مقاله پژوهشی

مجله ماشین‌بینایی و پردازش‌تصویر
سال نهم، شماره اول، بهار ۱۴۰۱

ارائه روشی مبتنی بر رای‌گیری برای ترکیب خروجی‌های شبکه‌های عمیق جهت آنالیز قالب‌بندی اسناد چاپی

امیررضا فاتح[۱]، محسن رضوانی[۲]، علیرضا تجری[۳]، منصور فاتح[۴]

چکیده

در چند دهه گذشته، تحقیقات فراوانی در زمینه OCR یا نویسه‌خوان نوری انجام شده است. نویسه‌خوان نوری، یکی از راه‌های تبدیل تصاویر متنی به متن قابل ویرایش و شناسایی حروف و کلمات به صورت خودکار است. تشخیص مناطق متنی و غیرمتنی درون سند به آنالیز قالب‌بندی اسناد شناخته می‌شود و یکی از گام‌های کلیدی در روند تبدیل تصویر سند به متن قابل ویرایش است. جداسازی مناطق متنی و غیرمتنی درون یک تصویر از تاثیرگذارترین پیش‌پردازش‌های ممکن در سیستم‌های نویسه‌خوان نوری است. نبودن یک قالب یکسان در تمامی صفحات، وجود پس‌زمینه‌های پیچیده، نویزهای مختلف، کیفیت پایین، چرخش تصاویر و تصاویر چندین ستونه مانع از شناسایی درست مناطق حاوی متن می‌شوند. عدم تشخیص درست مناطق حاوی متن و به‌تبع آن عدم تشخیص صحیح مختصات خطوط، تمامی بخش‌های بعدی یک سیستم نویسه‌خوان نوری را دچار اخلال می‌کند. در این تحقیق، روشی نوین برای تشخیص مناطق متنی درون تصویر ارائه شده است. روش پیشنهادی، با بکارگیری از چندین روش مختلف و استفاده از سیستم رای‌گیری در میان آن‌ها، مناطق متنی تصویر را استخراج می‌نماید که تا کنون در کارهای پیشین از آن بهره گرفته نشده است. روش پیشنهادی بر روی دادگانی از تصاویر با بیش از ۹۵۰ صفحه مورد آموزش و آزمون قرار گرفته است که نتایج آزمون حاکی از ارائه دقت ۹۷,۹۴% در روش پیشنهادی است. مجموعه دادگان ارائه شده در این مقاله به صورت آزاد در دسترس است.

کلیدواژه‌ها

تقسیم‌بندی تصویر، آنالیز قالب‌بندی سند، آشکارسازی متن، آشکارسازی تصویر، رای‌گیری

۱ مقدمه

بسیاری از سازمان‌ها در تلاش‌اند تا اسناد درون بخش بایگانی خود را به صورت دیجیتال ذخیره نمایند تا امکان جستجو در این فایل‌ها فراهم شود. برای این منظور نیاز است که تمامی این اسناد توسط یک یا چند نفر در رایانه بازنویسی و در پایگاه داده‌ی آن سازمان ذخیره شوند. راه آسان‌تر، استفاده از سیستم‌های صفحه‌خوان[۱] در حوزه هوش مصنوعی است. در این سیستم‌ها ابتدا از صفحات یک سند عکس گرفته می‌شود و سپس با دادن تصاویر گرفته شده به این سیستم‌ها، متن درون آن‌ها به متن قابل ویرایش تبدیل می‌شود.

این مقاله در فروردین ماه سال ۱۴۰۰ دریافت، در تیرماه همان سال بازنگری و سپس پذیرفته شد.

[۱] کارشناس ارشد هوش مصنوعی، دانش‌آموخته دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شاهرود
رایانامه: amirreza.fateh@shahroodut.ac.ir
[۲] دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شاهرود
رایانامه: mrezvani@shahroodut.ac.ir
[۳] دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شاهرود
رایانامه: tajary@shahroodut.ac.ir
[۴] دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شاهرود
رایانامه: mansoor_fateh@shahroodut.ac.ir

مولف مسئول: منصور فاتح
DOR: 20.1001.1.23831197.1401.9.1.4.5

[1] Page Reader

در واقع، بسیاری از اسناد و کتاب‌ها، در فرمت‌های تصویری مانند BMP، JPG، PNG، TIFF و غیره در دسترس هستند که امکان جستجو و ویرایش کلمات در آن‌ها وجود ندارد. با استفاده از سیستم‌های صفحه خوان، این مشکل مرتفع شده و امکان جستجو در اسناد فراهم می‌شود [۱].

در یک سیستم صفحه‌خوان، تصویر یک سند به عنوان ورودی گرفته می‌شود که این تصویر می‌تواند توسط اسکنر، دوربین یا نرم‌افزارهای خاص، تولید شده باشد. این سیستم‌ها از طریق یک فرآیند دو مرحله‌ای شامل " آنالیز قالب‌بندی سند (DLA)[1]" و "نویسه‌خوان نوری (OCR)[2]"، هر تصویر از سند را به یک فایل متنی، تبدیل می‌کند [۲, ۳]. در واقع، سیستم‌های صفحه‌خوان از دو مرحله‌ی اصلی تشکیل شده‌اند. در مرحله‌ی اول، جداسازی و استخراج مناطق متنی و غیرمتنی درون تصویر انجام می‌شود که به آن آنالیز قالب‌بندی سند گفته می‌شود [۴].

در مرحله‌ی دوم، ناحیه‌ی متنی تصویر ورودی، به سیستم نویسه‌خوان داده شده و به متن قابل ویرایش تبدیل می‌شود. تنها ناحیه‌ی متنی سند به سیستم نویسه‌خوان داده می‌شود، لذا می‌بایست نواحی متنی از تصویر اصلی جدا شوند. عدم جداسازی متن از تصویر و عدم تشخیص صحیح خطوط و پارگراف‌ها، دقت سیستم نویسه‌خوان را کاهش می‌دهد. به همین جهت، DLA یک مرحله بسیار مهم در سیستم‌های صفحه‌خوان است [۵].

یکی از مشکلات اساسی در زمینه سیستم‌های DLA، نبودن یک قالب یکسان در تمامی صفحات است. بدین معنا که ممکن است در یک صفحه پیچیدگی خاصی از نظر شکل، جدول یا گراف وجود نداشته باشد ولی در صفحه دیگر وجود داشته باشد. بعلاوه وجود پس‌زمینه‌های پیچیده، نویزهای مختلف، کیفیت پایین و چرخش تصاویر در تشخیص و استخراج مناطق حاوی متن موثر است [۵]. از دیگر چالش‌هایی که در این زمینه با آن روبرو هستیم، تصاویر چندین ستونه است که متن و تصویر در مجاورت یکدیگر در دو ستون مجزا قرار گرفته‌اند که باعث افزایش خطای سیستم‌های آنالیز قالب‌بندی می‌شوند [۶].

یکی از نیازهای مهم و اساسی در طراحی یک الگوریتم DLA، دقت بالا همراه با محاسبات کم است [۷]. اما دلایلی مثل کج گذاشتن صفحات در اسکنر، وجود انحنا در اسکن صفحات کتاب، تاری تصاویر، وضوح پایین تصاویر حاصل از دوربین و نور فلاش دوربین، مانع دستیابی الگوریتم به دقت‌های بالا می‌شوند [۸, ۹].

در این تحقیق سعی بر آن است که برخی از چالش‌های گفته شده مرتفع و روش نوینی در ارتباط با سیستم‌های DLA ارائه شود. در این تحقیق بر روی تصاویر اسکن شده تمرکز شده است و سعی در حل چالش‌های مربوط به تصاویر اسکن شده، شده است. به همین منظور، با بکارگیری از چندین الگوریتم مختلف آنالیز قالب‌بندی مرتبط با تصاویر اسکن شده و استفاده از سیستم رای‌گیری بین آن‌ها، محتمل‌ترین مناطق متنی و غیرمتنی با کمترین چالش، استخراج می‌شوند. این روش بر روی تصاویر سه کاناله و تک کاناله قابل اجراست.

نوآوری اصلی در این مقاله، ارائه یک سیستم DLA با سرعت و دقت خوب است. همچنین استفاده از چندین الگوریتم مختلف و بکارگیری از یک سیستم رای‌گیری جدید و مبتنی بر پنجره در بین آن‌ها و استخراج محتمل‌ترین مناطق متنی از دیگر نوآوری‌های این تحقیق است. الگوریتم‌های استفاده شده به صورت همزمان اجرا می‌شوند که باعث افزایش سرعت سیستم می‌شوند. با وجود موانعی همچون تنوع زیاد در قالب اسناد که باعث کاهش دقت سیستم‌های DLA می‌شوند، به‌دلیل بکارگیری از چندین الگوریتم متفاوت در روش پیشنهادی، دقت نهایی سیستم افزایش می‌یابد. روش پیشنهادی به نحوی طراحی شده است که با کاهش سرعت، امکان استخراج دقیق‌تر متون میسر می‌شود. نکته حائز اهمیت امکان بروزرسانی آسان سیستم در چارچوب تعریف شده است و به‌راحتی می‌توان الگوریتم‌های شرکت‌کننده در سیستم رای‌گیری را افزایش داد. استفاده از سیستم رای‌گیری پنجره‌ای از دیگر نوآوری‌های این تحقیق است که در زمان نزدیکی آرا ارائه می‌شود. همچنین برای افزایش بهره‌وری این سیستم، از یک مدل اضافه‌تر نیز استفاده شده است تا در صورت نزدیک بودن آرا و عدم کارایی سیستم راگیری پنجره‌ای، بتوان از این مدل نیز کمک گرفت.

در ادامه این مقاله و در بخش دوم شرح مختصری از الگوریتم‌های ارائه شده در زمیته تشخیص مناطق متنی و غیرمتنی را ارائه خواهیم کرد. روش پیشنهادی در بخش سوم توضیح داده خواهد شد. به ارائه ارزیابی عملکرد الگوریتم پیشنهادی در بخش چهارم خواهیم پرداخت. در نهایت نتیجه‌گیری را در بخش پنجم ارائه خواهیم کرد.

۲ کارهای پیشین

با پیشرفت علم و تکنولوژی، کارایی سیستم‌های هوش‌مصنوعی بیشتر شده است و به دلیل استفاده روزافزون از این سیستم‌ها، نیازمندی‌های این سیستم‌ها نیز در حال افزایش است. لذا سیستم‌های هوش مصنوعی، باید به‌طور مداوم بهبود یابند تا قادر به رفع نیاز کاربران باشند. سیستم‌های DLA و به‌تبع آن، سیستم‌های OCR نیز از این قاعده مستثنی نیستند. به طور کلی الگوریتم‌های مرتبط با قطعه‌بندی سند[3]، مرتبط با این پژوهش هستند. در این بخش ابتدا تاکیدی بر مبانی قطعه‌بندی و انواع آن خواهیم داشت و در ادامه، برخی از پژوهش‌های انجام شده در حوزه DLA را بررسی خواهیم کرد.

[1] Document Layout Analysis
[2] Optical Character Recognition
[3] Segmentation

۲-۱ قطعه‌بندی

به طور کلی به فرآیندی که در آن یک تصویر به چندین بخش یا منطقه تقسیم می‌شود، قطعه‌بندی تصویر گفته می‌شود. این مناطق که مجموعه‌ای از پیکسل‌های تصویر را به خود اختصاص می‌دهند، به عنوان اشیاء درون تصویر نیز تعریف می‌شوند [۱۰]. قطعه‌بندی تصویر معمولا برای تعیین مکان اشیاء و مرزها درون یک تصویر بکار می‌رود. به طور تخصصی‌تر، قطعه‌بندی تصویر به فرآیندی گفته می‌شود که در آن به هر یک از پیکسل‌های تصویر یک برچسب[1] اختصاص می‌یابد. در قطعه‌بندی، پیکسل‌هایی با یک برچسب در یک یا چند ویژگی خاص مشابه هستند. این ویژگی‌ها با توجه به ماهیت اطلاعاتی یک تصویر، متفاوت هستند [۱۱, ۱۲]. رویکردهای متفاوتی در زمینه قطعه‌بندی تصاویر وجود دارد که از مهم‌ترین آن‌ها می‌توان به روش‌های مبتنی بر لبه‌های تصویر[2]، روش‌های مبتنی بر ناحیه تصویر[3]، روش‌های بر پایه آستانه‌گذاری[4] روش‌های مبتنی بر خوشه‌بندی[5] اشاره کرد [۱۳-۱۶]. در ادامه به توضیح هر یک از این روش‌های خواهیم پرداخت.

۲-۲ قطعه‌بندی بر اساس لبه‌های تصویر

یکی از رایج‌ترین عملیات‌ها در زمینه پردازش تصویر، تشخیص و موقعیت لبه‌ها درون یک تصویر می‌باشد. اگر تشخیص لبه‌ها درون یک تصویر به‌درستی انجام پذیرد، ادامه فرآیند قطعه‌بندی تصویر راحت‌تر و با دقت بالاتری انجام می‌شود. فرآیند تشخیص لبه‌ها درون یک تصویر، یکی از پرکاربردترین علمیات‌ها در حوزه پردازش تصویر است. از تشخیص لبه‌ها در گام‌های میانی یک عملیات پردازش تصویر استفاده می‌شود و با استفاده از آن ویژگی‌های مختلفی از تصویر استخراج می‌شود. سیستم‌های DLA [۱۷]، آنالیز تصاویر پزشکی [۱۸]، آنالیز و استخراج ویژگی از تصاویر ماهواره‌ای [۱۹] و تشخیص خطوط درون جاده [۲۰] از جمله مهم‌ترین کاربردهای فرآیند تشخیص لبه بشمار می‌روند.

