پنجمین کنفرانس سیستمهای فازی ایران
هفدهم الی نوزدهم شهریورماه ۱۳۸۳

کاربرد درخت تصمیم‌گیری فازی در تشخیص اشارات الفبای فارسی

رقیه یزدانی[۱]، محمدرضا کنگاوری[۲]
دانشگاه علم و صنعت ایران
r_yazdani@nigc.ir

چکیده

ابزارهای ورودی و خروجی جدیدی که طراحی می‌شوند سعی در هوشمندکردن هرچه بیشتر ارتباط بین انسان و کامپیوتر دارند. یکی از روشهایی که برای رسیدن به این هدف می‌تواند مفید باشد، استفاده از تشخیص اشارات دست به وسیله کامپیوتر است. یکی از کاربردهای تشخیص اشاره مربوط به تشخیص زبان اشاره (زبان افراد ناشنوا) می‌باشد که می‌تواند برای بهبود ارتباط بین افراد ناشنوا با افراد شنوا مفید باشد.

در این مقاله از روش درخت تصمیم فازی برای تشخیص اشارات الفبای فارسی استفاده شده است. به کمک نمونه های مجموعه رشد، درخت فازی تولید شده و سپس بر اساس دو متد مختلف و با استفاده از نمونه‌های مجموعه هرس، درخت فازی تولید شده هرس گردیده است و در نهایت با کمک دانش تولید شده از درخت فازی، اقدام به تشخیص اشارات نموده‌ایم. نتایج نشان می‌دهد که استفاده از تکنیکهای یادگیری و بخصوص درخت تصمیم فازی می‌تواند به عنوان روشی مناسب در زمینه تشخیص اشارات مطرح شود. براساس این نتایج درخت فازی تولید شده بر اساس معیار ابهام کلاسبندی بهترین دقت را برای تشخیص اشارات الفبای فارسی تولید می‌کند.

واژه های کلیدی : زبان اشاره – تشخیص اشاره – درخت تصمیم فازی – خصیصه غالب – هرس

۱- مقدمه

اشارات دست (Hand Gestures) را می‌توان از نقطه نظر علمی به دو دسته جدا تقسیم کرد : اشارات دینامیک و اشارات استاتیک. حرکت یک بخش اساسی در اشاره دینامیک است در حالیکه یک اشاره استاتیک، موقعیتی خاص از دست و پیکربندی آن است. بنابراین می‌توان گفت یک اشاره دینامیک از چند اشاره استاتیک تشکیل شده است. [1] روشهای تشخیص اشاره(Gesture Recognition) را می‌توان به دو گروه کلی تقسیم نمود :

۱) روشهایی که از یک دستکش مخصوص (Cyber Glove) استفاده می‌کنند.

۲) روشهایی که از تکنیک‌های بینایی کامپیوتر(Computer Vision) بهره می‌گیرند.

دستکش‌هایی که در روش اول استفاده می شوند، سنسورهای متعددی دارند که اطلاعات دست را تأمین می‌کنند. نوع اولیه این دستکشها در سال ۱۹۷۰ توسط زیمرمن (Zimerman) توسعه داده شد. این دستکش دارای سنسور فیبر نوری در پشت انگشتان می‌باشد. خم کردن انگشتان موجب خم شدن فیبر می‌شود و از طریق نور، این خمیدگی قابل انتقال می‌باشد.

۱- کارشناس ارشد هوش مصنوعی – دانشکده کامپیوتر – دانشگاه علم و صنعت ایران

۲- استادیار - دانشکده کامپیوتر – دانشگاه علم و صنعت ایران، kangavari@iust.ac.ir

سیگنال آنالوگی که به یک پردازشگر ارسال می‌شود زوایای اتصال انگشتان را بیان می‌کند. این روش در مقایسه با روشهای مبتنی بر بینایی کامپیوتر، از مزایایی چون هزینه کمتر، قدرت پردازش real-time و توانایی استخراج داده قطعی برخوردار است. از معایب آن مقید کردن کاربر به استفاده از دستکش است. همچنین از آنجا که این دستکشها اندازه یکسانی دارند، لذا برای استفاده هر کاربر باید با پارامترهای خاصی کالیبره شوند. [1]

