





دانشکده مهندسی برق و رباتیک

رشته مهندسی برق گرایش مخابرات

پایان نامه کارشناسی ارشد

استخراج متن از نقشه جغرافیایی با استفاده از کوانتیزاسیون فضای رنگ

نگارنده: ساناز هادی پور آبکنار

استاد راهنما:

دکتر علیرضا احمدی فرد

بهمن ۱۳۹۵

تقدیم

گرچه میدانم هیچ کلام و عملی از سوی من نمی‌تواند متی ذره‌ای از دریای محبت بیکران

شما را جبران کند، اما ...

این پایان‌نامه را به شما تقدیم می‌کنم، شما که همچون شمع ذره‌ذره به پای من

آب گشتید و از هیچ کاری برای آسایش و فوشمالی من دریغ ننمودید، تقدیم به شما که

بهترین فلايق فداييد و نامتان همواره روشني‌بخش زندگي من و مايه افتخاره فواهد بود.

"تقدیم به پدر و مادر عزیزم و برادر مهربانم"

تشکر و قدردانی

پیش از هر چیز سپاسگزار خداوند منانی، هستم که هر چه داشته و دارم از اوست، بر آستان بیکرانش سر تعظیم فرود آورده و به پاس هر آن چه به من بخشیده است، سجده شکر به جای می آورم.

بانهایت تواضع و احترام از استاد رهنمای ارجمندم، جناب آقای دکتر علیرضا احمدی فرد شکر و قدردانی می نمایم. رهنمایی های ایشان کمک بزرگی در انجام این پروژه برای اینجانب بود. مهم تر از آن نصیحت های ارزشمندی است که همواره آویزه گوش و چراغ راه آینده من است. شکر صمیمانه خود را به محض ایشان تقدیم می دارم و می دانم که بی شک انجام این تحقیق بدون رهنمایی ها و تشویق های ایشان میسر نبود.

همچنین از اعضای محترم کمیته ارزشیابی، جناب آقای دکتر حسین خسروی و جناب آقای دکتر مهدی گرایلو که بانهایت صبر و متامل این پایان نامه را مورد بررسی قرار داده و نکات ارزشمندی را به بنده گوشزد نموده اند، قدردانی می نمایم.

همچنین از تمامی اساتیدی که در طول دوران تحصیل چه در دانشگاه صنعتی شاهرود و چه در دانشگاه کیلان مطالب گرانبهائی را به من آموختند و لطف خود را از بنده حقیر دریغ ننموده اند، کمال شکر را دارم. بر خود لازم می دانم بار دیگر از استاد کرامتدارم جناب آقای دکتر علیرضا احمدی فرد بابت تدریس دوس پردازش تصویر و تشخیص آماری الگو و نیز از جناب آقای دکتر حسین خسروی بابت تدریس درس شبکه های عصبی شکر نمایم؛ چرا که این دوس منبع اصلی ایده های من برای انجام این پروژه بوده اند. از جناب آقای دکتر

احسان الله بکیریم که به صورت محکمه ای از یاری ایشان استفاده نموده ام و لطف ایشان شامل حال بنده شده است، سپاسگزارم.

در طول زندگی دوستانی داشته ام که همواره موجب دلگرمی من بوده اند. دوستانی که در لحظات شادی همراه من بوده اند و بهترین

خاطرات زندگیم را رقم زده اند و در لحظات تلخ با من همدردی نموده اند. نهایت بی انصافی است که از دوستی بی نظیرشان یادی نکنم و از

آن ها سپاسگزاری ننمایم.

آخر از همه امانه کمتر از همه! بهترین سپاس و درود خود را تقدیم به پدر و مادر مهربانم و برادر، همواره همراهم می کنم که در هیچ یک از لحظات

زندگی مرا تنها نگذاشتند و همواره دلم به بودشان، گلشان، هدایشان و محبتشان گرم بود.