یکی از مهم‌ترین کاربردهای تشخیص لبه‌ها درون تصویر، در سیستم‌های DLA است. این فرآیند بویژه در مرحله تشخیص خطوط حاوی متن، نقش بسزایی را ایفا می‌کند. روش‌های گوناگونی برای انجام فرآیند تشخیص لبه‌ها درون تصویر وجود دارد. از جمله این روش‌ها لبه‌یاب پریویت[6]، لبه‌یاب سوبل[7] و لبه‌یاب کنی[8] است [۲۱-۲۳].

۲-۳ قطعه‌بندی بر اساس ناحیه تصویر

در این دسته از روش‌های قطعه‌بندی، نواحی به طور مستقیم مشخص داده می‌شوند. در پیکسل‌هایی که موقعیت هر یک از اشیاء درون تصویر را نشان می‌دهند، خصوصیات مشترکی وجود دارد که این خصوصیات مشترک در بین پیکسل‌های یک ناحیه، با پیسکل‌های نواحی دیگر متفاوت است [۲۴].

۲-۴ قطعه‌بندی بر اساس آستانه گذاری

در این دسته، ابتدا تصاویر سه کاناله به یک کانال خاکستری تبدیل می‌شود. در یک تصویر خاکستری، به هر پیکسل عددی بین ۰ تا ۲۵۵ اختصاص می‌یابد. سپس با مقایسه مقدار عددی هر پیکسل با حد آستانه، قطعه‌بندی تصویر انجام می‌پذیرد. این نوع قطعه‌بندی برای تصاویری کاربرد دارد که سطوح خاکستری آن دارای تغییرات زیادی نیست. اگر نیاز باشد که یک تصویر خاکستری به دودویی تبدیل شود، این عملیات پس از تعیین یک حد آستانه انجام می‌پذیرد. بدین صورت که به تمامی پیکسل‌هایی که مقدار عددی آن‌ها در تصویر خاکستری کمتر از حد آستانه باشد، مقدار صفر و به مابقی پیکسل‌ها مقدار یک(یا ۲۵۵) اختصاص می‌یابد [۲۴].

۲-۵ قطعه‌بندی بر اساس خوشه‌بندی

به طور کلی خوشه‌بندی به فرآیندی گفته می‌شود که در آن داده‌هایی که ویژگی‌های آن‌ها با یکدیگر بیشترین شباهت را دارند در یک دسته قرار می‌گیرند. بدین صورت یک مجموعه داده بر اساس شباهت‌های داده‌های درون خوشه و تفاوت‌ها آن‌ها با داده‌های دیگر خوشه‌ها، به چندین خوشه تقسیم می‌شود. تصویر نیز از این قاعده مستثنی نیست. در تصویر می‌توان هر داده را به یک پیکسل یا یک شی درون آن تصویر نسبت داد و ویژگی آن‌ها را می‌توان رنگ، شکل و بافت پیکسل‌ها در نظر گرفت [۱۵, ۲۵].

۲-۶ کارهای انجام شده در زمینه DLA

در مرجع [۲۶]، روشی برای تشخیص مناطق حاوی متن و اشکال ارائه شده است. در این روش، با ارتقای عملکرد عملیات آستانه‌گذاری و استفاده از عملیات‌های مورفولوژیک، به تشخیص مناطق متنی و اشکال پرداخته است. اما این الگوریتم در تشخیص عناصر غیرمتنی مثل نمودار به خوبی عمل نمی‌کند. لذا در مرجع [۲۷]، با انجام اصلاحاتی در ساختار اصلی این الگوریتم، نظیر تکنیک‌های مبتنی بر لبه که نسبت به نویز مقاوم هستند، سعی شده تا عناصر غیرمتنی مانند نمودارها نیز تا حدودی استخراج شوند.

عملیات‌های مورفولوژیکی موجود در زمینه پردازش تصویر، از نظریه‌هایی با پیشتوانه ریاضی قوی برخوردار هستند [۲۶]. یکی از مهم‌ترین کاربردهای این نوع از عملیات‌ها، تحلیل و پردازش ساختارهای هندسی[9] است. با بکارگیری این نوع از عملیات‌ها،

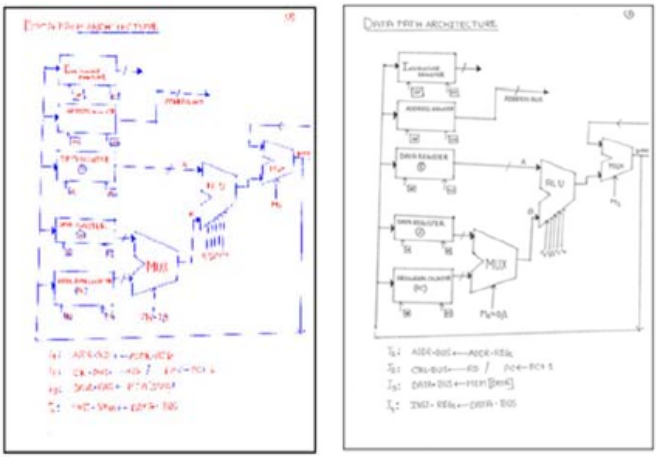
[1] Label
[2] Edge based segmentation
[3] Region based segmentation
[4] Thresholding based segmentation
[5] Clustering based segmentation
[6] Prewitt edge detector
[7] Sobel edge detector
[8] Canny edge detector
[9] Geometrical structure

یک الگوریتم، راحت‌تر قادر به شناسایی اشیای مورد نظر خود براساس ساختارها[1] و اشکال[2] آن‌ها است [۱۲].

برای استفاده از عملیات‌مورفولوژیکی، باید به ماهیت فرآیند پردازشی، دقت ویژه‌ای داشت. نظریه‌های قدرتمندی که در پشت روش‌های مورفولوژی وجود دارد، انعطاف‌پذیری این روش‌ها را بالا برده است که این انعطاف‌پذیری باعث افزایش کاربردهای این دسته از عملیات شده است. از جمله کاربردهایی که از عملیات موفولوژیکی استفاده می‌کنند می‌توان به شناسایی خطوط و شناسایی ساختارهای هندسی متفاوت در تصاویر مختلف اشاره نمود. پایه و اساس روش‌هایی که مبتنی بر مورفولوژی هستند، نظریه مجموعه‌ها است [۲۴]. عملگرهایی مانند عملگر سایش[3]، گسترش[4]، انتقال[5] و انعکاس[6] در عملیات مورفولوژی بکار می‌رود. می‌رود. بسیاری از کاربردهای موجود در الگوریتم‌های فرآیندهای پردازشی، توسط این عملگرها قابل انجام است.

یکی از عملیات‌های مورفولوژی، عملیات نازک‌سازی است [۲۸]. این عملیات معمولا در مرحله پیش‌پردازش یک فرآیند صورت می‌گیرد. یکی از ملزومات اساسی در آماده‌سازی ساختارهای موجود در تصویر، عملیات نازک‌سازی است. در این عملیات، اندازه ساختارهای موجود در یک تصویر کاهش پیدا می‌کند. در این کاهش اندازه، معمولا قسمت‌هایی از یک شکل یا ساختار با بیشترین شباهت به پس‌زمینه تصویر، حذف می‌شوند. این کاهش اندازه، به‌صورت لایه‌به‌لایه و از خارجی‌ترین پیکسل‌های متعلق به یک شکل صورت می‌پذیرد [۲۸]. برای انجام عملیات نازک‌سازی، روش‌های متنوعی ارائه شده است که الگوریتم‌های مبتنی بر تکرار[7]، الگوریتم‌های غیر تکراری[8] و الگوریتم‌هایی با رویکرد موازی[9] از جمله این روش‌ها هستند [۲۸].

در مرجع [۲۹]، برای تشخیص مناطق متنی و غیرمتنی، ابتدا با استفاده از الگوریتم‌های مبتنی بر الگوهای دودویی محلی (LBP)[10]، برای هر یک از CC[11]های درون تصویر یک بردار ویژگی ساخته می‌شود. برای ساخت این بردار ویژگی، روش‌های متعددی مانند LBP پایه[12]، LBP بهبود یافته (ILBP)[13]، LBP یکنواخت (ULBP)[14]، LBP ثابت نسبت به چرخش (RILBP)[15]، LBP مقاوم و یکنواخت (RULBP)[16] و LBP ثابت به چرخش و یکنواخت (RIULBP)[17] وجود دارد [۳۰-۳۳]. در مرجع [۲۹]، بیان شده است که بردار ویژگی استخراج شده توسط هر یک از الگوریتم‌های مبتنی بر LBP دارای ابعاد و طول زیادی است که علاوه بر پیچیدگی محاسباتی بالا به همراه محدودیت‌های زمانی و فضایی، باعث کاهش دقت در الگوریتم دسته‌بندی نیز می‌شود. لذا با استفاده از نوعی بازی مشارکتی اقدام به انتخاب ویژگی می‌شود. با این فرآیند محتمل‌ترین مناطق با بیشترین اطلاعات، استخراج می‌شوند و دسته‌بندی بسیار راحت‌تر انجام می‌شود. در شکل ۱ نمونه‌ای از نتایج این الگوریتم مشاهده می‌شود. ورودی این سیستم تک کاناله است و تصویر نهایی بر اساس دو رنگ قرمز و آبی که به‌ترتیب نمایشگر مناطق متنی و غیرمتنی هستند، نمایش داده شده است.



الف: تصویر اصلی ب: تصویر خروجی

شکل ۱: ورودی و نتیجه نهایی تشخیص مناطق متنی و غیرمتنی [۲۹]

در گذشته، رویکردها در زمینه آنالیز قالب‌بندی اسناد، به دو نوع پایین به بالا[18] و بالا به پایین[19] تقسیم می‌شدند [۵, ۳۴]. امروزه، می‌توان رویکردهای آنالیز قالب‌بندی را به دو دسته مبتنی بر یادگیری با نظارت و یادگیری بدون نظارت تقسیم کرد. در مرجع [۳۴]، الگوریتمی برای تشخیص مناطق متنی و غیرمتنی در تصاویر اسناد هندی با استفاده از رویکرد پایین به بالا ارائه شده است. ابتدا محدوده هر CC با کشیدن یک کادر محصورکننده[20] حول آن، مشخص می‌شود. سپس کادرهای نزدیک به هم در راستای افق و عمود به یکدیگر متصل می‌شوند. در نهایت الگوریتم با استفاده از خصوصیاتی نظیر مساحت مناطق حاصل، به تشخیص مناطق منتی و غیرمتنی می‌پردازد. همانطورکه در شکل ۲ مشاهده می‌شود، پس از اتصال کادرها به‌یکدیگر، مناطق متنی و غیرمتنی در تصویر اصلی مشخص شده‌اند.

[1] Structure
[2] Shape
[3] Erosion
[4] Dilation
[5] Transition
[6] Reflection
[7] Iterative Thinning Algorithms
[8] Non-Iterative Algorithms
[9] Fully Parallel Approaches
[10] Local Binary Pattern
[11] Connected Component
[12] Basic LBP
[13] Improved LBP
[14] Uniform LBP
[15] Rotation Invariant
[16] Robust and Uniform LBP
[17] Rotation Invariant and Uniform LBP
[18] Bottom-Up
[19] Top-Down
[20] Bounding Box

الف: تصویر اصلی ب: کشیدن کادر حول CCها

ج: استخراج مناطق غیرمتنی د: استخراج مناطق متنی

شکل ۲: مراحل تشخیص مناطق متنی و غیرمتنی [۳۴]

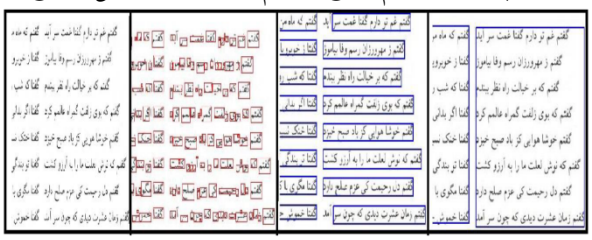
استفاده از درخت تصمیم یکی دیگر از راهکارهای تشخیص مناطق متنی و غیرمتنی درون تصویر است که در زمره روش‌های بالا به پایین قرار می‌گیرد. یکی از روش‌های تقسیم‌بندی تصویر مبتنی بر درخت تصمیم، برش (X Y)[1] نام دارد که در آن هر صفحه از یک سند در ریشه درخت قرار می‌گیرد و برگ‌ها مناطق نهایی تقسیم‌بندی شده هستند. در هر گره[2] این درخت ناحیه مورد نظر را به دو مستطیل کوچک‌تر تقسیم می‌کند و این عملیات تا آنجا ادامه می‌یابد که دیگر ناحیه‌ای را نتوان تقسیم نمود [۳۵].