در روش دوم با استفاده از دوربین، تصاویری دیجیتالی از اجرای اشارات تهیه می‌شود. در این تکنیکها، بحثهای مختلفی مطرح می‌شود از جمله مدل‌کردن شکل دست، مدل کردن ساختار حرکتی دست، ردیابی (Tracking) حرکت دست در فریم های متعدد، تجزیه و تحلیل حرکت دست و ... . تکنیکهای مختلفی که تاکنون برای تشخیص اشارات استفاده شده است، عموماً مبتنی بر موارد زیر می‌باشند: پایگاه دانش قوانین(Rule-Base)، مدل مخفی مارکف(Hidden Markov Model)، شبکه های بایزین (Bayesian Networks) و شبکه های عصبی (Neural Networks).[2]

تشخیص اشارات در موارد متعددی از جمله طراحی سه بعدی، واقعیت مجازی (Virtual Reality)، تشخیص زبان اشاره (Sign Language)، کنترل روبوت و کنفرانس از راه دور بکار گرفته شده است.[1]

در این مقاله، از روش درخت تصمیم‌گیری فازی (FDT : Fuzzy Decision Tree) جهت تشخیص اشارات مربوط به الفبای فارسی استفاده شده است. بدلیل در اختیار نبودن دستکش مخصوص و یا سخت‌افزار مورد نیاز مبتنی بر بینایی کامپیوتر جهت استخراج موقعیت دست از تصاویر ویدئویی ، لذا امکان تهیه داده به روشهای معمول وجود نداشت. به همین خاطر از یک روش دستی برای استخراج قوانین موقعیت دست از تصاویر بهره گرفته شد و سپس بر اساس این قوانین، داده‌هایی بصورت تصادفی برای اشارات مربوطه تولید شد که بخشی از آنها برای ساخت درخت تصمیم، بخشی برای هرس (Pruning) درخت تصمیم و باقیمانده برای تست درخت هرس شده بکار گرفته شدند. برای ساخت درخت تصمیم فازی از روشهای مختلفی برای تعیین خصیصه (Attribute) غالب در هر گره استفاده شده است. همچنین دو روش جهت هرس درخت پیاده شد که در مجموع باعث کارایی بهتر درخت گردید.

۲- درخت تصمیم فازی و هرس آن

مهمترین خصوصیت درختهای تصمیم، قابلیت آنها در شکستن فرایند پیچیده تصمیم‌گیری به یک مجموعه از تصمیمات ساده‌تر است که براحتی قابل تفسیر هستند. ID3 که یک متد مؤثر و عمومی تصمیم‌گیری برای کلاسبندی داده‌های فازی است معمولاً برای کار با داده‌های عددی مناسب نیست و این در حالیست که بیشتر مسائل دنیای واقعی بصورت غیر فازی (عددی، پیوسته) رفتار می‌کنند. ترکیب مجموعه‌های فازی با درختهای تصمیم، توانایی استفاده از قابلیتهای استنتاج تقریبی اولی و قابلیت درک آسان دومی را دارد. الگوریتم‌های ساخت درخت تصمیم فازی اساساً توسعه یافته الگوریتم های ساخت درخت تصمیم قطعی (CDT: Crisp Decision Tree) هستند. [3] نتایج آزمایشات نشان می‌دهد که قابلیت عمومیت (Generalization) در درخت تصمیم فازی بیش از درخت تصمیم قطعی است.[4]

الگوریتم تولید درخت فازی بر روی مجموعه‌های آموزشی فازی اعمال شده و یک درخت تصمیم فازی تولید می‌کند. یک درخت تصمیم فازی متشکل از گره‌ها به عنوان خصیصه آزمون، یالها برای انشعاب ناشی از مقادیر فازی (ارزشهای فازی خصیصه آزمون) و برگها به عنوان برچسب کلاسها با مقدار قطعیت می‌باشد. دو مسأله مهمی که در ساخت و استفاده از درخت تصمیم فازی مطرح می‌شود [5]، عبارتست از :

۱) روش انتخاب بهترین خصیصه در هر نود برای توسعه درخت : هر معیاری که برای تقسیم کردن نمونه‌ها در یک گره هنگام ساخت درخت تصمیم قطعی استفاده می‌شود، در اینجا باید تغییر کند زیرا هنگام تقسیم کردن نمونه‌ها در یک گره، ممکن است یک نمونه در چند یال منشعب از گره ظاهر شود. در ادامه سه روش انتخاب بهترین خصیصه مطرح می‌شود.

۲) روش استنتاج در درخت تصمیم فازی : در درخت تصمیم قطعی به هنگام کلاسبندی یک نمونه جدید، به کمک مقادیر خصیصه‌ها در این نمونه، حداکثر به یک گره برگ خواهیم رسید، که برچسب کلاس آن برگ، کلاس پیشنهادی برای نمونه مورد نظر است اما هنگام کلاسبندی این نمونه با استفاده از یک درخت تصمیم فازی، ممکن است به گره‌های برگ متعددی

برسیم که هر کدام با یک درجه قطعیت متمایز، کلاسی را برای نمونه پیشنهاد می‌کنند و حال باید با استفاده از مکانیزم مناسبی بهترین کلاس پیشنهادی را انتخاب کرد.

