

۹- تنظیمات مدل بازشناسی

۹-۱ مدل مخفی مارکوف

یک مدل مخفی مارکوف گسسته با N حالت (پنهان) و M نماد قابل مشاهده، توسط احتمالات انتقالی حالت‌های پنهان $A = [a_{ij}]_{N \times N}$ ، احتمالات مشاهدات $B = [b_{jk}]_{N \times M}$ و احتمالات اولیه حالت‌های پنهان $\pi = [\pi_1 \pi_2 \dots \pi_N]$ توصیف می‌شود. از مدل مخفی مارکوف گسسته راست به چپ ($i > 1$ برای $\pi_i = 0$; $i > 1$ یا $i > 1 + 1$ برای $a_{ij} = 0$) به عنوان ابزار طبقه‌بندی استفاده شده است (شکل ۹). جهت اطلاع یافتن از جزئیات مدل مخفی مارکوف به [۲۲] رجوع شود. مدل راست به چپ، پیچیدگی کمتری نسبت به مدل‌های پیچیده که از هر حالت به تمامی حالت دیگر می‌توان رفت، دارد و کارایی آن اثبات نشده که نسبت به مدل‌های پیچیده‌تر کمتر باشد. برای هر حرف فارسی یک نوع از مدل ذکر شده انتخاب شده و با توجه با اینکه حروف "س"، "ش" و "م" دو نوع شکل نوشتاری دارند، برای هر کدام از این اشکال نیز یک مدل در نظر گرفته شده است. از آنجائی که بازشناسی حروفی که بیش از یک بخش دارند ابتدا با بازشناسی جزء کوچکشان انجام می‌شود، برای هر کدام از اجزای کوچک نیز (غیر از نقطه) یک مدل در نظر گرفته شده است. تعداد حالت‌های پنهان برای هر حرف بر اساس سعی و خطا به دست آمده و همواره کوچکتر از ۲-کوتاهترین طول مشاهدات در مجموعه آموزشی برای هر حرف، در نظر گرفته شده‌اند. به عنوان مثال برای حرف "ا"، دو حالت پنهان و برای حرف "س"، هفت حالت پنهان در نظر گرفته شده است. تعداد حالت‌های قابل مشاهده مدل نیز برابر با تعداد انواع مشاهدات مختلف می‌باشند. بنابراین با توجه به در نظر گرفتن ۸ جهت برای رقمی کردن بردارهای ویژگی و همچنین در نظر گرفتن ویژگی‌های ساختاری حروف، تعداد ۱۱ حالت قابل مشاهده برای مدل‌های مربوط به حروف و برای مدل‌های مربوط به اجزای کوچک تعداد ۸ حالت قابل مشاهده در نظر گرفته شده است.



شکل ۹): ساختار استفاده شده برای طبقه‌بندی

۱۰- آموزش مدل

برای آموزش سیستم از ۹ نفر خواسته شد تا حداقل یک بار تمامی حروف فارسی را از "آ" تا "ی" بنویسند. تعداد حروف مجموعه آموزشی

۴- بین دو دره-راست متوالی حداقل برای چند نقطه حرکت قلم، در خلاف، یا هم جهت با محور عمودی شده باشد.

نقطه i را نقطه دره-چپ می‌نامیم اگر دارای چهار شرط زیر باشد:

۱- $v_y[i] < 0$ و $v_y[i+1] < 0$ باشند. یعنی حرکت بعد از نقطه i در

خلاف جهت محور عمودی باشد.

۲- $v_x[i-1] > 0$ ، $v_x[i] > 0$ و $v_x[i+1] > 0$ نباشند. یعنی حرکت

در نقطه i و قبل و بعد آن در جهت محور افقی نباشد.

۳- $deg[i-1] > 0$ ، $deg[i] > 0$ ، $deg[i+1] < 0$ باشند. یعنی تغییر زاویه

را در نقطه i شاهد باشیم.

۴- بین دو دره-چپ متوالی حداقل برای چند نقطه حرکت قلم، در

خلاف، یا هم جهت با محور عمودی شده باشد.