در مرجع [۵] نیز از روش پایین به بالا برای تشخیص مناطق متنی و غیرمتنی استفاده شده است. مرجع [۵] که همه مراحل یک سیستم DLA را در خود جای داده است، برای بخش تشخیص مناطق متنی و غیرمتنی از یک شبکه عصبی کانولوشنی عمیق (DCNN)[3] آموزش دیده، بهره گرفته است. این نوع مدل‌ها که در حوزه یادگیری انتقال (TL)[4] قرار می‌گیرند، بسیار مفید هستند. یادگیری انتقال، به مدل‌هایی گفته می‌شود که با استفاده از یک دادگان آموزش دیده‌اند و برای عملیاتی دیگر با دادگانی متفاوت استفاده می‌شوند [۳۶]. در شکل ۳ نمونه‌ای از خروجی این مرحله در این مرجع [۵]، نشان داده شده است.

الف: تصویر اصلی ب: تصویر خروجی

شکل ۳: خروجی مدل شبکه عصبی کانولوشنی عمیق [۵]

در مرجع [۳۷] روشی نوین برای تقسیم تصاویر حاوی متن برای زبان فارسی به سه دسته متن، شکل و جدول ارائه شده است. این روش که جز روش‌های پایین به بالا قرار دارد، از تکنیک‌های آستانه گذاری وفقي، برچسب‌زنی مولفه‌ها، عملیات ریخت شناسی و تبدیل هاف استفاده شده و با یک الگوریتم مکاشفه‌ای و معرفی قوانین خاصی برای ترکیب نواحی کوچک بدون ادغام نواحی غیریکسان، سند را به ناحیه‌های متنی، جدول و شکل تقسیم می‌کند. در شکل ۴ مراحل تشخیص بخش متنی تصاویر آورده شده است که در آن از سمت چپ به ترتیب عبارتند از سند اولیه، مؤلفه‌های پیوسته، ادغام افقی، ادغام عمودی و تحلیل نهایی.



شکل ٤: استخراج بخش متنی تصویر

در مرجع [۳۸] رویکرد جدیدی برای استخراج متون فارسی از تصاویر رنگی به کمک موجک گسسته هار ارائه شده است. این موجک عملکرد نسبتا خوبی در تشخیص لبه‌های متنی دارد و با اعمال آن بر روی تصویر می‌توان مناطق متنی را استخراج نمود. در شکل شکل ۵ نمونه‌ای از تصاویر خروجی این روکرد نشان داده شده است.

شکل ۵: نمونه‌ای از تصاویر خروجی در مرجع [۳۸]

[1] X Y-Cut
[2] Node
[3] Deep Convolutional Neural Networks
[4] Transfer Learning

همان‌طور که گفته شد، امروزه رویکردها در زمینه آنالیز قالب‌بندی را می‌توان به دو دسته یادگیری بانظارت و یادگیری بدون نظارت تقسیم کرد. یادگیری عمیق و استفاده از انواع شبکه‌های عصبی در زمره یادگیری بانظارت قرار می‌گیرند. یکی از روش‌های نوین مبتنی بر یادگیری عمیق، روش (YOLO)[1] است [۳۹]. این روش که اولین بار در سال ۲۰۱۶ ارائه شد، جزء روش‌های بلادرنگ[2] محسوب می‌شود که در مقایسه با دیگر روش‌های تشخیص اشیا، عملکرد بهتری دارد. YOLOv3 [۴۰] یکی از روش‌های جدید مبتنی بر YOLO است که دارای ۵۳ لایه آموزش دیده بر روی ImageNet، به همراه ۵۳ لایه دیگر برای تشخیص اشیا است. بدلیل وجود ۱۰۶ لایه در YOLOv3، این ساختار به نسبت دیگر ساختارهای YOLO مثل YOLOv2 کمی کندتر است اما دقت بالاتر و عملکرد بهتری را ارائه می‌دهد. ساختار کلی این الگوریتم در شکل ۶ نشان داده شده است.

یکی از روش‌های مناسب در زمینه تشخیص مناطق متنی و غیرمتنی، (Faster R-CNN)[3] است [۴۱]. تفاوت عمده این روش با روش‌های پیشین همانند R-CNN و Fast R-CNN، در نحوه استخراج مناطق است. این دو روش با بکارگیری جستجوی انتخابی[4] مناطق مورد نظر را استخراج می‌کنند در حالی که در Faster R-CNN جستجوی انتخابی جای خود را به (RPN)[5] داده است [۴۱]. همان‌طور که در شکل ۷ نشان داده شده است، در این روش، تصویر ورودی به تعدادی لایه کانولوشنی داده می‌شود تا نقشه ویژگی[6] تصویر استخراج شود. سپس با بکارگیری RPN بر روی این نقشه ویژگی، مناطقی که وجود اشیا در آن‌ها محتمل‌تر است، بدست می‌آیند و با کادرهایی خاص مشخص می‌شوند. در مرحله بعد این مناطق به اندازه‌ی اصلی تصویر برگردانده شده و در نهایت کادرها، طبقه‌بندی می‌شوند.

یکی دیگر از روش‌های مبتنی بر یادگیری عمیق (SSD)[7] است [۴۲]. همان‌طور که در شکل ۸ نشان داده شده است، در این روش از شبکه کانولوشنی برای محاسبه نقشه ویژگی تصویر ورودی استفاده می‌شود. سپس با بکارگیری شبکه‌های کانولوشنی کوچک ۳ در ۳ بر روی نقشه ویژگی، مناطق متنی و غیرمتنی تشخیص داده شده و دور آن‌ها کادر رسم می‌شود.

از دیگر روش‌های تشخیص مناطق حاوی متن، استفاده از Layout Parser است [۴۳]. Layout Parser ابزاری مبتنی بر یادگیری عمیق برای کارهای تجزیه و تحلیل طرح اسناد است. این ابزار که از طریق GitHub قابل دسترسی است، ابزاری نسبتا قدرتمند است که توانایی استخراج مناطق متنی و غیرمتنی از درون تصاویر نسبتا پیچیده را دارد [۴۳].

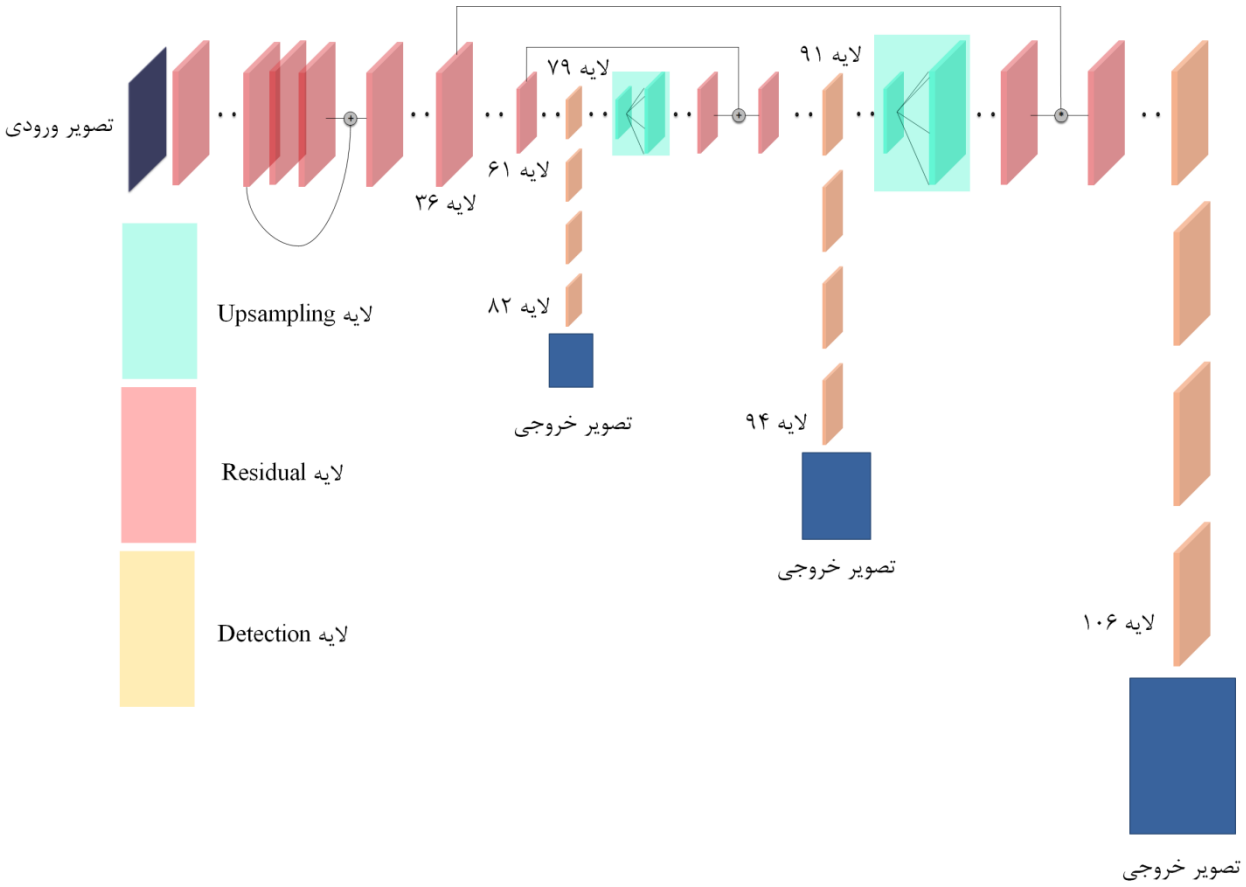
OCRopus یکی دیگر از رویکرد مبتنی بر شبکه عصبی عمیق است که اولین بار در سال ۲۰۰۸ معرفی شد. این روش پیش‌پردازش خوب و قابل توجهی را در زمینه OCR ارائه کرد [۴۴]. این روش تا به امروز چندین بار بروزرسانی شده است. آخرین بروزرسانی این روش در سال ۲۰۱۷ بوده است و شامل بخش‌های نظیر تشخیص مناطق متنی تصویر، استخراج خطوط و نویسه‌خوان نوری با استفاده از LSTM است.

یکی از روش‌های مناسب در زمینه تشخیص مناطق متنی و غیرمتنی، تبدیل هاف[8] است. با استفاده از تبدیل هاف که جزء دسته‌ی یادگیری بودن نظارت است می‌توان اشکال هندسی نظیر دایره، بیضی، مستطیل، خط و به‌طور کل هر شکل دیگر هندسی با رابطه‌ی مشخص را به‌راحتی تشخیص داد. به عنوان مثال، خط را می‌توان با استفاده از دو پارامتر شیب و عرض از مبدا تعریف کرد. لذا می‌توان با استفاده از تبدیل هاف آن را استخراج نمود [۲۴].

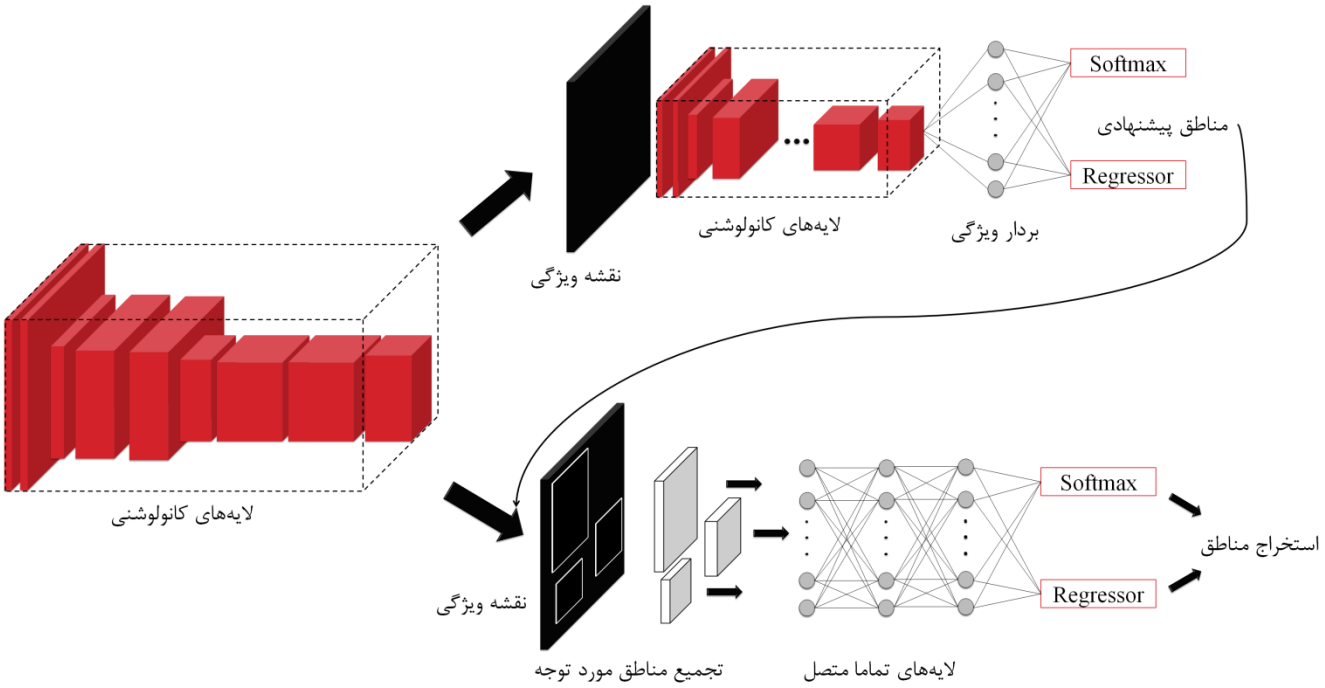
برای تشخیص اشیا درون تصویر با استفاده از تبدیل هاف، ابتدا می‌بایست لبه‌های درون تصویر استخراج شوند. در مرحله بعد، شناسایی اشیا با استفاده از این لبه‌ها انجام می‌پذیرد [۲۴]. در تبدیل هاف، فضای جدید به وسیله‌ی تعدادی از سلول‌ها که به انباشتگر[9] معروف‌اند، توصیف می‌شود. در یک تصویر دودویی، برای هر یک از مقادیر مربوط به لبه، یک عمل نگاشت یا همان فرآیند رای‌گیری به فضای انباشتگر انجام می‌شود. در پایان با در نظر گرفتن مختصات نقاط بیشینه در فضای انباشتگر (مختصات بیشترین رأی در فضای انباشتگر) و جایگذاری آن‌ها در معادله مدل مربوط به آن شی، شکل مورد نظر در تصویر شناسایی می‌شود [۲۴].