۳- یادگیری درخت تصمیم فازی

تمام نمونه‌های موجود را به دو مجموعه آموزشی و تست تقسیم می‌کنیم، که این نسبت تقسیم متفاوت است ولی نسبت ۷۰٪ به ۳۰٪ متداولتر است. سپس نمونه‌های مجموعه آموزشی را به دو زیرمجموعه دیگر به نامهای مجموعه رشد (Growing Set) و مجموعه هرس (Pruning Set) افراز می‌کنیم که در اینجا هم از نسبت تقسیم ۷۰٪ به ۳۰٪ استفاده شده است. از مجموعه رشد برای ساخت درخت استفاده می‌کنیم و مجموعه هرس را برای هرس درخت فازی تولید شده بکار می‌بریم و سرانجام دقت درخت نهایی را با استفاده از نمونه‌های مجموعه تست بدست می‌آوریم.[6]

در این مقاله برای توسعه درخت، سه روش انتخاب خصیصه غالب در هر نود بکار گرفته شده و در نهایت نتایج بدست آمده از این سه روش مختلف با یکدیگر مقایسه شده است. این روشها عبارتند از :

۱) معیار بهره اطلاعاتی (IG : Information Gain) [7]
۲) معیار ابهام کلاسبندی (AM : Ambiguity Measure) [8]
۳) معیار انتروپی رخدادهای فازی (SE : Shannon Entropy) [8]

برای استنتاج از درخت تصمیم فازی، به ازای هر برگ باید مسیری را که از ریشه به آن منتهی می‌شود در نظر گرفت و درجه عضویت مقدار قطعی نمونه در هر خصیصه موجود در مسیر را به ترم زبانی تعیین شده روی یال مسیر تعیین کرد. سپس باید ترکیب تمام این درجات عضویت را با یک تابع S-norm محاسبه کرده و نتیجه را با درجه عضویت تعیین شده در برگ برای هر کلاس، مجدداً ترکیب S-norm نمود.[7]

پس از اینکه این عملیات به ازای هر برگ انجام گرفت، در نهایت درجات عضویت بدست آمده برای این نمونه به هر کلاس در برگهای مختلف، با هم ترکیب T-norm می‌شود. حال می‌توان تصمیم گرفت که مثلاً درجه عضویت نمونه به هر کلاس بیشتر بود، نمونه متعلق به آن تصور شود. در متد استنتاج «+ − × = ×» برای S-norm از ضرب و برای T-norm از جمع استفاده می‌شود. می‌توان از توابع S-norm و T-norm دیگری هم استفاده کرد.[7]

برای افزایش دقت و کاهش پیچیدگی، درخت فازی تولید شده به روشهای بالا را هرس می‌نماییم. با توجه به اینکه هیچگونه تکنیک خاصی برای هرس درخت تصمیم فازی وجود ندارد، در این مقاله تغییراتی در دو متد "هرس کمترین خطا" (MEP : Minimal Error Pruning) و "هرس خطای کاهش یافته" (REP : Reduced Error Pruning) صورت گرفت و سپس برای هرس درختهای تولید شده بکار گرفته شدند.