تمامی این شرایط با توجه به شکل و جهت نوشتاری حروف فارسی

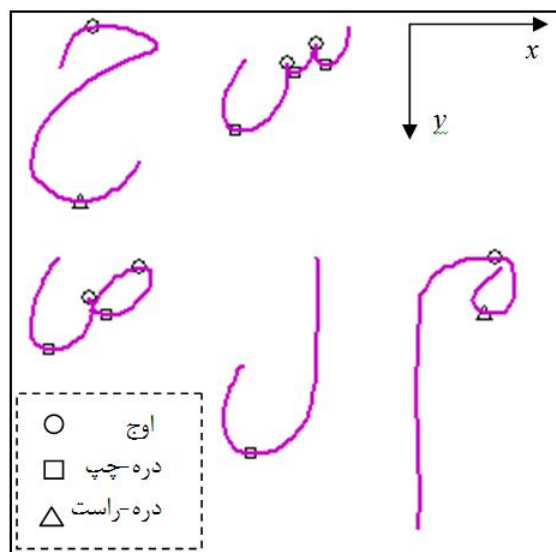
استخراج شده‌اند. در اکثر حروف فارسی یک یا چند نوع از این

ویژگی‌های ساختاری دیده می‌شوند. شکل (۸) چند حرف فارسی

(پیش‌پردازش شده) را به همراه ویژگی‌های ساختاری به دست آمده

روی آن‌ها نشان می‌دهد. ویژگی‌های ساختاری قبل از اعمال مراحل

پیش‌پردازش استخراج می‌شوند.



شکل (۸): چند حرف فارسی و ویژگی‌های ساختاری استخراج شده

بعد از به دست آوردن این نقاط و تعیین محل آن‌ها در مجموعه نقاط،

مسئله بعدی قرار دادن آن‌ها در بردار ویژگی و ترکیب کردن آن‌ها با

ویژگی‌های محلی است. در مقالات مختلف روش‌های متفاوتی پیشنهاد

شده است. در این مقاله برای نقاط اوج، دره-راست و دره-چپ به

ترتیب اعداد ۸، ۹ و ۱۰ در نظر گرفته شده و بسته به مکان رخدادشان

در بردار ویژگی نهایی بین زوایای رقمی شده قبلی (ویژگی‌های محلی)

قرار می‌گیرند. در نتیجه هر عنصر بردار ویژگی ساخته شده عددی

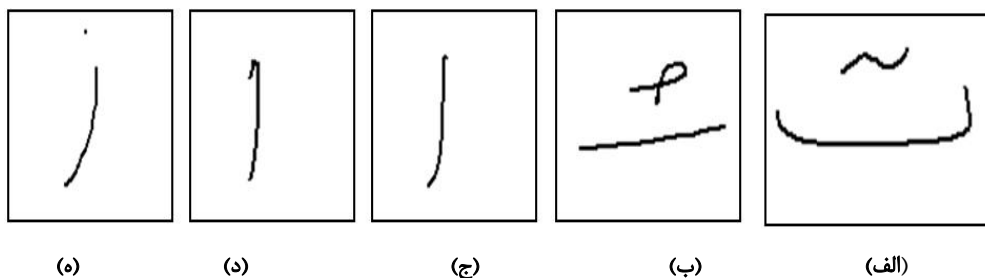
کارشناسی ارشد (زن و مرد، راست دست و چپ دست) دانشگاهی بودند. در شیوه نوشتاری افراد نیز هیچ‌گونه محدودیتی در نظر گرفته نشده است. کلیه پیاده‌سازی‌ها در محیط Visual C++ 2005 انجام شده و متوسط نرخ بازشناسی نیز برای هنگامی که حروف از قبل نوشته شده‌اند و به سیستم داده می‌شوند (و نه همزمان با نوشتن) برابر با ۲۴.۷ حرف در ثانیه، به دست آمده است. شکل (۱۰) تعدادی از خطاهای سیستم بازشناسی را نشان می‌دهد. در این شکل به ترتیب از (۱۰-الف) تا (۱۰-ه) مقصود نویسنده حروف "ت"، "ث"، "ا"، "آ" و "ز" بوده است که سیستم به ترتیب "ث"، "ژ"، "د"، "م" و "ذ" تشخیص داده است. در شکل ۱۰-الف در نوشتن دونقطه کمی انحنای نقاط بیشینه (اوج) و کمینه (حضيض) بیش از اندازه می‌باشند و قسمت ابتدایی تا اواسط آن مانند یک سه‌نقطه نوشته شده است. علت بروز خطا در این مورد، کم بودن این نوع نوشتاری دونقطه در مجموعه آموزشی (با در نظر گرفتن این نکته که این دونقطه از چپ به راست نوشته شده است) می‌باشد. همان طور که در سایر موارد شکل (۱۰) ملاحظه می‌شود اغلب خطاهای سیستم بازشناسی مربوط به شکل حروفی هستند که به گونه‌ای از حالت طبیعی‌شان خارج شده‌اند و پس از فرآیند پیش‌پردازش (مخصوصاً یکسان کردن اندازه) بیشتر مشابه حرف دیگری در مجموعه حروف فارسی شده‌اند. در اینجا باید ذکر شود که هدف از ارائه شکل (۱۰) نمایش دادن چند نمونه از خطاهای سیستم بوده و این طور نیست که سیستم بازشناسی، تمام حروف فارسی‌ای (مانند "ت" یا "ث") را که مشابه با یکی از شکل‌های موجود در این شکل نوشته شوند را نادرست تشخیص می‌دهد. بلکه خطاهای رخ داده در این شکل‌ها می‌توانند علت‌های متنوعی از جمله کم بودن این طرز نوشتاری در مجموعه آموزشی، لرزش دست نویسنده و حتی نوشتن برخلاف شیوه نوشتاری مرسوم حروف (مانند نوشتن حرف "ا" از پائین به بالا) را داشته باشند. بنابراین ویژگی‌های استخراج شده و در نتیجه حرف شناسایی شده، ممکن است منطبق با خواسته کاربر نباشد.