در مرجع [۴۵]، یک روش جدید برای تشخیص مناطق متنی و غیرمتنی از درون تصاویر صحنه‌های طبیعی[10] ارائه شده است. دستیابی به تشخیص درست با دقت بالا، منوط به انتخاب ویژگی‌ها و طبقه‌بندهای بهینه است. مرجع [۴۵]، توانسته با بهبود دادن الگوریتم (MSER)[11]، محتمل‌ترین مناطق متنی درون تصویر را تشخیص دهد. در مرحله بعدی، یازده ویژگی را از درون این مناطق استخراج می‌کند. این ویژگی‌ها شامل ویژگی‌های مرتبط با متن، مبتنی بر لبه، مبتنی بر رنگ و مبتنی بر شکل هندسی هستند. سپس با استفاده از پارامترهای BFS و CfsSubsetEval درون ابزار وکا[12] [۴۶]، از بین یازده ویژگی، مجموعه ویژگی بهینه را انتخاب می‌کند تا بتواند بخش متنی و غیرمتنی را از هم تفکیک کند. در نهایت با بکارگیری ابزار وکا بر روی بخش آموزش دادگان ICDAR 2013 [۴۷]، طبقه‌بندهای متعددی توسط این روش آموزش می‌بینند.

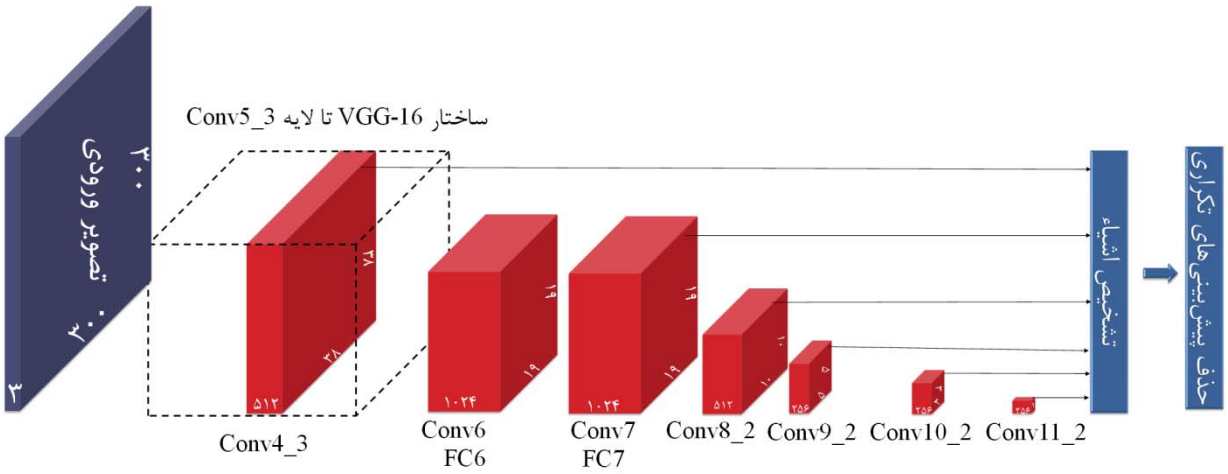
[1] You Only Look Once
[2] Real-time
[3] Faster Region-based Convolution Neural Network
[4] Selective Search
[5] Region Proposal Network
[6] Feature map
[7] Single Shot Detector
[8] Hough Transform
[9] Accumulator
[10] Natural scene images
[11] Maximally Stable Extremal Region
[12] Weka tool



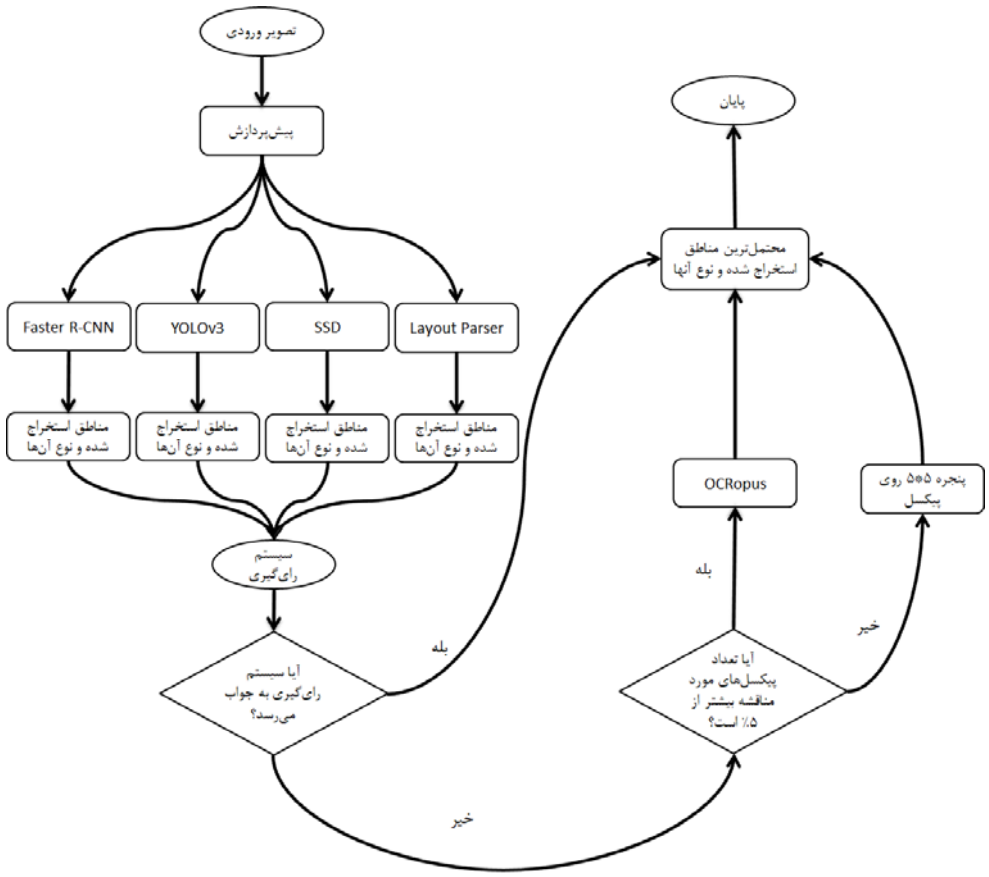
شکل ٦: ساختار کلی الگوریتم YOLOv3



شکل ۷: ساختار کلی الگوریتم Faster R-CNN



شکل ۸: ساختار کلی الگوریتم SSD



شکل ۹: مراحل روش پیشنهادی

۳ روش پیشنهادی

در این بخش به توصیف روش پیشنهادی به منظور استخراج مناطق متنی تصویر خواهیم پرداخت. در روش پیشنهادی، از چهار مدل از قبل آموزش دیده[1] Faster R-CNN، YOLOv3، SSD و Layout Parser استفاده شده است. این چهار مدل مبتنی بر یادگیری عمیق هستند که دقت و سرعت نسبتا بالایی نسبت به دیگر روش‌های این حوزه دارند. دیاگرام روش پیشنهادی در شکل ۵ نشان داده شده است. همان‌طور که در شکل ۹ نشان داده شده، تصاویر به عنوان ورودی به هر یک از این چهار روش داده می‌شود. خروجی این روش‌ها یک دیتافریم[2] است که اطلاعاتی نظیر مختصات کادر استخراج شده و نوع متنی یا غیرمتنی بودن منطقه را در خود جای داده است. پس از استخراج مناطق توسط این چهار روش، با بکارگیری یک روش رای گیری، محتمل‌ترین مناطق متنی و غیرمتنی بدست می‌آیند. همچنین در صورت برابر بودن آرا به سراغ مدل OCRopus یا استفاده از پنجره ۵*۵ می‌رویم. مدل OCRopus دقت مناسب و سرعت پایینی دارد و فقط در شرایط خاص که تعداد پیکسل‌های دارای مناقشه زیاد است، از آن استفاده

[1] Pre-trained models

[2] DataFrame

می‌شود. در ادامه به توضیح مراحل روش پیشنهادی خواهیم پرداخت.

۳-۱ پیش‌پردازش

همان‌طور که گفته شد، چهار مدل ابتدایی که در رای‌گیری دخیل هستند، با مجموعه‌ای از دادگان از قبل آموزش دیده‌اند. اما برای استفاده از آن‌ها نیاز است که پیش‌پردازش‌های لازم را بر روی آن‌ها انجام دهیم. همان‌گونه که در شبه کد شکل ۱۰ نشان داده شده است، برای پیاده‌سازی و استفاده از مدل Layout Parser ابتدا نیاز است که Detectron2 را بر روی سیستم خود یا google colab [۴۸] نصب نماییم. Detectron2 سیستم نرم‌افزاری است که توسط محققین هوش مصنوعی شرکت فیس‌بوک[1] ارائه شده است و پیشرفته‌ترین الگوریتم‌های تشخیص اشیاء را پیاده‌سازی کرده است [۴۹]. در مرحله بعدی، نیاز به نصب کتابخانه layoutparser است که با نصب آن قادر به تشخیص مناطق حاوی متن خواهیم بود. Parser ابزاری مبتنی بر یادگیری عمیق برای کارهای تجزیه و تحلیل طرح اسناد است. این ابزار که از طریق GitHub قابل دسترسی است، ابزاری نسبتا قدرتمند است که توانایی استخراج مناطق متنی و غیرمتنی از درون تصاویر حتی نسبتا پیچیده را نیز دارد [۴۳].

علاوه بر مدل Layout Parser، در این تحقیق از سه روش مبتنی بر یادگیری عمیق دیگر نیز بهره گرفته شده است. مدل‌های پیاده‌سازی شده توسط این سه روش که عبارت‌اند از Faster R-CNN، YOLOv3 و SSD، همانند مدل Layout Parser از قبل آموزش دیده شده‌اند. این مدل‌ها توسط شرکت Monk AI پیاده‌سازی و آموزش دیده شده‌اند و در دسترس عموم قرار گرفته‌اند [۵۰]. Monk که یک مجموعه ابزار بینایی رایانه‌ای است، بیش از ۲۰ مدل را برای تشخیص اشیا پیاده‌سازی کرده و با آموزش آن‌ها کار را برای دیگر محققین در این زمینه آسانتر کرده است. پس از انجام و نصب موارد گفته شده، مدل از قبل آموزش دیده شده مورد نظر را دریافت و همچنین آدرس کتابخانه‌هایش را به مسیر سیستم[2] اضافه می‌نماییم.

Algorithm 1: Pre-processing steps

```
Result: ready to run
Installing Detectron2;
Installing Monk;
Downloading pre-trained models;
Adding libraries path to the system path;
```

شکل ۱۰: مراحل پیش‌پردازش

۳-۲ ساخت، اجرا و آماده‌سازی سیستم جهت استفاده مناسب از مدل‌ها

زمانی که یک تصویر به هر یک از این چهار مدل داده می‌شود، این مدل‌ها تلاش می‌کنند که محتمل‌ترین مناطق متنی و غیرمتنی را استخراج نمایند. پس از استخراج این مناطق توسط هر چهار مدل، نیاز به انجام تغییرات بر روی نتایج بدست آمده است. از هر مدل، اطلاعات یکسانی را استخراج و درون یک دیتافریم ذخیره می‌شود. این دیتافریم شامل مختصات دو نقطه از کادر محصورکننده (مختصات بالاترین نقطه سمت چپ و پایین‌ترین نقطه سمت راست مشابه شکل ۱۱ کادر محصورکننده قرمز) و نوع منطقه است.