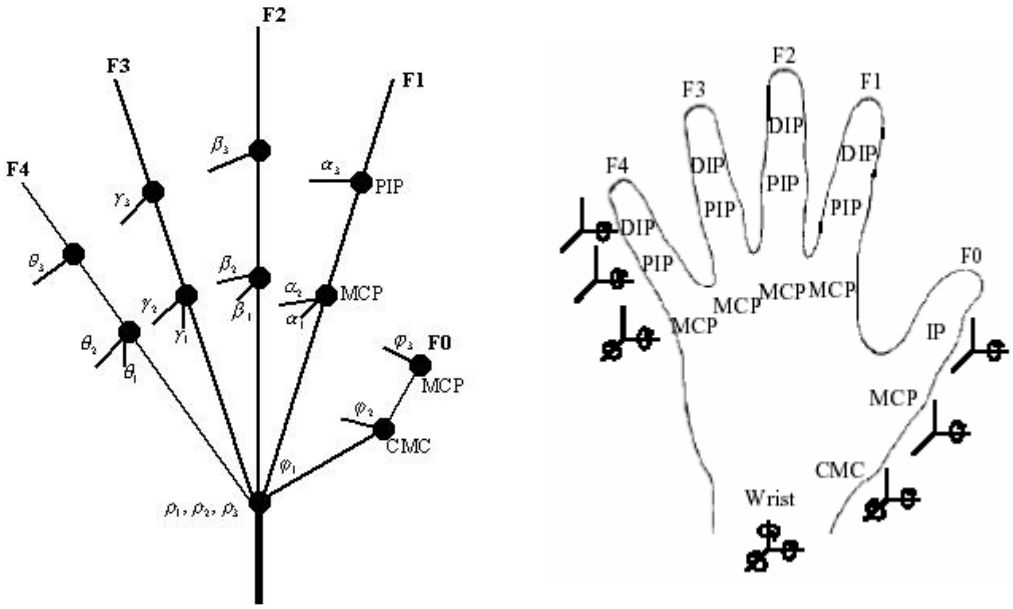
هرس خطای کاهش یافته : برای هرس با این روش، بایستی درخت را به‌صورت پایین به بالا و با ترتیب post-order پیمایش نمود. به ازای هر گره غیر برگی که در مسیر پیمایش با آن برخورد می‌کنیم، باید خطای کلاسبندی را در آن گره و در زیر درخت مربوط به آن گره محاسبه کرده و چنانچه خطای کلاسبندی گره کمتر یا مساوی خطای کلاسبندی زیر درخت آن گره باشد، باید آن زیر درخت را هرس نمود و گره مربوطه را به عنوان یک برگ در نظر گرفت. لازم به ذکر است که در این متد، ازمجموعه هرس برای هرس درخت استفاده می‌شود.[9]

هرس کمترین خطا : در این روش نیز درخت بصورت پایین به بالا و با ترتیب Post-order پیمایش می‌شود. به ازای هر گره غیر برگ، خطای استاتیک و خطای دینامیک را در آن گره محاسبه می‌کنیم و چنانچه خطای استاتیک کمتر یا مساوی خطای دینامیک باشد، درخت را هرس کرده و این گره تبدیل به یک برگ خواهد شد. در این روش از مجموعه هرس جداگانه‌ای برای هرس کردن درخت استفاده نمی‌شود و همان مجموعه رشد در اینجا هم بکار می‌رود.[10]

۴- تهیه مجموعه آموزشی برای اشارات الفبای زبان فارسی

محققین زیادی در زمینه تشخیص زبان اشاره کار کرده اند، از جمله زبانهای اشاره ای که در این راستا تحقیقاتی روی آنها صورت گرفته است عبارتند از : زبان اشاره استرالیایی (AUSLAN : Australian Sign Language)، زبان اشاره امریکایی (ASL : American Sign Language)، زبان اشاره انگلیسی (BSL : British Sign Language)، زبان اشاره تایوانی (TSL : Taiwanese Sign Language)، زبان اشاره چینی (CSL : Chinese Sign Language) و زبان اشاره ژاپنی (JSL : Japanese Sign Language).

در رابطه با زبان اشاره فارسی تاکنون کاری در این زمینه صورت نگرفته است. پیوست ۱ اشارات مربوط به الفبای زبان فارسی را نشان می‌دهد. این اشارات توسط کمیته پژوهش و توسعه زبان اشاره در سازمان بهزیستی کشور استاندارد شده است. دست در حالت کامل و ایده آل ۲۶ درجه آزادی (Degree Of Freedom) دارد. شکل ۱ – (الف) ملاحظه شود. هر انگشت دارای چهار درجه آزادی است، حرکت انتقالی دست سه درجه آزادی و حرکت چرخشی مچ هم سه درجه آزادی دارد. (جمعاً ۲۶ درجه آزادی) بر اساس کاربردهای مختلف می‌توان این مدل را ساده‌تر کرد. از آنجا که در این پروژه تنها با اشارات استاتیک (اشارات الفبای فارسی) سر و کار داریم، لذا می‌توان از سه درجه آزادی مربوط به انتقال در مچ دست، صرف نظر کرد. همچنین معمولاً زوایای PIP و DIP با هم خم می‌شوند. زاویه IP از انگشت F0 هم وضعیتی مشابه زوایای DIP دارد، بنابراین برای ساده کردن مدل از زوایای DIP در انگشتان F1 تا F4 و زاویه IP از انگشت F0 صرف نظر می‌کنیم. شکل۱ - (ب) مدل پیشنهادی دست و زوایای مربوطه را برای تشخیص اشارات الفبای فارسی نشان می‌دهد.