تعهد نامه

اینجانب ساناز هادی پور آبکنار دانشجوی دوره کارشناسی ارشد رشته مهندسی برق / مخابرات دانشکده مهندسی برق و رباتیک دانشگاه صنعتی شاهرود نویسنده پایان نامه استخراج متن از نقشه جغرافیایی با استفاده از کوانتیزاسیون فضای رنگ تحت راهنمایی دکتر علیرضا احمدی فرد متعهد می شوم:

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

تاریخ:

امضای دانشجو:

مالکیت نتایج و حق نشر

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

لایه‌های متنی در نقشه‌های جغرافیایی، به واسطه مرتبط ساختن اسامی خیابان‌ها و ساختمان‌های مهم به موقعیت آن‌ها روی نقشه‌ها، اطلاعات جغرافیایی ارزشمندی را فراهم می‌آورند. بیشتر تحقیقات انجام شده در این زمینه، بر روی متون لاتین متمرکز می‌باشند. روش‌های ارائه شده بر روی نقشه‌های با زبان فارسی نیز، نتیجه قابل قبولی را ارائه نمی‌کنند.

در این پایان‌نامه یک روش جدید برای استخراج خودکار متن از نقشه‌های شهری با استفاده از نگاشت خود سامان‌ده ارائه شده است. در رهیافت پیشنهادی، از یک الگوریتم جابه‌جایی میانگین برای کاهش تغییرات فضای رنگ استفاده می‌شود. سپس با استفاده از روش نگاشت خود سامان‌ده، ابعاد فضای رنگ کاهش پیدا می‌کند و فضای رنگ کوانتیزه می‌شود. نتایج اعمال الگوریتم پیشنهادی روی نقشه‌های مختلف نشان می‌دهد که متون در نقشه‌ها در یکی از سه لایه یک، پنج یا هفت حاصل از نگاشت خود سامان‌ده قرار دارند. بنابراین سیستم تعداد عناصر متصلی را که مساحت آن‌ها در یک آستانه مشخص قرار دارد، در هر یک از این سه لایه می‌شمارد؛ لایه‌ای که بیشترین تعداد عناصر متصل را داشته باشد، لایه منتخب متنی است. در این لایه هر یک از متون بسته به رنگی که در تصویر اصلی داشته‌اند، دارای درجه روشنایی متفاوتی در تصویر سطح خاکستری هستند و پس‌زمینه سیاه فاقد هر گونه اطلاعات متنی است. با در نظر گرفتن رنگ سیاه موجود در لایه منتخب به عنوان پس‌زمینه و سایر رنگ‌ها به عنوان پیش‌زمینه، لایه متنی بدست می‌آید.

برای آماده‌سازی کلمات جهت ورود به نرم‌افزار OCR، با استفاده از یک المان ساختاری و عملگر ریخت‌شناسی گسترش، تصویر دودویی را گسترش داده، اجزای متصل موجود در تصویر گسترش‌یافته را می‌یابیم. هر یک از این اجزای گسترش‌یافته را در تصویر دودویی اصلی مرتبط با آن جزء "AND" می‌کنیم تا تصویر هر کلمه به صورت جداگانه ظاهر شود. سپس از خاصیت جعبه محیطی استفاده

می‌کنیم و پروفایل هر تصویر را در راستای سطرها درون جعبه محیطی هر کلمه در زوایای مختلف بین ۹۰- تا ۹۰+ درجه می‌یابیم. هر زاویه‌ای که در آن تعداد مقادیر غیرصفر پروفایل تصویر کلمه، کمترین مقدار خود را داشته باشد، زاویه چرخش مطلوب ما است. بنابراین هر کلمه به اندازه زاویه چرخش مطلوب خودش چرخانده می‌شود تا در راستای افقی قرار گیرد.

این روش مستقل از قلم، اندازه، جهت و رنگ متون است و می‌تواند هم در مورد نقشه‌های فارسی و هم نقشه‌های لاتین به کار رود. نتایج تجربی دقت ۹۰/۴٪ و فراخوانی ۸۶/۶٪ را برای رهیافت پیشنهادی در استخراج متون از نقشه‌های جغرافیایی نشان می‌دهد.

کلیدواژه: استخراج متن - آماده‌سازی برای ورود به OCR - جابه‌جایی میانگین - نگاشت خود سامان‌ده -

کوانتیزاسیون فضای رنگ

لیست مقالات مستخرج:

- 1- S. Hadipour Abkenar and A. Ahmadyfard, "Text Extarction from Raster Maps using Color Space Quantization," *fifth International Conference on Data Mining and Knowledge Management Process (DKMP)*, 29 Jan-2017.
- ۲- ساناز هادی پور آبکنار و علیرضا احمدی فرد، "استخراج متن از نقشه جغرافیایی با استفاده از نگاشت خود سامان ده"، ۱۲۵مین کنفرانس مهندسی برق ایران (ICEE 2017)، اردیبهشت ۱۳۹۶ (تحت داوری).