در نتایج بدست آمده احتمال همپوشانی کامل دو منطقه وجود دارد، علی الخصوص در مدل Layout Parser که این میزان همپوشانی بسیار بیشتر است. همپوشانی کامل دو منطقه استخراج شده بدین صورت است که منطقه‌ای که به عنوان متن در نظر گرفته شده است به طور کامل درون یک منطقه متنی دیگر باشد. یا یک منطقه غیرمتنی با یک منطقه متنی همپوشانی داشته باشد. برای هر تصویر خروجی از هر چهار مدل، دو ماتریس هم‌اندازه با تصویر اصلی ساخته می‌شود که مقادیر اولیه درایه‌های آن‌ها صفر است. این دو ماتریس نشان‌دهنده پیکسل‌هایی با برچسب متنی یا غیرمتنی هستند. به عنوان مثال اگر پیکسلی در یک منطقه متنی قرار داشته باشد، درایه متناظر آن در ماتریس متنی، برابر با عدد یک خواهد شد. همچنین اگر پیکسلی متعلق به بیش از یک منطقه باشد، همان مقدار یک را در درایه متناظر آن در ماتریس قرار خواهیم داد. یعنی اگر پیکسلی متعلق به دو یا چند منطقه متنی باشد، مقدار متناظر آن در ماتریس متنی یک خواهد بود. همچنین اگر پیکسلی متعلق به دو یا چند منطقه غیرمتنی باشد، مقدار متناظر آن در ماتریس غیرمتنی یک خواهد بود. اما اگر پیکسل متعلق به دو منطقه باشد که یکی متنی و یکی غیرمتنی باشد، درایه متناظر آن پیکسل در هر دو ماتریس متنی و غیرمتنی برابر با یک خواهد شد.

شکل ۱۱: کادرهای محصورکننده تشخیص داده شده بر روی تصویر

به عنوان نمونه، برای تصویر نشان داده شده در شکل ۱۲، روش پیشنهادی ابتدا دو ماتریس متنی و غیرمتنی هم‌اندازه با تصویر اصلی خواهد ساخت و تمامی مقادیر درایه‌های آن‌ها را صفر قرار می‌دهد. در مرحله بعد، نوع منطقه‌ی هر پیکسل را مشخص کرده و

[1] Facebook
[2] System path

درایه متناظر با آن پیکسل در ماتریس متنی یا غیرمتنی را برابر با یک قرار خواهد داد. به عنوان مثال، همانند شکل ۱۲ پیکسل (x_i,y_j) چون در هر دو منطقه متنی و غیرمتنی قرار دارد، درایه متناظرش در هر دو ماتریس برابر با یک خواهد شد. این دو ماتریس برای خروجی هر چهار مدل ساخته شده و مقادیر درایه‌های آن‌ها به همین نحوه قرار خواهند گرفت. در نهایت با ترکیب چهار ماتریس متنی و جمع مقادیر درایه‌های آن‌ها یک ماتریس نهایی برای مناطق متنی خواهیم داشت و به طور مشابه ماتریس متعلق به مناطق غیرمتنی هم ساخته خواهند شد.

تمام مراحل ساخت ماتریس‌های متناظر با مناطقی متنی و غیرمتنی در شبه کد شکل ۱۳ نشان داده شده است.

شکل ۱۲: چالش همپوشانی مناطق

Algorithm 2: Layout Analysis steps

Result: text and non-text matrices

Function `FasterR-CNN(`*Input − image*`)`**:**
- Input-image $\xrightarrow{\text{FasterR-CNN}}$ text-regions;
- text-regions $\xrightarrow{x1,y1,x2,y2,region-type}$ DataFrame;
- Text-matrix, Non-text-matrix = zero matrix with the same size of input-image → (h,w);
- **for** *each pixel of Input-image* **do**
  - **if** *the pixel is in one of the text-region of the DataFrame* **then**
    - Text-matrix [pixel's row, pixel's column] = 1
  - **end**
  - **if** *the pixel is in one of the non-text-region of the DataFrame* **then**
    - Non-text-matrix [pixel's row, pixel's column] = 1
  - **end**
- **end**
- **return** Text-matrix, Non-text-matrix;

Function `YOLOv3(`*Input − image*`)`**:**
- .. like Faster R-CNN ..;
- **return** Text-matrix, Non-text-matrix;

Function `SSD(`*Input − image*`)`**:**
- .. like Faster R-CNN ..;
- **return** Text-matrix, Non-text-matrix;

Function `LayoutParser(`*Input − image*`)`**:**
- .. like Faster R-CNN ..;
- **return** Text-matrix, Non-text-matrix;

Function `Main`**:**
- Input-image = from dataset read an image;
- F-Text-matrix, F-Non-text-matrix = FasterR-CNN(Input-image);
- Y-Text-matrix, Y-Non-text-matrix = YOLOv3(Input-image);
- S-Text-matrix, S-Non-text-matrix = SSD(Input-image);
- L-Text-matrix, L-Non-text-matrix = LayoutParser(Input-image);
- Text-matrix = Combining all four text-matrices;
- Non-text-matrix = Combining all four non-text-matrices;

شکل ۱۳: مراحل پیاده‌سازی، اجرا و پس‌پردازش مدل‌های آنالیز قالب‌بندی

۳-۳ سیستم رای‌گیری

پس از مشخص شدن مقادیر درایه‌های دو ماتریس متنی و غیرمتنی برای یک تصویر، به سراغ رای‌گیری می‌رویم. در این مرحله می‌بایست محتمل‌ترین مناطق متنی و غیرمتنی را به کمک دو ماتریس بخش قبل استخراج نماییم. مقادیر درون درایه‌های این دو ماتریس، حداقل صفر و حداکثر چهار هستند. به عنوان مثال اگر مقدار درایه‌ای مانند (i,j) در درون ماتریس غیرمتنی، برابر با سه و در درون ماتریس متنی دو باشد، یعنی سه روش پیکسل (xi,yj) در تصویر اصلی را غیرمتنی و دو روش آن را متنی تشخیص داده‌اند. تعداد روش‌ها برای رای‌گیری چهار است، اما گاهی مجموع مقادیر متنی و غیرمتنی از چهار بیشتر می‌شود. دلیل به وجود آمدن این اتفاق وجود همپوشانی مناطق غیر همنوع در یکی از روش‌ها است.

در سیستم رای‌گیری باید مشخص کنیم که هر پیکسل و به‌تبع آن هر درایه از دو ماتریس متنی و غیرمتنی که مقداری به جز صفر دارد، در کدام یک از دو نوع منطقه متنی و غیرمتنی قرار خواهد گرفت. در صورتی که اختلاف مقادیر درایه‌های دو ماتریس در یک پیکسل، بزرگتر یا مساوی یک باشد، سیستم رای‌گیری آن پیکسل را در زمره منطقه‌ای قرار خواهد داد که مقدار درایه ماتریسش بزرگتر باشد. به عنوان نمونه اگر مقدار پیکسل (x_i,y_j) در درایه متناظرش در ماتریس متنی برابر با سه و در ماتریس غیرمتنی برابر با یک باشد، این پیکسل به عنوان یکی از پیکسل‌های منطقه متنی قلمداد خواهد شد.

اما در غیر این صورت سیستم رای‌گیری به تنهایی قادر به تصمیم‌گیری نیست. یکی از دلایل این اتفاق ضعیف‌تر بودن مدل Layout Parser به نسبت سه مدل دیگر است. در این مدل علاوه بر همپوشانی زیاد مناطق، میزان دقت تشخیص درست مناطق نیز پایین‌تر است. لذا در صورتیکه تعداد پیکسل‌های مورد مناقشه بیش از ۵% پیکسل‌های تعیین تکلیف شده باشد، از مدلی دیگر کمک می‌گیریم. آستانه ۵% با سعی و خطا و بررسی سرعت و دقت حاصل شده است.

این مدل که OCRopus نام دارد، دقت بالایی در تشخیص مناطق حاوی متن دارد ولی همانطورکه در جدول ۱ نشان داده شده است، نسبت به چهار روش بکار گرفته شده، بسیار کندتر است. بعلاوه تصاویر ورودی این مدل حتما می‌بایست وضوح 300 dpi داشته باشند، در غیر این صورت امکان استفاده از این مدل وجود ندارد. به همین سبب در سیستم رای‌گیری از این مدل استفاده نشده است و تنها در صورت نیاز به سراغ این مدل خواهیم رفت. بدین صورت قادر خواهیم بود تا محتمل‌ترین مناطق متنی و غیرمتنی درون تصویر را تشخیص و جدا کنیم. نکته حائز اهمیت، اجرای همزمان چهار مدل اصلی سیستم رای‌گیری است. یعنی هر تصویر در کمتر از ۲,۵ ثانیه وارد سیستم رای‌گیری می‌شود که باعث افزایش چشمگیر سرعت روش‌پیشنهادی در مقایسه با مدل OCRopus است.

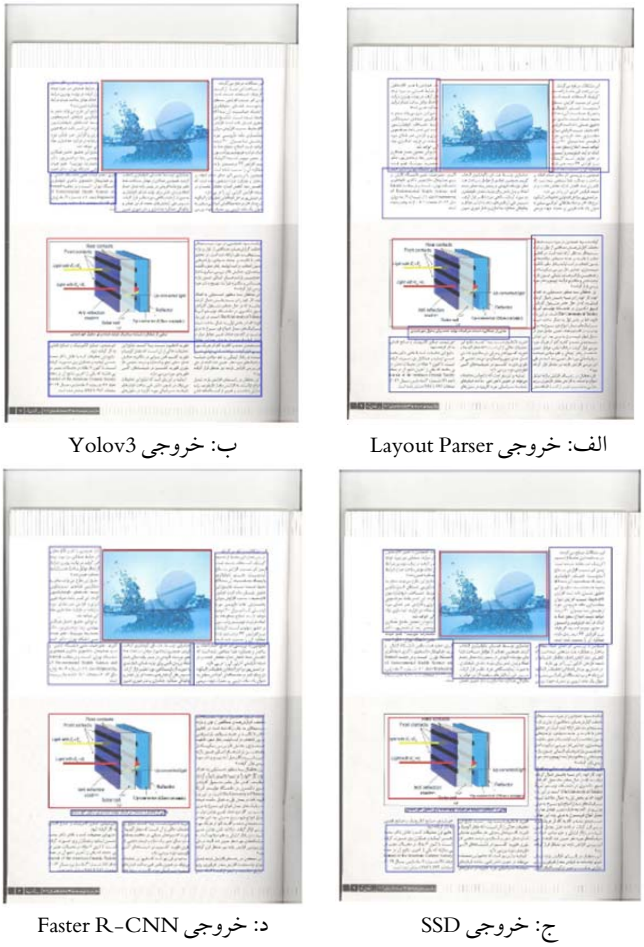
جدول ۱: میانگین زمان اجرا برای هر تصویر در پنج مدل استفاده شده روش پیشنهادی بر روی محیط Google colab

نام مدل	میانگین زمان اجرا برای هر تصویر (s)
Layout Parser	۱,۴۴
SSD	۰,۵۰۷
YOLOv3	۱,۳۹
Faster R-CNN	۲,۲۳۸
OCRopus	۱۸,۰۵۱

در صورتیکه تعداد پیکسل‌های مورد مناقشه کمتر از ۵% پیکسل‌های تعیین تکلیف شده باشد، از مدل OCRopus بهره گرفته نمی‌شود. در این موارد برای پیکسل‌های مورد مناقشه، پنجره‌ای ۵*۵ به مرکز پیکسل مورد نظر در نظر گرفته و نتیجه رای‌گیری روی این پنجره را مبنا قرار می‌دهیم. به عنوان نمونه شکل ۱۴ را در نظر بگیرید که در آن تعداد پیکسل‌های مورد مناقشه در کل تصویر بسیار اندک است، لذا نیازی به استفاده از OCRopus نیست. روش پیشنهادی، ابتدا همانند شکل ۱۵ خروجی هر یک از چهار روش را بدست می‌آورد. حال یکی از پیکسل‌های مورد مناقشه را در نظر می‌گیریم. سیستم پنجره‌ای ۵*۵ به مرکز پیکسل مورد نظر باز می‌کند و مقادیر درایه‌های متناظر پنجره را در دو ماتریس متنی و غیرمتنی، بدست می‌آورد. نتایج پیش‌بینی چهار روش، در این پنجره ۵*۵ در جدول ۲ و جدول ۳ آورده شده است. همان‌طور که مشاهده می‌شود، مجموع تعداد آرای غیرمتنی بیشتر است، لذا این پیکسل در زمره پیکسل‌های غیرمتنی قرار خواهد گرفت.

مراحل رای‌گیری و استفاده از سیستم پنجره‌ای ۵*۵ و OCRopus در شبه کد شکل ۱۶ نشان داده شده است.