(الف) (ب)

شکل ۱ – (الف) نمایشی از تمام درجات آزادی دست [11] (ب) مدل پیشنهادی برای اشارات زبان فارسی

اگر اشارات پایه ای مختلف را به عنوان کلاسهای جداگانه ای در نظربگیریم، آنگاه می‌توان هریک از این زوایای مربوط به درجات آزادی مختلف را به عنوان خصیصه های کلاس اشاره فرض کرد که ارزشهای متنوع این خصیصه‌ها در ترکیب با یکدیگر، می‌تواند ما را به سمت کلاس اشاره خاصی هدایت کند. تمامی این خصیصه‌ها دارای مقدار حقیقی و در نتیجه ارزش پیوسته هستند. هدف نهایی در این مقاله ساخت درخت تصمیم گیری بر اساس این خصیصه‌ها است. از انجا که مقادیر تمامی خصیصه‌ها از نوع عدد حقیقی می‌باشند، لذا نمی‌توان مرز دقیقی بین دامنه‌های مختلف ارزش خصیصه‌ها برای جداسازی کلاسهای متفاوت از یکدیگر تعیین کرد و درنتیجه یک درخت تصمیم فازی کارایی بهتری از درخت تصمیم قطعی خواهد داشت. برای ساخت درخت تصمیم فازی، اولین قدم تعیین متغیرهای فازی و ترمهای زبانی آنها است.

هریک از زوایای مربوط به درجات آزادی مختلف را به‌عنوان یک متغیر فازی در نظر می‌گیریم. پیوست ۲ این متغیرهای فازی و ترمهای زبانی را به همراه توابع عضویتشان نشان می‌دهد.

۵- پیاده سازی سیستم تشخیص اشارات

برای ارزیابی و مقایسه معیارهای مختلف جهت توسعه درخت فازی و همچنین روشهای استفاده شده در این پروژه برای هرس درخت فازی، سیستم تشخیص اشارات با استفاده از نرم افزار Microsoft Visual C++ پیاده سازی گشت که نتایج بدست آمده از آن در ادامه آورده شده است. قبل از بررسی این نتایج، لازم به ذکر است که وقتی برنامه با تعداد نمونه مشخصی اجرا می‌شود، همانگونه که در قسمت قبلی مطرح شد ابتدا این نمونه‌ها به سه مجموعه رشد، هرس و تست تقسیم می‌شوند. برای ساخت درخت، تنها از نمونه‌های مجموعه رشد استفاده می‌شود. مجموعه هرس بسته به متدی که برای هرس درخت استفاده می‌شود، بکار خواهد رفت و مجموعه تست نیز برای تعیین دقت درخت تولید شده استفاده می‌شود. جدول ۱ تعداد نمونه‌های مختلفی که در اجراهای متفاوت برنامه استفاده شده است را نشان می‌دهد. هر سطر از این جدول مشخص می‌کند که به هنگام اجرای برنامه با تعداد نمونه خاصی، چه تعدادی از آنها برای مجموعه رشد، چه تعداد برای مجموعه هرس و چه تعداد برای مجموعه تست در نظر گرفته می‌شود.

جدول ۱ - نحوه تقسیم بندی نمونه‌های آموزشی به سه مجموعه رشد و هرس و تست

Examples No.	Growing No.	Pruning No.	Testing No.
5	3	1	1
10	5	2	3
15	8	3	4
20	10	4	6
25	13	5	7
30	15	6	9
40	20	8	12
50	25	10	15

در جدول ۲ نتایج بدست آمده از محاسبه دقت برای ارزیابی درختهای فازی تولید شده با روشهای مختلف و به ازای تعداد نمونه‌های متفاوت نمایش داده شده است. به عنوان مثال دومین سطر از این جدول نشان می‌دهد که چنانچه برای ساخت درخت از روش ابهام کلاسبندی (AM) استفاده شود وسپس آن درخت با دو متد MEP و REP بصورت جداگانه هرس شود، به ازای هرکدام از این سه درخت (درخت بدون اعمال هرس، درخت هرس شده با روش MEP، درخت هرس شده

با REP) با استفاده از نمونه‌های مجموعه تست و متد استنتاج $+,\times,\times$ چه دقتهایی بدست می‌آید. لازم به ذکر است که دقتهای بدست آمده، میانگین دقتهای محاسبه شده از ۱۰ اجرای متوالی است که به ازای هر اجرا نحوه تقسیم بندی نمونه‌ها به سه مجموعه رشد و هرس و تست بصورت تصادفی تغییر یافته است. در این جدول پس از نتایج دقتهای بدست آمده از کاربرد روش AM در ساخت درخت، از سطر نهم به بعد، نتایج دقتهای محاسبه شده برای استفاده از روش انتروپی رخدادهای فازی (SE) آورده شده است و از سطر هفدهم نیز نتایج بدست آمده از کاربرد روش بهره اطلاعاتی (IG) در ساخت درخت نمایش داده شده است.