فهرست مطالب

۱	پیشگفتار
۲	۱-۱ انگیزه تحقیق
۸	۲-۱ هدف تحقیق
۹	۳-۱ مروری بر فصل‌های پایان‌نامه
۱۱	پیش‌درآمد موضوع و مروری بر کارهای گذشته
۱۲	۲-۱ مقدمه
۱۳	۲-۲ استخراج خطوط خیابان
۱۴	۳-۲ استخراج تقاطع خیابان
۱۸	۴-۲ استخراج ناحیه مورد نظر
۱۹	۵-۲ استخراج متن
۳۶	۶-۲ جمع‌بندی و نتیجه‌گیری
۴۱	تئوری الگوریتم‌های مورد استفاده
۴۲	۱-۳ مقدمه

۴۲ کوانتیزاسیون ۲-۳
۴۲ کوانتیزاسیون رنگ ۱-۲-۳
۴۶ کاربرد ۲-۲-۳
۴۷ الگوریتم جابه‌جایی میانگین ۳-۳
۵۰ انواع کرنل‌ها ۱-۳-۳
۵۱ خوشه‌بندی الگوریتم جابه‌جایی میانگین ۲-۳-۳
۵۴ نگاشت خود سامان‌ده (SOM) ۴-۳
۵۴ تعریف نگاشت خود سامان‌ده ۱-۴-۳
۵۵ ساختار نگاشت خود سامان‌ده ۲-۴-۳
۵۸ الگوریتم یادگیری ۳-۴-۳
۵۹ شبکه کوهنن ۴-۴-۳
۶۱ جمع‌بندی و نتیجه‌گیری ۵-۳
۶۳ روش پیشنهادی و پیاده‌سازی آن
۶۴ مقدمه ۱-۴
۶۶ اعمال روش جابه‌جایی میانگین ۲-۴
۶۸ اعمال روش نگاشت خود سامان‌ده ۳-۴
۷۶ تحلیل اجزای متصل ۴-۴
۸۲ آماده‌سازی جهت ورود به OCR ۵-۴

۶-۴ نتایج تجربی ۸۶

نتیجه‌گیری و پیشنهادها ۹۳

۱-۵ مروری بر فصل‌های پایان‌نامه ۹۴

۲-۵ نتیجه‌گیری ۹۵

۳-۵ کارهای آتی ۹۶

مراجع ۹۷

فهرست شکل‌ها

شکل ۱-۱: همپوشانی متون با خطوط خیابان روی بخشی از نقشه شهر تهران ۵

شکل ۱-۲: همپوشانی متون با یکدیگر روی بخشی از نقشه شهر تهران ۵

شکل ۱-۳: نمونه‌ای از همپوشانی متون با گرافیک‌ها روی بخشی از نقشه شهر تهران ۶

شکل ۱-۴: بخشی از نقشه کوآلامپور مالزی که نمایانگر یک سند تک‌جهته است و دو نمونه از متون

آن با مستطیل مشخص شده است ۷

شکل ۱-۵: بخشی از نقشه تهران که دارای خطوط چندجهته است و تعدادی از آن‌ها با مستطیل

مشخص شده‌اند ۷

شکل ۱-۶: بخشی از نقشه تهران که دارای متون خمیده است که با بیضی مشخص شده‌اند ۱۱

شکل ۱-۲: مثالی از نتایج برداری‌سازی جاده روش چیانگ و نوبلاک روی یک نقشه (الف) یک نقشه

(ب) پیکسل‌های خیابان نقشه (الف) (ج) داده‌های برداری خیابان از نقشه (الف) ۱۴

شکل ۲-۲: استخراج تقاطع خیابان توسط روش چیانگ و همکارانش (نقشه مکان‌نگاری USGS،

۲ m/pixel، سنت لوئیس) ۱۷

شکل ۲-۳: استخراج آبراه با استفاده از الگو در روش تبسم و شریف‌الدین ۱۹

شکل ۲-۴: استخراج خطوط متنی توسط روش لیچر و هونز ۲۱

شکل ۲-۵: اعمال روش هرم در کارتن و ان‌جی‌اف (نقشه اصلی ب) متون استخراج شده ۲۲

شکل ۲-۶: نتایج استخراج خط متنی توسط روش روی و همکارانش روی الف) یک روزنامه ب) یک

سند معنایی ج) نقشه ۲۷

شکل ۲-۷: نتیجه استخراج متن از یک نقشه توسط روش دکتر کبیر و همکارانش. الف) تصویر سطح