برابر با ۵۳۹ حرف است. از الگوریتم Baum-Welch برای آموزش مدل مخفی مارکوف استفاده شده است. نقطه ضعف این روش آموزش در صفر کردن وزن حالت‌های قابل مشاهده‌ای است که در طی فرآیند آموزش در یک حالت پنهان دیده نشده‌اند. بنابراین اگر در زمان به‌کارگیری سیستم، بردار مشاهداتی به دست آید که در تمامی عناصر به جز یک عنصر عیناً برابر با یک برداری در مجموعه آموزشی باشد و وزن این عنصر متفاوت در حالت (های) مربوطه‌اش صفر باشد، احتمال به دست آمده برابر با صفر خواهد بود. بنابراین انجام یک مرحله پس‌پردازش روی وزن‌های حالت‌های قابل مشاهده برای بالا بردن دقت سیستم ضروری است. پس از مرحله آموزش، در هر حالت پنهان مدلی، ابتدا در بین وزن‌های حالت‌های قابل مشاهده کوچکترین وزنی که صفر نباشد، یافت می‌شود. سپس وزن‌های حالت پنهان مربوطه که صفر هستند، به یک صدم وزن یافت شده، تغییر می‌یابند. از سایر وزن‌ها نیز مقدار مناسب کاسته می‌شود تا مجموع وزن‌های حالت‌های قابل مشاهده برابر با یک شوند.

۱۱ - نتایج

در مرحله بازشناسی و به‌کارگیری سیستم با توجه به دیاگرام رسم شده در شکل (۱) مراحل بازشناسی انجام می‌شوند. بر اساس بردار ویژگی استخراج شده، مدل حرفی که بالاترین احتمال را در بین مدل‌های نامزد کسب کند به عنوان حرف شناسایی شده در کنترل مربوطه روی واسط گرافیکی اضافه می‌شود و در صورتی که احتمال بردار ویژگی برای همه مدل‌های نامزد صفر به دست آید، مرحله بازشناسی حرفی را پیشنهاد نمی‌کند.

همان طور که در بخش قبل ذکر شد، مجموعه آموزشی برابر با ۵۳۹ حرف بود که توسط ۹ نفر نوشته شده بودند. میزان خطای سیستم روی این مجموعه برابر با ۲.۷۸٪ به دست آمد. به منظور اطلاع یافتن از قدرت عمومیت‌دهی سیستم بازشناسی و انجام آزمایش مستقل از نویسنده، از ۸ نفر دیگر که در مرحله آموزش حضور نداشتند خواسته شد تا حداقل یک بار حروف فارسی را از "آ" تا "ی" بنویسند. تعداد حروف این مجموعه برابر با ۴۳۱ حرف است. میزان خطای بازشناسی بر روی این مجموعه نیز برابر با ۵.۱٪ به دست آمد. افراد شرکت‌کننده در این آزمایش‌ها تعدادی از دانشجویان و فارغ‌التحصیلان کارشناسی و