جدول ۲ – میانگین دقت تشخیص اشارات فارسی در ۱۰ تکرار متوالی
با استفاده از روشهای مختلف تولید درخت فازی

Examples No. (10 iteration)	Used Ambiguity Measure	Used MEP	Used REP
	Accuracy	Accuracy	Accuracy
5	0.77	0.74	0.62
10	0.84	0.81	0.78
15	0.85	0.83	0.81
20	0.88	0.87	0.83
25	0.88	0.87	0.85
30	0.90	0.89	0.87
40	0.91	0.89	0.88
	Used Shannon Entropy	Used MEP	Used REP
	Accuracy	Accuracy	Accuracy
5	0.79	0.81	0.75
10	0.80	0.81	0.83
15	0.81	0.83	0.88
20	0.83	0.83	0.85
25	0.82	0.83	0.84
30	0.82	0.84	0.87
40	0.83	0.84	0.87
	Used Information Gain	Used MEP	Used REP
	Accuracy	Accuracy	Accuracy
5	0.78	0.81	0.76
10	0.78	0.79	0.83
15	0.81	0.83	0.87
20	0.83	0.83	0.86
25	0.81	0.82	0.84
30	0.81	0.83	0.87
40	0.83	0.85	0.88

این جدول نشان می‌دهد که روش AM بطور یکنواخت تری نسبت به بقیه رفتار می‌کند و تا حد زیادی قابل پیش‌بینی است. با استفاده از روش AM در ساخت درخت، هرچه تعداد نمونه‌های آموزشی افزایش یابد دقت درخت تولیدشده نیز بالاتر خواهد رفت. اما چنانچه از روشهای SE و یا IG برای توسعه درخت استفاده شود، همیشه اینگونه نخواهد بود. به ازای این دو روش هنگام استفاده از ۲۰ نمونه آموزشی، درخت تولید شده بهترین دقت را داشته است و با افزایش تعداد نمونه‌ها، دقت کاهش یافته است و مجدداً هنگام استفاده از ۴۰ نمونه آموزشی دقت افزایش یافته است.

نکته دیگری که از این جدول نتیجه می‌شود اینست که استفاده از هرس، دقت درخت را به میزان کمی جابجا می‌کند. هنگامی که از روش AM برای ساخت درخت استفاده می‌شود، همواره هنگام استفاده از هرس، دقت درخت کاهش یافته است و از بین دو روش هرس، متد REP دقت را به میزان بیشتری کاهش می‌دهد. اما زمانی که از روش SE یا IG برای توسعه درخت بهره گرفته می‌شود، بالعکس استفاده از هرس در اکثر موارد دقت درخت را افزایش می‌دهد. تنها موردی که منجر به کاهش دقت درخت پس از هرس شده است هنگام استفاده از تعداد نمونه آموزشی ۵ و متد هرس REP می‌باشد.در مورد این دو روش توسعه درخت، متد REP دقت را به میزان بیشتری افزایش می‌دهد.

۶- نتیجه گیری

این مقاله اولین کار در زمینه تشخیص اشارات فارسی است که بر اساس تکنیکهای یادگیری عمل می‌کند. در نمونه کارهای دیگری که بر اساس پایگاه دانش عمل می‌کنند، عموماً دانش را بصورت دستی استخراج و فرموله می‌کنند.[11] اما در اینجا دانش از درخت تصمیم فازی بدست می‌آید و سپس از این دانش تولید شده برای تشخیص اشارات بهره گرفته شده است. در اینجا اگرچه بر اساس داده های تصادفی درخت تصمیم فازی تولید شده، ولیکن در جمع آوری این داده‌ها حداکثر دقت اعمال گردیده تا هرچه طبیعی تر باشند. نتایج روش پیشنهادی نشان می‌دهد که استفاده از تکنیکهای یادگیری ماشین می‌تواند به تولید دانش مؤثر منجر گردد و از آنجا که داده‌های اشارات دست دارای فرمت خاصی (پیوستگی و همپوشانی) هستند، روش یادگیری فازی نتیجه مطلوبی را نتیجه می‌دهد.