شکل ۱۴: نمونه‌ای از یک پیکسل مورد مناقشه



الف: خروجی Layout Parser ب: خروجی Yolov3

ج: خروجی SSD د: خروجی Faster R-CNN

شکل ۱۵: خروجی چهار مدل ارائه شده بر روی یک نمونه تصویر

جدول ۲: جمع مقادیر درایه‌های متناظر پنجره ۵*۵ در چهار ماتریس غیرمتنی

Non-text sum=44	j-2	j-1	j	j+1	j+2
i-2	3	3	2	1	1
i-1	3	3	2	1	1
i	3	3	2	1	1
i+1	2	2	1	1	1
i+2	2	2	1	1	1

جدول ۳: جمع مقادیر درایه‌های متناظر پنجره ۵*۵ در چهار ماتریس متنی

Text sum=34	j-2	j-1	j	j+1	j+2
i-2	0	0	0	1	1
i-1	1	1	2	2	2
i	1	1	2	2	2
i+1	1	1	2	2	2
i+2	1	1	2	2	2

```
Algorithm 3: Voting
  TM:Text-matrix
  NTM:Non-text-matrix
  pr: pixel's row
  pc: pixel's column
  DP: Disputed pixels
  DP = []
  for each pixel of text-matrix do
      if TM [pr, pc] > NTM [pr, pc] then
          pixel considered as text-region
      end
      if TM [pr, pc] < NTM [pr, pc] then
          pixel considered as non-text-region
      end
      if TM [pr, pc] == NTM [pr, pc] then
          add pixel to DP
      end
  end
  if Number of DP's pixels is less than 5% of all tagged pixels then
      for each pixel of DP do
          if Sum(TM [pr-2:pr+2, pc-2:pc+ 2]) >= Sum(NTM [pr-2:pr+2,
           pc-2:pc+ 2]) then
              pixel considered as text-region
          else
              pixel considered as non-text-region
          end
      end
  else
      Using OCRopus
  end
```

شکل ۱۶: مراحل رای‌گیری

۴ ارزیابی عملکرد الگوریتم

در بخش قبل، روشی برای پیاده‌سازی سیستم‌های DLA ارائه شد که در آن مناطق حاوی متن از درون تصویر استخراج می‌شوند. در این بخش ابتدا عملکرد روش پیشنهادی را ارزیابی می‌کنیم و سپس به انجام آزمایش‌های مختلف بر روی مدل پیشنهادی خواهیم پرداخت. تصویر ورودی به سیستم DLA پیشنهادی، می‌تواند توسط اسکنر یا دوربین اخذ شود که ما از یک اسکنر برای تهیه تصاویر کمک گرفتیم تا مجموعه دادگان استانداردتری داشته باشیم. مجموعه تصاویر استفاده شده برای آزمایش مدل پیشنهادی از منابع مختلفی مانند مجله، روزنامه و کتاب‌های گوناگون جمع‌آوری شده‌اند. روش پیشنهادی با زبان برنامه‌نویسی پایتون در محیط Google colab شبیه‌سازی شده‌اند [۴۸]. Google colab یک محیط، ابزار رایگان و قدرتمند برای ساخت و اجرای مدل‌های مبتنی بر یادگیری ماشین است. این ابزار دارای سخت‌افزارهای قدرتمندی همچون GPU و TPU است که در اختیار کاربران خود قرار می‌دهد. از دو نسخه پایتون ۳,۷ و پایتون ۲,۷ برای پیاده‌سازی روش پیشنهادی و کارهای مرتبط بهره گرفتیم.

۴-۱ مجموعه دادگان

برای ارزیابی عملکرد روش پیشنهادی، نیاز به مجموعه دادگان مناسب است. اما ما با کمبود مجموعه دادگان مناسب برای استفاده در سیستم DLA روبرو هستیم. به همین دلیل ساخت یک مجموعه داده استاندارد انجام شد که در هر دو فاز آزمون و یادگیری قابل استفاده است. این مجموعه داده ۹۵۲ تصویر دارد. ۸۰۰ تصویر برای آزمون و ۱۵۲ تصویر برای استخراج مقادیر بهینه بعضی از پارامترهای مدل‌ها، تعیین اندازه مناسب برای روش پنجره و دیگر موارد بکار برده شده است. این تصاویر توسط یک اسکنر مدل HP Scanjet 4890 با تنظیمات پیش‌فرض تهیه شده است. درجه تفکیک در برخی منابع 200 dpi و در برخی دیگر از منابع 300 dpi است که نمونه‌هایی از تصاویر پایگاه داده با درجه تفکیک 300 dpi در شکل ۱۱ و شکل ۱۲ و با درجه تفکیک 200 dpi در شکل ۱۷ مشاهده می‌شود. این مجموعه دادگان به صورت آزاد در دسترس خواهد بود تا به پیشرفت علم در این زمینه افزوده شود. مجموعه دادگان ارائه شده این مقاله در مرجع [۵۱] قرار داده شده است.

شکل ۱۷: نمونه‌ای از تصویر با درجه تفکیک 200 dpi

همان‌طور که در تصویر شکل ۱۷ مشاهده می‌شود، تصاویر این مجموعه داده از پیچیدگی نسبتا زیادی برخوردار هستند. یکی از پیچیدگی‌های اصلی این مجموعه داده کج بودن و چرخش در تصاویر است که پیچیدگی زیادی را ایجاد می‌کند. در این مجموعه داده انواع مختلف تصاویر با پیچیدگی‌های متنوع وجود دارد. تصاویر این مجموعه داده از منابع مختلفی همچون روزنامه‌ها، مجلات و کتاب‌ها تهیه شده‌اند.

برای ارزیابی بهتر روش‌های ارائه شده در این مقاله از یک مجموعه دادگان دیگر به‌نام PRImA نیز بهره گرفتیم [۵۲]. تصاویر این مجموعه دادگان از مجلات و مقالات تهیه شده‌اند. همچنین تعداد تصاویر این مجموعه دادگان ۴۷۸ تصویر است. تصاویر این مجموعه داده، حاوی اشکال و متون متنوعی هستند. بدلیل عدم وجود انحنا و کجی در تصاویر، این مجموعه دادگان از پیچیدگی کمتری نسبت به مجموعه دادگان پیشنهادی در این مقاله برخوردارند. روش‌های مختلفی همچون DICE، Fraunhofer، REGIM-ENIS، Tesseract، FineReader با استفاده از این مجموعه دادگان آزموده شده‌اند که نتایج آنها در ICDAR2009 Page Segmentation Competition قرار داده شده است [۵۳].

۴-۲ ارزیابی عملکرد روش پیشنهادی

برای ارزیابی عملکرد الگوریتم، ما علاوه بر آزمون روش پیشنهادی، پنج مدل از قبل آموزش دیده Faster R-CNN، YOLOv3، SSD، Parser Layout و OCRopus را نیز بر روی

دادگان خود آزمودیم. روش‌های مختلفی برای آنالیز قالب‌بندی در زبان‌های گوناگون ارائه شده است. این روش‌ها سابقا تا حدی وابسته به زبان بوده‌اند. اما روش‌های نوین و مخصوصا روش‌های مبتنی بر یادگیری عمیق، تا حد زیادی مستقل از زبان می‌باشند. شبکه‌های استفاده شده در این مقاله بر روی مجموعه دادگان مختلفی همچون PubLayNet [۵۴]، HJDataset [۵۵]، TableBank [۵۶]، Newspaper [۵۷] و PRImA [۵۲] آموزش دیده‌اند که به جز HJDataset بقیه مجموعه دادگان به زبان انگلیسی می‌باشند. همچنین شبکه‌های از قبل آموزش دیده شده برای آنالیز قالب‌بندی متون فارسی دقت مناسبی داشته‌اند و با توجه به عدم وجود دادگانی بزرگ و متنوع از متون فارسی برچسب خورده در حیطه آنالیز قالب‌بندی متون فارسی، آموزش مجدد بر روی مجموعه دادگان زبان فارسی انجام نشد. در واقع، دقت مدل‌های از قبل آموزش دیده شده به حدی است که در عمل آموزش مجدد آنها حائز اهمیت نمی‌باشد.

همان‌طور که در جدول ۴ نشان داده شده است، روش تلفیقی بکار برده شده، بهترین جواب را گرفته است. دلیل پایین بودن دقت مدل Layout Parser، همپوشانی نسبتا زیاد مناطق غیر همنوع است که باعث کاهش دقت مدل Layout Parser در استخراج مناطق متنی شده است.

برای محاسبه دقت هر روش، از رابطه ۱ بهره گرفته شده است. در این رابطه، تعداد پیکسل‌های متنی درست تشخیص داده شده با $T_{correct}$ و تعداد کل پیکسل‌های متنی با T_{total} نمایش داده شده است. همچنین زمان اجرای روش پیشنهادی بسته به تعیین تکلیف شدن تصویر در بخش رای‌گیری یا استفاده از OCRopus و روش پنجره متفاوت است.

$$Accuracy_{text} = \frac{T_{correct}}{T_{total}} \quad (۱)$$

جدول ٤: دقت و سرعت تشخیص صحیح مناطق متنی توسط روش پیشنهادی و مدل‌های دیگر

نام مدل	دقت	میانگین زمان اجرا برای هر تصویر (s)
Layout Parser	%۷۸,۳۱	۱,۴۴
SSD	%۸۷,۵۱	۰,۵۰۷
YOLOv3	%۹۳,۳۳	۱,۳۹
Faster R-CNN	%۹۵,۶۷	۲,۲۳۸
OCRopus	%۹۷,۵۶	۱۸,۰۵۱
روش پیشنهادی تلفیقی	**%۹۷,۹۴**	۳,۰۲۶

پس از اتمام رای‌گیری در میان چهار روش، ۲۰۸ تصویر از ۸۰۰ تصویر آزمون دارای پیکسل‌های مورد مناقشه هستند که باید درباره آن‌ها جداگانه تصمیم‌گیری شود. در نتیجه ۵۹۲ تصویر، پیکسل‌های مورد مناقشه ندارند که تکلیف آن‌ها مشخص است. زمان اجرا و دقت روش پیشنهادی برای این ۵۹۲ تصویر در جدول ۶ آمده است که دقت و سرعت مناسبی است.

جدول ٥: دقت و سرعت تشخیص صحیح مناطق متنی توسط روش پیشنهادی برای ۵۹۲ تصویر بدون مناقشه از دادگان

نام مدل	دقت	میانگین زمان اجرا برای هر تصویر (s)
روش پیشنهادی تلفیقی	۹۸,۰۱	۲,۲۴۱

از میان ۲۰۸ تصویر باقیمانده، ۱۹۹ تصویر دارای پیکسل‌های مورد مناقشه کمتر از ۵% کل تصویر هستند و تنها ۹ تصویر دارای پیکسل‌های مورد مناقشه بیش از ۵% کل تصویر هستند. در روش پیشنهادی برای این ۱۹۹ تصویر از پنجره ۵*۵ برای رای‌گیری بهره گرفته شده است و برای ۹ تصویر باقیمانده از OCRopus کمک گرفته شده است.

برای ۱۹۹ تصویر، در جدول ۶ دقت و سرعت مدل OCRopus در مقایسه با روش پیشنهادی آورده شده است. همان‌طور که مشاهده می‌شود، زمانی که تنها از مدل OCRopus برای تشخیص بهره گرفتیم، با وجود کند بودن این مدل نسبت به مدل پیشنهادی، دقت آن نیز در کل کمی پایین‌تر است. پس عدم استفاده از مدل OCRopus تنها سرعت را افزایش داده است. اما با بررسی‌های انجام گرفته روی ۹ تصویر باقیمانده، یعنی تصاویری که تعداد پیکسل‌های مورد مناقشه زیاد دارند (بیشتر از ۵% پیکسل‌های تعیین تکلیف شده)، در می‌یابیم که مدل OCRopus کارا است. لازم به ذکر است که ۹ تصویر، حدود ۱% تعداد کل تصاویر مجموعه دادگان است. لازم به ذکر است که استفاده از تکنین پنجره برای رای‌گیری نیز روش پیشنهادی را کمی کند می‌کند. استفاده از تکنیک رای‌گیری پنجره‌ای حدود ۲,۳۴ ثانیه زمان روش پیشنهادی را افزایش می‌دهد. به همین دلیل برای همه تصاویر از این روش بهره گرفته نشد و تنها برای تصاویری باپیکسل‌های مورد مناقشه، از تکنیک رای‌گیری پنجره‌ای بهره گرفته شد.

جدول ٦: مقایسه دقت و سرعت مدل OCRopus و پنجره ۵*۵ بر روی ۱۹۹ تصویر کم مناقشه

نام مدل	دقت	میانگین زمان اجرا برای هر تصویر (s)
پنجره ۵*۵ پس از رای‌گیری	۹۷,۷۱	۴,۵۸۲
OCRopus	۹۷,۵۱	۱۸,۰۵۱

با بررسی‌های انجام گرفته بر روی ۹ تصویر باقیمانده از دادگان، دریافتیم که استفاده از تکنیک پنجره ۵*۵ در تصاویری با پیکسل‌های دارای مناقشه زیاد، کارا نیست. جدول ۷ دقت دو روش پنجره ۵*۵ و OCRopus را بر روی این ۹ تصویر نشان می‌دهد. همانطور که در جدول ۶ مشخص است، دقت مدل OCRopus بیش از ۲ درصد بیشتر از تکنیک پنجره است که تفاوت قابل ملاحظه‌ای است. این تفاوت قابل ملاحظه، استفاده از مدل OCRopus برای این تصاویر را ضروری می‌کند. البته زمان اجرای این دو الگوریتم تفاوت معناداری با هم دارند. تفاوت زیاد زمان اجرا تنها در صورت تفاوت معنادار دقت‌ها قابل اغماز است. تفاوت دقت ۲% این دو روش حتی با زمان اجرای نامناسب OCRopus، استفاده از این روش را توجیه‌پذیر می‌کند.