شکل (۱۰): تعدادی از خطاهای سیستم بازشناسی

۱۲ - نتیجه گیری

در این مقاله سیستمی برای بازشناسی برخط حروف فارسی ارائه شد. مراحل این بازشناسی از یک سو شامل غربال کردن حروف بر اساس تعداد بخش‌ها و اجزای کوچکشان و از سوی دیگر شامل یکسان کردن اندازه بدنه حروف، دوباره نمونه برداری و استخراج ویژگی می‌باشد. در نهایت نتایج حاصل از این دو مسیر موازی که خروجی اولی تعدادی حرف نامزد و خروجی دومی بردار ویژگی حرف ناشناخته است در مرحله طبقه‌بندی ترکیب می‌شوند. مرحله غربال کردن حروف بر اساس تعداد بخش‌های آن‌ها و همچنین بر اساس اجزای کوچکشان باعث کاهش تعداد حروف نامزد در مرحله بازشناسی و در نتیجه بالا رفتن دقت آن شده است. استفاده از مراحل یکسان کردن اندازه حروف به اندازه از پیش تعریف شده‌ای که بر اساس میزان گسترش حروف در جهت محور افقی است، و همچنین به‌کارگیری صافی گوسی و اتصال سهمی‌گون نقاط متوالی باعث مقاوم شدن ویژگی‌های استخراج شده به دست خط‌های مختلف شده می‌باشد. استخراج ویژگی‌های ساختاری مناسب برای حروف فارسی نیز در بالا بردن دقت بازشناسی سهم به‌سزایی داشته است. به منظور طبقه‌بندی نیز از مدل مخفی مارکوف استفاده شده است که یک طبقه‌بند مناسب جهت کاربردهایی است که خاصیت زمانی و ترتیب وقوع عناصر بردارهای ویژگی در آن‌ها مهم می‌باشند. نتایج حاصل از آزمایش‌ها حاکی از دقت مطلوب سیستم بازشناسی می‌باشد.

اغلب خط‌های سیستم بازشناسی نیز یا مربوط به شکل حروفی هستند که به گونه‌ای از حالت طبیعی‌شان خارج شده‌اند و پس از فرآیند پیش‌پردازش، مشابه حرف دیگری در مجموعه حروف فارسی شده‌اند و یا مربوط به حروفی می‌باشند که به نوع خاصی نوشته شده و در مجموعه آموزشی به ندرت رخ داده‌اند. بنابراین کارهای آینده در این زمینه می‌تواند در انجام پیش‌پردازش‌های مناسب‌تر و یا استخراج ویژگی‌هایی باشد که نسبت به این گونه خط‌ها مقاوم باشند.