خاکستری ب) لایه متنی استخراج شده ۳۵

شکل ۲-۸: استخراج متن از نقشه با استفاده از تبدیل فاصله الف) یک تصویر سطح خاکستری ب)

متن بازیابی شده ۳۶

شکل ۳-۱: نمونه‌ای از عمل کوانتیزاسیون رنگ روی یک تصویر واقعی و در فضای رنگ سه‌بعدی .. ۴۳

شکل ۳-۲: عمل کوانتیزاسیون در صفحه تصویر ۴۶

شکل ۳-۳: چگونگی یافتن نقطه همگرایی در الگوریتم جابه‌جایی میانگین ۵۰

شکل ۳-۴: پیدا کردن قله جابه‌جایی میانگین ۵۲

شکل ۳-۵: خوشه‌بندی جابه‌جایی میانگین ۵۳

شکل ۳-۶: نتیجه اعمال روش جابه‌جایی میانگین روی تصویر یک دلقک ۵۴

شکل ۳-۷: نمایش نگاهت از فضای با ابعاد بالا به فضای با ابعاد کم ۵۵

شکل ۳-۸: ساختار شبکه SOM ۵۶

شکل ۳-۹: تعریف‌های همسایگی: الف) خطی، ب) مربعی، ج) شش‌ضلعی. همسایگی در اطراف یک

نرون برنده ۵۷

شکل ۳-۱۰: شرح آموزش یک نگاهت خود سامان‌ده ۵۸

- شکل ۳-۱۱: نرون‌های لایه خروجی یک بعدی ۵۹
- شکل ۴-۱: روندنمای مراحل پیاده‌سازی الگوریتم جهت استخراج لایه متنی ۶۵
- شکل ۴-۲: بخشی از نقشه شهر تهران ۶۶
- شکل ۴-۳: نتیجه اعمال الگوریتم جابه‌جایی میانگین روش شکل ۴-۲ ۶۸
- شکل ۴-۴: لایه اول حاصل از اعمال SOM روی شکل ۴-۳ ۷۰
- شکل ۴-۵: لایه دوم حاصل از اعمال SOM روی شکل ۴-۳ ۷۰
- شکل ۴-۶: لایه سوم حاصل از اعمال SOM روی شکل ۴-۳ ۷۱
- شکل ۴-۷: لایه چهارم حاصل از اعمال SOM روی شکل ۴-۳ ۷۱
- شکل ۴-۸: لایه پنجم حاصل از اعمال SOM روی شکل ۴-۳ ۷۲
- شکل ۴-۹: لایه ششم حاصل از اعمال SOM روی شکل ۴-۳ ۷۲
- شکل ۴-۱۰: لایه هفتم حاصل از اعمال SOM روی شکل ۴-۳ ۷۳
- شکل ۴-۱۱: لایه هشتم حاصل از اعمال SOM روی شکل ۴-۳ ۷۳
- شکل ۴-۱۲: تصویر سطح خاکستری لایه اول SOM ۷۴
- شکل ۴-۱۳: تصویر سطح خاکستری لایه پنجم SOM ۷۵
- شکل ۴-۱۴: تصویر سطح خاکستری لایه هفتم SOM ۷۵
- شکل ۴-۱۵: تصویر سطح خاکستری لایه منتخب ۷۶