بر اساس نتایج بدست آمده از اجراهای مختلف می‌توان گفت استفاده از روش AM برای ساخت درخت و استفاده از متد « +,×,× » هنگام استنتاج بدون اعمال هیچ‌گونه هرسی بهترین دقت را برای تشخیص اشارات الفبای فارسی خواهد داشت اما در صورتیکه بخواهیم از روش SE یا IG برای ساخت درخت استفاده کنیم، برای بهبود دقت با ید از هرس استفاده شود و ترجیحاً بهتر است که متد هرس REP بکار رود.

مراجع

[1] H. Brik and T. B. Moeslund, “Recognizing gestures from the hand alphabet using principal component analysis”, Master’s Thesis, Laboratory of Image Analysis, aalborg University, Denmark, 1996

[2] Y. Wu and T. S. Huang, “Hand modeling, analysis and recognition”, in IEEE Signal Processing Magazine, pp. 51-60, May 2001

[3] S. Mitra, K. M. Konwar and s. K. Pal, “Fuzzy decision trees, linguistic rules and fuzzy knowledge-based network: generation and evaluation”,IEEE, 2002

[4] D. S. Yeung, J. SUN and X. Z Wang, “An initial comparison of generalization- capability between crisp and fuzzy decision trees”, Proceedings of the First Internationl Conference on Machine Learning and Cybernetices, november 2002

[5] C. Z. Janikow, “fuzzy decision trees: Issues and methods”, IEEE Transactions on Systems, Man and Cybernetics, vol. 28, 1998

[6] F. Esposito, D. Malerba and G. Semeraro, “A comparative analysis of methods for pruning decision trees”, IEEE Transaction Analysis and Machine Intelligence, vol. 19, No. 5, May 1997

[7] M. Umano, H. Okamoto, H. Tamura, F. Kawachi, S. Umedzu and J. Kinoshita, “ Fuzzy decision trees by fuzzy id3 algorithm and its application to diagnosis systems”, Department of Systems Engineering and Precision Engineering, Osaka University, Japan, IEEE, 1994

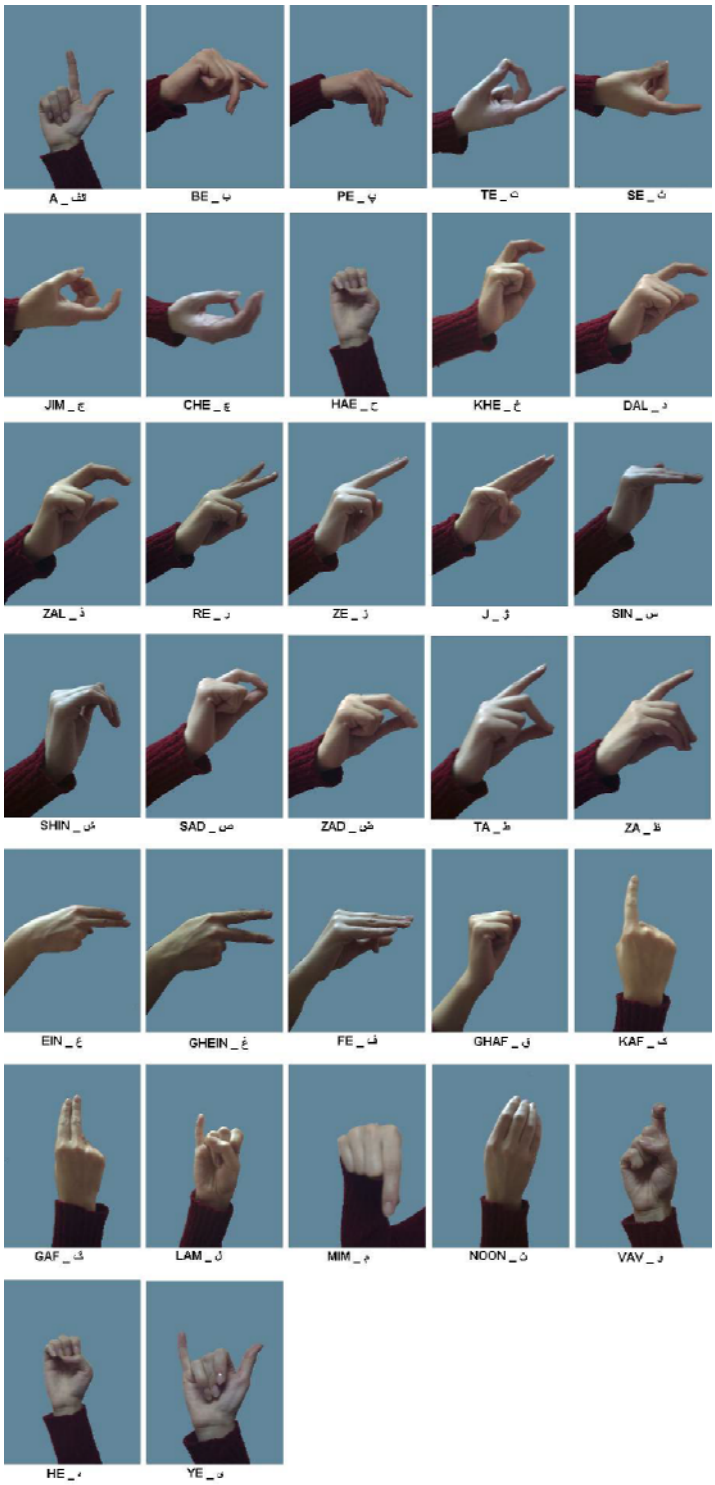
[8] C. Marsala and B. B-Meunier, “Choice of a method for the construction of fuzzy decision trees”, University P. e M. Curie, Paris, The IEEE International Conference on Fuzzy Systems, 2003