مراجع

- in Information Science and Information Technology, pp. 787-795, 2005.
- [4] R. Niels, "Dynamic Time Warping: An Intuitive Way of Handwriting Recognition?" Master's thesis, Radboud University Nijmegen, 2004.
- [5] R. Niels and L. Vuurpijl, "Using Dynamic Time Warping for Intuitive Handwriting Recognition". In Advances in Graphonomics, proceedings of the 12th Conference of the International Graphonomics Society (IGS2005), Eds. A. Marcellini and C. De Stefano, pp. 217-221, Salerno, Italy, Jun. 26-29, 2005.
- [6] J. Di Martino, "Dynamic time warping algorithms for isolated and connected word recognition". In Nato Asi Series, Vol. F16, New Systems and Architectures for Automatic Speech Recognition and Synthesis. Edited by R. De Mori and C.Y. Suen, Springer-Verlag Berlin Heidelberg 1985.
- [7] V. Vuokko, "Adaptive Methods for On-Line Recognition of Isolated Handwritten Characters". PhD thesis, Finnish Academies of Technology, 2002.
- [8] S. Salvador and P. Chan, "FastDTW: Toward accurate dynamic time warping in linear time and space". 3rd Workshop. On Mining Temporal and Sequential Data, ACM KDD '04. Seattle, Washington, Aug. 22-25, 2004.
- [9] R. Dugad and U. B. Desai, "A Tutorial On Hidden Markov Models", Signal Modeling Techniques In Speech Recognition, Proceedings of the IEEE, vol. 81, no. 9, Sep 1993.
- [10] J. Hu, S. G. Lim and M. k. Brown, "Writer Independent On-Line Handwriting Recognition Using an HMM Approach," Pattern Recognition, vol. 33, pp. 133-147, Jan. 2000.
- [11] J. Hu, S. G. Lim and M. k. Brown, "HMM Based Writer Independent On-Line Handwritten Character and Word Recognition," Proc. Int'l Workshop Frontiers in Handwriting Recognition, pp.143-155, 1998.
- [12] J. Hu, M. K. Brown and W. Turin, "Handwriting Recognition with Hidden Markov Models and Grammatical Constraints," in Proceedings Fourth Int. Workshop on Frontiers in Handwriting Recognition, Taipei, Taiwan, pp. 195-205, December 1994.
- [۱۳] رضوی، س.م. و کبیر، ا.، "روشی ساده برای بازشناسی برخط حروف مجزای فارسی"، نشریه دانشکده مهندسی دانشگاه فردوسی مشهد، ۱۳۸۳.
- [۱۴] رضوی، س.م. و کبیر، ا.، "بازشناسی برخط زیرکلمات فارسی با استفاده از نقاط و علائم حروف"، دومین کنفرانس بین المللی
- [1] R. Plamondon and N. Shihari, "On-line and Off-line Handwriting Recognition: A Comprehensive Survey", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 1, pp. 63-84, 2000.
- [2] G. Seni, N. Nasrabadi and R. Srihari, "An on-line cursive word recognition system", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1994.
- [3] W. A. Salameh and M. A. Otair, "Online Handwritten Character Recognition Using an Optical Backpropagation Neural Network", Issues

زیرنویس‌ها

- 1- On-line
- 2- Off-line
- 3- Time Delay Neural Network
- 4- Back propagation
- 5- Dynamic Time Warping
- 6- Hidden Markov Model
- 7- Writer Dependent
- 8- Writer Independent

فناوری اطلاعات و دانش، دانشگاه صنعتی امیرکبیر، ۳-۵ خرداد ۱۳۸۴.

[۱۵] رضوی، س.م. و کبیر، ا.، "روشی ساده برای بازشناسی برخط زیرکلمات فارسی"، نشریه مهندسی برق و مهندسی کامپیوتر ایران، سال ۳، شماره ۲، پاییز و زمستان ۱۳۸۴.

[۱۶] رضوی، س.م. و کبیر، ا.، "بازشناسی برخط کلمات دست‌نویس فارسی با واژگانی گسترده"، پنجمین کنفرانس ماشین بینایی و پردازش تصویر ایران، دانشگاه تبریز، ۱۳۸۷.

[17] H. S. M. Beigi, K. Nathan, J. G. Clary and J. Subrahmonia, "Challenges of Handwriting Recognition in Farsi, Arabic and Other Languages with Similar Writing Styles in An On-line Digit Recognizer", Proceedings of the 2nd Annual Conference on Technological Advancements in Developing Countries, Columbia University, New York, Jul. 23-24, 1994.

[18] F. Biadisy, J. El-Sana and N. Habash, "Online Arabic handwriting recognition using Hidden Markov Models". In Proceedings of the 10th International Workshop on Frontiers of Handwriting and Recognition, 2006.

[19] M. Soleymani Baghshah, S. Bagheri Shouraki and S. Kasaei, "A Novel Fuzzy Approach to Recognition of Online Persian Handwriting", In: Proc. The 5th International Conference on Intelligent Systems Design and Applications, pp. 268-273, Wroclaw, Poland, September 8-10, 2005.

[۲۰] مسعود فرکی، نادر کریمی و مازیار پالهنک، "شناسایی حروف فارسی در وایت‌برد بدون ماژیک"، دومین کنفرانس ملی داده کاوی، دانشگاه صنعتی امیرکبیر، آبان ۱۳۸۷.

[21] D. Douglas and T. Peucker, "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature". The Canadian Cartographer, 10(2):112-122, 1973.

[22] L. R. Rabiner, "A tutorial on Hidden Markov Models and selected applications in speech recognition". IEEE Proceedings on Readings in Speech Recognition, 77(2):257-286, 1989.

[23] Dr Thomas Sederberg, BYU Bézier curves, http://www.tsplines.com/resources/class_notes/Bezier_curves.pdf.