برای بررسی‌های دقیق‌تر و نتیجه‌گیری مناسب‌تر مقایسه‌ای بر روی ۲۰۸ تصویر با پیکسل‌های مورد مناقشه انجام گرفته است. در جدول ۸ روش پیشنهادی که از هر دو تکنیک پنجره ۵*۵ و مدل OCRopus استفاده کرده، با مدل OCRopus و روش پنجره ۵*۵ مقایسه شده است. نتایج جدول ۷، نمایانگر دقت و سرعت مناسب روش پیشنهادی است که با استفاده مناسب از دو تکنیک رای‌گیری پنجره‌ای و مدل OCRopus به سرعت و دقت مناسبی دست یافته است.

جدول ۷: مقایسه دقت و سرعت مدل OCRopus و پنجره ۵*۵ بر روی ۹ تصویر پر مناقشه

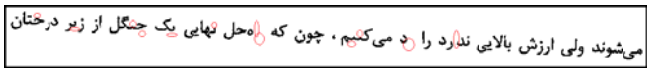
میانگین زمان اجرا برای هر تصویر (s)	دقت	نام مدل
۴,۶۱۱	۹۶,۱۱	پنجره ۵*۵ پس از رای‌گیری
۱۸,۰۵۹	۹۸,۳۱	OCRopus

جدول ۸: مقایسه دقت و سرعت مدل OCRopus و پنجره ۵*۵ بر روی کل ۲۰۸ تصویر مورد مناقشه

میانگین زمان اجرا برای هر تصویر (s)	دقت	نام مدل
۴,۵۸۳	۹۷,۶۴	پنجره ۵*۵ پس از رای‌گیری
۱۸,۰۵۱۳	۹۷,۵۴	OCRopus
۵,۲۶۲	۹۷,۷۴	مدل پیشنهادی

همان‌طور که در بخش قبل گفته شد، برای استفاده از مدل OCRopus، تمامی تصاویر می‌بایست کیفیت 300 dpi داشته باشند. از این رو، برای تصاویر با پیکسل‌های مورد مناقشه بیشتر از ۵%، قبل از دادن تصویر به مدل OCRopus، کیفیت تصویر را به 300 dpi تبدیل می‌نماییم.

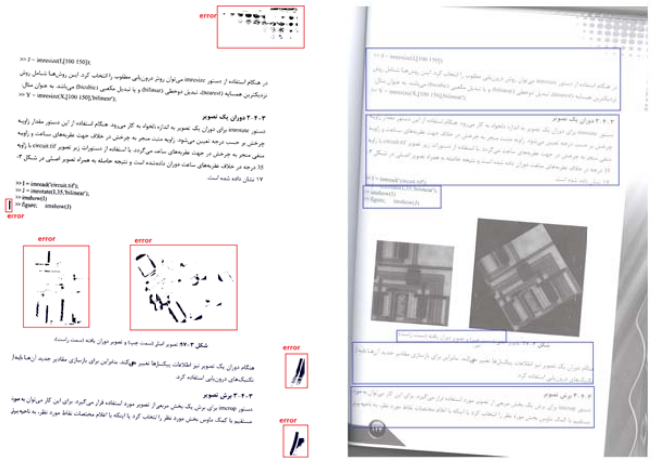
مشکلات OCRopus به این نقطه ختم نمی‌شود. سرعت اجرای این روش بسیار کندتر از روش‌های دیگر است و پیچیدگی زمانی بسیار زیادی را نسبت به دیگر روش‌ها ایجاد می‌کند. یکی دیگر از اشکالات اساسی این روش، عدم استخراج مناطق غیرمتنی تصویر است و فقط برای تشخیص مناطق متنی تصویر می‌توان از این روش استفاده نمود. این روش، مناطق متنی را در قالب فایل‌های جداگانه که هر فایل نشان دهنده یک خط می‌باشد، در خروجی قرار می‌دهد. همچنین همانند شکل ۱۸، در بعضی از تصاویر خروجی، تعدادی از نقاط، عناصر کوچک و حروف درون تصویر حذف می‌شود که این حذف، از دیگر مشکلات این روش است.



شکل ۱۸: نمونه‌ای از تصاویر خروجی OCRopus که بعضی از نقاط حذف شده‌اند.

با وجود دقت بالای مدل OCRopus در تشخیص و استخراج مناطق متنی درون تصویر، این مدل بعضی از مناطق غیرمتنی را به عنوان مناطق متنی در نظر گرفته و در فایل‌های خروجی قرار می‌دهد. شکل ۱۹ یکی از تصاویر ورودی به مدل OCRopus است. در شکل ۲۰ خروجی مدل OCRopus پس از حذف مناطق غیرمتنی به نمایش گذاشته شده است. همانطور که مشاهده می‌شود، با وجود استخراج مناطق متنی، چندین منطقه غیرمتنی نیز باقی مانده است. این اشتباه، در دیگر روش‌های مبتنی بر یادگیری عمیق و همچنین روش پیشنهادی کمتر رخ می‌دهد. این خطا، جزء خطا سیستم OCRopus در نظر گرفته نشده است. اگر این موارد به عنوان خطا لحاظ شوند، دقت روش پیشنهادی بسیار متفاوت از روش OCRopus خواهد بود. ما از روش OCRopus تنها در شرایط خاص گفته شده استفاده کرده‌ایم تا سرعت روش پیشنهادی کاهش نیابد.

شکل ۱۹: تصویری از مجموعه داده



الف: خروجی روش پیشنهادی ب: خروجی مدل OCRopus

شکل ۲۰: مقایسه خروجی روش پیشنهادی با مدل OCRopus

برای ارزیابی دقیق‌تر روش پیشنهادی از یک معیار ارزیابی دیگر بهره گرفته شده است تا میزان خطای سیستم در تشخیص مناطق غیرمتنی مشخص شود. با توجه به رابطه ۱، روشی با خروجی کل تصویر به عنوان بخش متنی، دقتی برابر با ۱۰۰% خواهد داشت که طبیعتا سیستم را در بخش‌های بعدی OCR با مشکل مواجه می‌کند. یعنی اگر بخش غیرمتنی ورودی یک سیستم OCR باشد، خروجی این سیستم، متونی بی‌مفهوم خواهند بود. پس تشخیص متون غیرمتنی نیز حائز اهمیت است. با استفاده از یک معیار دیگر، می‌توان درصد خطای سیستم را در استخراج مناطق غیرمتنی مشخص کرد.

برای محاسبه دقت تشخیص مناطق غیرمتنی، از رابطه ۲ بهره گرفته شده است. در این رابطه، تعداد پیکسل‌های غیرمتنی درست تشخیص داده شده با $N_{correct}$ و تعداد کل پیکسل‌های غیرمتنی با N_{total} نمایش داده شده است. نتایج ارزیابی با این معیار در جدول ۹ آمده است. همانطور که در جدول ۹ مشاهده می‌شود، روش پیشنهادی در استخراج مناطق غیرمتنی، کمترین میزان خطا را نسبت به دیگر روش‌ها داشته است. دلیل این کاهش خطا استفاده از ترکیب روش‌هایی است که برخی در استخراج مناطق متنی و برخی دیگر در استخراج مناطق غیرمتنی عملکرد مناسبی دارند.

$$Accuracy_{nontext} = \frac{N_{correct}}{N_{total}} \quad (۲)$$

جدول ۹: دقت تشخیص مناطق غیرمتنی روش پیشنهادی و مدل‌های دیگر

نام مدل	دقت تشخیص مناطق غیرمتنی
Layout Parser	%۹۴,۵۳
SSD	%۹۳,۳۹
YOLOv3	%۹۴,۰۲
Faster R-CNN	%۹۴,۳۶
OCRopus	%۹۲,۸۹
روش پیشنهادی تلفیقی	**%۹۴,۷۷**

$$F-measure = \frac{T_{correct}}{T_{correct} + \frac{1}{2}(T_{incorrect} + N_{incorrect})} \quad (۳)$$

در جدول ۱۰ نتایج ارزیابی روش پیشنهادی و سایر روش‌ها بر روی مجموعه دادگان PRImA نشان داده شده است. در این جدول، علاوه بر دو معیار دقت استخراج مناطق متنی و غیرمتنی از یک معیار دیگر به‌نام f-measure استفاده شده است. همان‌طور که در رابطه ۳ نشان داده شده است، معیار f-measure برای ارزیابی جامع‌تر روش‌های مختلف برای استخراج توام مناطق متنی و غیرمتنی استفاده می‌شود. در معیار f-measure، Tincorrect بیانگر تعداد پیکسل‌های به غلط متنی تشخیص داده شده و Nincorrect بیانگر تعداد پیکسل‌های به غلط غیرمتنی تشخیص داده شده، هستند. همانطور که در جدول ۱۰ مشاهده می‌شود، دقت روش پیشنهادی هم در تشخیص مناطق متنی و هم در تشخیص مناطق غیر متنی نسبت به سایر روش‌ها بالاتر بوده است که نشان دهنده توانایی بالای روش پیشنهادی در استخراج توام مناطق متنی و غیرمتنی است. همانگونه که پیش‌تر بیان شد، دلیل توانایی بالای روش پیشنهادی استفاده از ترکیب روش‌هایی است که برخی در استخراج مناطق متنی و برخی دیگر در استخراج مناطق غیرمتنی عملکرد مناسبی دارند. روش‌های جدول ۱۰ را می‌توان به دو دسته کلی روش‌های کلاسیک و روش‌های مبتنی بر یادگیری عمیق تقسیم کرد. تفاوت کلی این دو دسته از روش‌ها، تمرکز بیشتر برای استخراج مناطق غیرمتنی در روش‌های مبتنی بر یادگیری عمیق است. در واقع دقت کلی تمام روش‌ها در بخش استخراج مناطق متنی نزدیک به هم است و تفاوت عمده در بخش استخراج مناطق غیرمتنی است. با توجه به توضیحاتی که پیش‌تر ارائه شد، نمی‌توان نسبت به استخراج مناطق غیرمتنی بی‌تفاوت بود و از این رو روش‌های مبتنی بر یادگیری عمیق برای آنالیز قالب‌بندی پیشنهاد می‌شوند.

جدول ۱۰: نتیجه استفاده از معیار f-measure روش پیشنهادی و سایر روش‌ها بر روی مجموعه دادگان PRImA

نام مدل	متنی	غیرمتنی	f-measure
Layout Parser v2020	%۸۵,۴۷	%۹۲,۱	%۹۰,۷۴
SSD v2016	%۸۹,۳۲	%۹۲,۱	%۹۲,۹۳
YOLOv3 v2018	%۹۵,۷۸	%۹۳,۹	%۹۶,۷۳
Faster R-CNN v2016	%۹۳,۵۳	%۹۲,۲	%۹۵,۲۳
OCRopus v2017	%۹۳,۵۵	%۹۰,۹	%۹۵,۰۲
DICE v2009	%۹۲,۲۱	%۶۶,۲۲	%۹۰,۰۹
Fraunhofer v2009	%۹۵,۰۴	%۷۵,۱۵	%۹۳,۱۴
REGIM-ENIS v2009	%۹۱,۷۳	%۶۷,۱۳	%۸۷,۸۲
Tesseract v2009	%۹۲,۵	%۷۴,۲۳	%۹۱,۰۴
FingerReader v2009	%۹۳,۰۹	%۷۱,۷۵	%۹۱,۹
روش پیشنهادی تلفیقی	**%۹۶,۱۱**	**%۹۳,۹**	**%۹۶,۹**

۵ نتیجه‌گیری

در این تحقیق، روشی مبتنی بر رای‌گیری برای استخراج مناطق حاوی متن از درون تصویر اصلی ارائه شد. روش رای‌گیری ارائه شده در این تحقیق، جدید بوده و تا کنون از آن برای استخراج مناطق متنی بهره گرفته نشده است. با توجه به وجود چالش‌های موجود و تمرکز تحقیق بر روی سرعت مناسب روش پیشنهادی، روش رای‌گیری در دو مرحله انجام شد تا زمان روش پیشنهادی بهینه باشد. همچنین رای‌گیری به نحوی تنظیم شد تا محتمل‌ترین مناطق متنی استخراج شوند. وجود قالب‌های بسیار متنوع و وجود چندین ستون در تصویر، کار استخراج را با مشکل مواجه می‌کرد که ما توانستیم با بکارگیری از چندین روش متنوع و بکارگیری مدل OCRopus در مواقع ضروری، تا حد قابل قبولی این مشکلات را مرتفع سازیم. البته با توجه به سرعت پایین مدل OCRopus، این مدل تنها در برخی از تصاویر بسیار پیچیده، به کار گرفته شد. نتایج حاصل از روش پیشنهادی، خوب بوده است ولی همچنان باید کار بر روی آن را ادامه داد.

در این تحقیق، از مدل‌های از قبل آموزش دیده شده، استفاده شد و رای‌گیری بر روی این مدل‌ها انجام پذیرفت. در ادامه این تحقیق، می‌توان با استفاده از دادگان مناسب برای آموزش این مدل‌ها، دقت این مدل‌ها را افزایش داد. حتی با تمرکز بر روی دادگانی مناسب می‌توان مدلی جدید برای استخراج مناطق متنی ارائه کرد.