[9] J. R. Quinlan, “Simplifying decision trees”, Int’1 J. Man-Machine Studies, vol. 27, pp. 221-234, 1987

[10] T. Niblett and I. Bratko, “Learning decision rules in noisy domains”, Proc. Expert Systems 86, Cambridge, Cambridge University Press, 1986

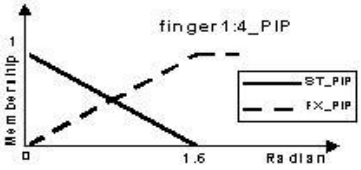
[11] E. J. Holden, R. Owens and Geoffrey G. Roy, “Adaptive fuzzy expert system for sign recognition”, 1999

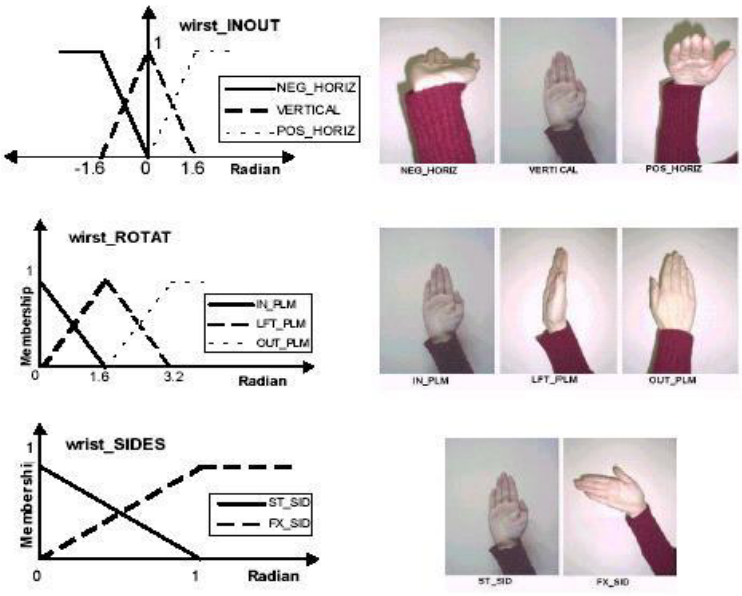
پیوست ۱ : اشارات الفبای فارسی



پیوست ۲ : متغیرهای فازی، ترمهای زبانی و توابع عضویت پیشنهادی برای اشارات الفبای فارسی

زاویه	متغیر فازی	ترم های فازی	شرح ترم فازی
α_3	**Finger1_PIP**	ST_PIP_F1 FX_PIP_F1	Straight Flexed
β_3	**Finger2_PIP**	ST_PIP_F2 FX_PIP_F2	Straight Flexed
γ_3	**Finger3_PIP**	ST_PIP_F3 FX_PIP_F3	Straight Flexed




wirst_INOUT
1
NEG_HORIZ
VERTICAL
POS_HORIZ
-1.6
0
1.6
Radian
NEG_HORIZ
VERTICAL
POS_HORIZ
wirst_ROTAT
Membership
1
IN_PLM
LFT_PLM
OUT_PLM
0
1.6
3.2
Radian
IN_PLM
LFT_PLM
OUT_PLM
wrist_SIDES
1
ST_SID
FX_SID
0
1
Radian
ST_SID
FX_SID