- شکل ۴-۱۶: تصویر باینری لایه متنی ۷۷
- شکل ۴-۱۷: بخشی از نقشه شهر تهران ۷۸
- شکل ۴-۱۸: نتیجه اعمال روش جابه‌جایی میانگین روی شکل ۴-۱۷ ۷۸
- شکل ۴-۱۹: لایه منتخب حاصل از اعمال روش نگاشت خود سامان‌ده روی شکل ۴-۱۸ ۷۹
- شکل ۴-۲۰: لایه متنی حاصل شده ۷۹
- شکل ۴-۲۱: بخشی از نقشه شهر لس‌آنجلس ۸۰
- شکل ۴-۲۲: نتیجه اعمال روش جابه‌جایی میانگین روی شکل ۴-۲۱ ۸۰
- شکل ۴-۲۳: لایه منتخب حاصل از اعمال روش نگاشت خود سامان‌ده روی شکل ۴-۲۲ ۸۱
- شکل ۴-۲۴: لایه متنی حاصل شده ۸۱
- شکل ۴-۲۵: روندنمای مراحل آماده‌سازی کلمات برای ورود به OCR ۸۲
- شکل ۴-۲۶: تصویر دودویی گسترش‌یافته شکل ۴-۱۶ ۸۳
- شکل ۴-۲۷: پروفایل تصویر در چند زاویه مختلف ۸۴
- شکل ۴-۲۸: نمونه‌هایی از کلماتی که توسط سیستم به صورت جداگانه شناسایی و چرخانده می‌شوند ۸۶
- شکل ۴-۲۹: مثالی از هریک از منابع نقشه مورد استفاده. الف) نقشه بدست آمده از Bing [۵۲] ب) نقشه بدست آمده از کتاب اول [۸] ج) نقشه بدست آمده از سایت شهرداری تهران [۵۱] د) نقشه

بدست آمده از google maps [۱۱] ه) نقشه بدست آمده از Tehran maps [۵۳] ی) نقشه متفرقه‌ای

که با جستجو در گوگل بدست آمده است ۸۸

فهرست جداول

جدول ۴-۱: منبع هر یک از ۴۰ نقشه مورد بررسی، ابعاد، زبان، درصد دقت و فراخوانی آن..... ۹۰

جدول ۴-۲: درصد استخراج متون از نقشه‌های منابع مختلف ۹۲

علامہ و اختصارات

ANN	Artificial Neural Network
CDA	Conditional Dilation Algorithm
FN	False Negative
FP	False Positive
GIS	Geographical Information Systems
OCR	Optical Character Recognition
RGB	Red Green Blue
ROI	Region of Interest
SOM	Self Organizing Map
SSOD	Single String Orientation Detection algorithm
SVM	Support Vector Machine
TP	True Positive
USGS	United States Geological Survey

فصل اول

پیشگفتار

۱-۱ انگیزه تحقیق

تصاویر به عنوان یکی از مهم‌ترین رسانه‌ها برای انتقال اطلاعات در نظر گرفته می‌شوند. تأثیر یک تصویر بسیار بیشتر از صدها خط نوشته است. بنابراین، درک تصویر و استخراج اطلاعات از آن می‌تواند در شاخه یادگیری ماشین^۱ مهم باشد [۱].

بسیاری از سازمان‌ها، مجموعه‌ای بزرگ از اسناد گرافیکی شامل نقشه‌ها، ترسیم‌های مهندسی، شماتیک‌های الکتریکی و توضیحات فنی دارند که به صورت معمول از آن‌ها در انجام بسیاری از کارهایشان استفاده می‌کنند. پیشرفت‌های اخیر در تکنولوژی کامپیوتر به اطلاعات گرافیکی اجازه می‌دهد تا در سیستم ذخیره شده و به صورتی راحت‌تر و کارآمدتر از نظر هزینه و دقت در مقایسه با نمونه‌های کاغذی به صورت الکترونیکی در دسترس قرار گیرند [۲]. سازمان‌های دولتی و تجاری، اغلب باید نقشه‌های کاغذی موجود از شهرها و نواحی وسیع‌تر را به صورتی که برای کامپیوتر قابل فهم باشد، تبدیل کنند تا بتوانند واسطی برای سیستم‌های اطلاعات جغرافیایی^۲ (GIS) موجود، فراهم کنند [۳]. این قبیل اطلاعات الکترونیکی قابلیت به‌روزرسانی و ویرایش بیشتری نسبت به قالب کاغذی دارند و فضای کمتری را نیز اشغال می‌کنند.

یک نقشه نمایشی دیداری از یک ناحیه است؛ نمایشی نمادین که روابط بین عناصر از قبیل اشیاء، نواحی و موضوع‌ها را مشخص می‌کند. یک نقشه شهری نه تنها شامل خیابان‌ها و اسامی آن‌ها است؛ بلکه شامل اطلاعات مهم دیگری از قبیل مکان‌های دیدنی، مکان‌های عمومی از قبیل پارک‌ها، مساجد، بیمارستان‌ها، مدارس و گاهی شامل شبکه حمل و نقل مانند مترو یا خط‌آهن نیز هست [۱]. با گسترش روزافزون حجم اطلاعاتی که از نقشه‌ها بدست می‌آیند و با توجه به این که مسیرهای ارتباطی جاده‌ای پیوسته در حال افزایش هستند تا مردم بتوانند آسان‌تر از مکانی در یک شهر به

¹ Machine learning

² Geographical Information Systems

مکانی دیگر بروند، تعداد نقشه‌ها و تراکم اطلاعات روی آن‌ها رو به افزایش است.