قدردانی و تشکر

نویسندگان بر خود لازم می‌دانند، مراتب تشکر صمیمانه خود را از سردبیر محترم جناب آقای دکتر کبیر و داوران محترم، به پاس راهنمایی‌ها، نظرات و پیشنهادات سازنده‌شان، اعلام دارند.

مراجع

[1] Alkhateeb, F., et al., *Arabic optical character recognition software: A review.* 2017. **27**(4): p. 763-776.

[2] Chaudhuri, A., et al., *Optical character recognition systems*, in *Optical Character Recognition Systems for Different Languages with Soft Computing*. 2017, Springer. p. 9-41.

[3] Rahmati, M., et al., *Printed Persian OCR system using deep learning. 2020. IET Image Processing, 14(15), 3920-3931.*

[4] Hesham, A.M., et al., *Arabic document layout analysis.* 2017. **20**(4): p. 1275-1287.

[5] Amer, I.M., S. Hamdy, and M.G. Mostafa. *Deep Arabic document layout analysis*. in *2017 Eighth International Conference on Intelligent Computing and Information Systems (ICICIS). (pp. 224-231). 2017.* IEEE.

[6] Ayesh, M., et al., *A Robust Line Segmentation Algorithm for Arabic Printed Text with Diacritics.* 2017. **2017**(13): p. 42-47.

[7] Guo, Y., et al. *Text line detection based on cost optimized local text line direction estimation*. in *Color Imaging XX: Displaying, Processing, Hardcopy, and Applications*. 2015. International Society for Optics and *Photonics. vol. 9395, p. 939507.*

[8] Bukhari, S.S., et al., *Coupled snakelets for curled text-line segmentation from warped document images.* 2013. **16**(1): p. 33-53.

[9] Bukhari, S.S., F. Shafait, and T.M. Breuel. *Ridges based curled textline region detection from grayscale camera-captured document images*. in *International Conference on Computer Analysis of Images and Patterns. (pp. 173-180) 2009. Springer.*

[10] Lauren, B. and L.J.U.P.A. Lee, *Perceptual information processing system. Paravue Inc.* 2003. **10**(618,543).

[11] Jebari, I. and D.J.P.E. Filliat, *Color and depth-based superpixels for background and object segmentation.* 2012. **41**: p. 1307-1315.

[12] Kaur, M. and E.N.J.I. Singh, *Image segmentation techniques: An overview.* 2002. **2**(y2).

[13] Song, Y. and H. Yan. *Image Segmentation Techniques Overview*. in *2017 Asia Modelling Symposium (AMS). 2017. (pp. 103-107). IEE*E.

[14] Kumar, M.J., et al., *Review on image segmentation techniques.* 2014: p. 2278-0882.

[15] Liew, A.-C., H. Yan, and N.-F.J.I.T.o.F.S. Law, *Image segmentation based on adaptive cluster prototype estimation.* 2005. **13**(4): p. 444-453.

[16] Fateh, M., E.J.J.o.A. Kabir, and D. Mining, *Color Reduction in Hand-drawn Persian Carpet Cartoons before Discretization using image segmentation and finding edgy regions.* 2018. **6**(1): p. 47-58.

[17] Zaharescu, M., I.C.J.J.o.I.S. Petrescu, and O. Management, *Edge detection in document analysis.* 2013. **7**(1): p. 156-165.

[18] Threshold, H.N.N.J.E.J.o.B., *A Survey on relevant Edge Detection Technique For medical image processing.* 2016. **3**(7): p. 317-327.

[19] Pirzada, S.J.H. and A. Siddiqui. *Analysis of edge detection algorithms for feature extraction in satellite images*. in *2013 IEEE International Conference on Space Science and Communication (IconSpace). 2013. (pp. 238-242). IEEE.*

[20] Phueakjeen, W., et al. *A study of the edge detection for road lane*. in *The 8th Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI) Association of Thailand-Conference 2011. 2011. (pp. 995-998).* IEEE.

[21] Gao, W., et al. *Based on soft-threshold wavelet de-noising combining with Prewitt operator edge detection algorithm*. in *2010 2nd International Conference on Education Technology and Computer. 2010. vol. 5, pp. V5-155. IEEE.*

[22] Gao, W., et al. *An improved Sobel edge detection*. in *2010 3rd International conference on computer science and information technology. 2010. vol. 5, pp. 67-71. I*EEE.

[23] Rong, W., et al. *An improved CANNY edge detection algorithm*. in *2014 IEEE International Conference on Mechatronics and Automation. 2014. pp. 577-582 IEEE.*

[۲۴] حسن‌پور و اسدی, مفاهیم جامع پردازش تصویر دیجیتال به همراه پیاده سازی الگوریتم ها با *Matlab*. انتشارات دانشگاه صنعتی شاهرود, ۱۳۹۴.

[25] Dhanachandra, N., K. Manglem, and Y.J.J.P.C.S. Chanu, *Image segmentation using K-means clustering algorithm and subtractive clustering algorithm.* 2015. **54**: p. 764-771.

[26] Bloomberg, D.S. *Multiresolution morphological approach to document image analysis*. in *Proc. of the International Conference on Document Analysis and Recognition, Saint-Malo, France*. 1991.

[27] Bukhari, S.S., F. Shafait, and T.M. Breuel. *Improved document image segmentation algorithm using multiresolution morphology. in Document recognition and retrieval XVIII. 2011. (Vol. 7874, p. 78740D) International Society for Optics and Photonics.*

[28] Abhishek, L.K.J.I.J.o.L.T.i.E. and Technology, Thinning approach in digital image processing. 2017. 326-330.

[29] Ghosh, M., et al., *Coalition game based feature selection for text non-text separation in handwritten documents using LBP based features.* 2020: p. 1-21.
[30] Ghosh, S., et al., *Text/non-text separation from handwritten document images using LBP based features: An empirical study.* 2018. **4**(4): p. 57.
[31] Harwood, D., et al., *Texture classification by center-symmetric auto-correlation, using Kullback discrimination of distributions.* 1995. **16**(1): p. 1-10.
[32] Jin, H., et al. *Face detection using improved LBP under Bayesian f*ramework. in Third International Conference on Image and Graphics (ICIG'04). 2004. (pp. 306-309).
[33] Ojala, T., et al., *Multiresolution gray-scale and rotation invariant texture classification with local binary patterns.* 2002. **24**(7): p. 971-987.
[34] Singh, S., T. Patnaik, and S. Choudhary, *Document Layout Analysis for Hindi Newspapers.*
[35] Nagy, G., S. Seth, and M.J.C. Viswanathan, *A prototype document image analysis system for technical journals.* 1992. **25**(7): p. 10-22.
[36] Pan, S.J., Q.J.I.T.o.k. Yang, and d. engineering, *A survey on transfer learning.* 2009. **22**(10): p. 1345-1359.
[۳۷] حمزکانلو, م.گ .و ح. خسروی, ناحیه‌بندی تصاویر اسناد پیچیده فارسی به بلوکهای متن، شکل و جدول. ۱۳۹۳, دانشگاه صنعتی شاهرود.
[۳۸] عباسی, م .و ح. نظام‌آبادی‌پور, روشی جدید برای استخراج متن فارسی از تصاویر بر اساس آشکار سازی لبه های رنگی در حوزه موجک. ۱۳۹۱, دانشکده فنی مهندسی دانشگاه باهنر کرمان.
[39] Redmon, J., et al. *You only look once: Unified, real-time object detection*. in *Proceedings of the IEEE conference on computer vision and pattern recognition. 2016. pp. 779-788.*
[40] Redmon, J. and A.J.a.p.a. Farhadi, Yolov3: An incremental improvement. 2018. arXiv preprint arXiv:1804.02767, 20.
[41] Ren, S., et al., *Faster R-CNN: towards real-time object detection with region proposal networks.* 2016. **39**(6): p. 1137-1149.
[42] Liu, W., et al. *Ssd: Single shot multibox detector*. in *European conference on computer vision*. 2016.
[43] Dell, Z.S.a.R.Z.a.M. *LayoutParser*. 2020; Available from: https://github.com/Layout-Parser/layout-parser.
[44] Breuel, T.M. *The OCRopus open source OCR system*. in *Document recognition and retrieval XV. 2008.* (*Vol. 6815, p. 68150F*) International Society for Optics and Photonics.
[45] Soni, R., et al., *Optimal feature and classifier selection for text region classification in natural scene images using Weka tool.* 2019. **78**(22): p. 31757-31791.
[46] Hall, M., et al., *The WEKA data mining software: an update.* 2009. **11**(1): p. 10-18.
[47] Karatzas, D., et al. *ICDAR 2013 robust reading competition*. in *2013 12th International Conference on Document Analysis and Recognition. 2013. pp. 1484-1493 IEEE.*
[48] Bisong, E., *Building machine learning and deep learning models on Google cloud platform. 2019:* (*pp. 7-10*). *Berkeley: Apress.*
[49] Wu, Y., *et al., Detectron2. 2019. URL https://github. com/facebookresearch/detectron2 2, no. 3*
[50] Imaging, T., *Monk Object Detection - A low code wrapper over state-of-the-art deep learning algorithms.* 2019.
[51] Amirreza, F. *Persian dataset of scanned images*. 2021; Available from: https://drive.google.com/file/d/1bd7UxiDp705ocnMHp6IRpx8Zqlr_r2V3/view?usp=sharing.
[52] *Antonacopoulos, A., et al. A realistic dataset for performance evaluation of document layout analysis. in 2009 10th International Conference on Document Analysis and Recognition. 2009.* (*pp. 296-300*) *IEEE.*
[53] *Antonacopoulos, A., et al. ICDAR 2009 page segmentation competition. in 2009 10th International Conference on Document Analysis and Recognition. 2009.* (*pp. 1370-1374*). *IEEE.*
[54] *Zhong, X., J. Tang, and A.J. Yepes. Publaynet: largest dataset ever for document layout analysis. in 2019 International Conference on Document Analysis and Recognition (ICDAR). pp. 1015-1022 2019. IEEE.*
[55] *Shen, Z., K. Zhang, and M. Dell. A Large Dataset of Historical Japanese Documents with Complex Layouts. in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. 2020.* (*pp. 548-549*).
[56] *Li, M., et al. Tablebank: Table benchmark for image-based table detection and recognition. in Proceedings of the 12th Language Resources and Evaluation Conference. 2020.* (*pp. 1918-1925*).
[57] *Lee, B.C.G., et al. The Newspaper Navigator Dataset: Extracting Headlines and Visual Content from 16 Million Historic Newspaper Pages in Chronicling America. in Proceedings of the 29th ACM International Conference on Information & Knowledge Management. 2020.* (*pp. 3055-3062*).

امیررضا فاتح مدرک کارشناسی خود را در رشته مهندسی کامپیوتر گرایش نرم‌افزار از دانشگاه صنعتی شاهرود با معدل ۱۸,۸ و کسب رتبه دوم در سال ۱۳۹۸ دریافت کرد. ایشان در همان سال در مقطع کارشناسی ارشد در رشته مهندسی کامپیوتر و گرایش هوش مصنوعی و رباتیکز در دانشگاه صنعتی شاهرود مشغول به تحصیل شد و با معدل ۱۹,۳۴ بدون احتساب پایان‌نامه موفق به کسب رتبه اول گردید. پروژه کارشناسی ارشد ایشان "آنالیز قالب‌بندی اسناد فارسی" است. زمینه تحقیقاتی ایشان پردازش تصویر، یادگیری ماشین و یادگیری عمیق می‌باشد.

محسن رضوانی مدرک دکتری خود را در حوزه امنیت شبکه و از دانشگاه UNSW استرالیا دریافت نموده است. ایشان در حال حاضر دانشیار دانشگاه صنعتی شاهرود است.

علیرضا تجری استادیار دانشکده مهندسی کامپیوتر دانشگاه صنعتی شاهرود است. وی در دوره کارشناسی در رشته مهندسی کامپیوتر و گرایش مهندسی نرم‌افزار و در دوره‌های کارشناسی ارشد و دکتری، در رشته مهندسی کامپیوتر و گرایش معماری سیستم‌های کامپیوتری تحصیل کرده است. همه مدارک تحصیلی وی از دانشگاه صنعتی امیرکبیر اخذ شده است.

منصور فاتح مدرک کارشناسی خود را در رشته مهندسی برق از دانشگاه صنعتی شاهرود در سال ۱۳۸۶ دریافت کرد. سپس کارشناسی ارشد و دکتری خود را در رشته‌های مهندسی پزشکی و الکترونیک دیجیتال درسال‌های ۱۳۸۸ و ۱۳۹۳ از دانشگاه تربیت مدرس دریافت کرد. پروژه کارشناسی ارشد خود را با عنوان "بررسی نقش و اثر نور پلاریزه در درماتوسکپی از بدن با استفاده از شبیه سازی" و پروژه دکتری خود را با عنوان "خواندن خودکار نقشه‌های دستی فرش" به انجام رسانید. از سال ۱۳۹۴ ایشان عضو هیئت علمی دانشگاه صنعتی شاهرود بوده و زمینه تحقیقاتی ایشان پردازش تصویر و یادگیری ماشین می‌باشد.