همچنین نقشه‌ها در مقایسه با سایر داده‌های جغرافیایی از قبیل داده‌های برداری، تصاویر ماهواره‌ای، فرهنگ‌های جغرافیایی و غیره آسان‌تر در دسترس قرار می‌گیرند. به دلیل وجود اسکنرهای با کیفیت بالا و اینترنت، اکنون می‌توانیم به نقشه‌های مختلفی از نواحی گوناگون از سطح جهان دست یابیم. با تبدیل برچسب‌های متنی در یک نقشه به متون قابل ویرایش ماشین، می‌توانیم دانش جغرافیایی را برای فهم یک ناحیه از نقشه تولید کنیم؛ بدون این که سایر داده‌های جغرافیایی آن را در دست داشته باشیم. به علاوه می‌توان متون تشخیص داده شده را از نقشه استخراج کرد و برای شاخص‌دهی^۱ و بازبینی سایر داده‌های جغرافیایی به کار برد [۴].

برچسب‌های متنی در نقشه‌های جغرافیایی اسامی مکان‌ها را به موقعیت‌های جغرافیایی آن‌ها مرتبط می‌کنند. تبدیل برچسب‌های متنی در یک نقشه به متون ماشینی قابل ویرایش، به تولید دانش جغرافیایی جهت درک یک ناحیه کمک می‌کند [۵]. بنابراین متون در نقشه‌ها از جمله مهم‌ترین اطلاعات هستند.

در میان اسناد موجود، نقشه‌های شهری از جمله متداول‌ترین ابزارها برای راهنمایی افراد و یافتن موقعیت جغرافیایی مکان‌ها هستند [۶]. یافتن مکانی خاص در یک شهر یا مسیر ارتباطی بین دو نقطه در گرو نقشه‌خوانی صحیح است؛ به گونه‌ای که بتوانیم نام خیابان‌ها را به درستی پیدا کرده و طبق آن مسیر مورد نظر را بیابیم. نقشه‌خوانی در صنعت گردشگری نیز کاربرد دارد؛ چه برای افرادی که از شهری از داخل یک کشور به شهری دیگر می‌روند و چه برای افرادی که با زبانی دیگر از خارج از یک کشور به آن وارد می‌شوند. با فراهم ساختن شرایطی که طی آن افراد با زبان بیگانه بتوانند نام مکان مورد نظر خود را به نرم‌افزاری بدهند و آن را روی نقشه محلی کشور مقصد بیابند، می‌توان به صنعت

¹ Indexing

گردشگری یک کشور کمک بسیاری کرد. اگر بتوان متون موجود در نقشه را استخراج کرد، می‌توان نرم‌افزارهای مسیریاب پیشین را بهبود بخشید و مکان‌یاب جامع‌تری را تولید کرد. از سوی دیگر امروزه با پیشرفت تکنولوژی و پیدایش خودروهای بدون سرنشین^۱، گاهی لازم است تا کاربر اسامی مکان‌های مورد نظر خود را به عنوان ورودی به اتومبیل بدهد تا اتومبیل بدون هدایت او بتواند مسیر خود را یافته و به درستی به مقصد برسد. رسیدن به تمام این‌ها در گرو پیدایش روشی دقیق برای نقشه‌خوانی است که در آن دخالت کاربر و نیز زمان پردازش تا حد امکان کم باشد.

جداسازی متون و گرافیک‌های همپوشان یک مسأله چالش‌برانگیز در تحلیل تصاویر اسناد است. این موضوع کاربردهای بسیاری دارد. به عنوان مثال از آن می‌توان در پردازش فرم‌ها، تفسیر نقشه‌ها و تفسیر ترسیم‌های مهندسی استفاده کرد که در هر یک از آن‌ها متون و گرافیک‌ها در راه‌های متفاوتی پردازش می‌شوند [۷].

همان‌گونه که پیشتر نیز بیان شد، مهم‌ترین کاربرد استخراج متن از نقشه‌ها در سیستم‌های اطلاعات جغرافیایی (GIS) است؛ اما از سیستم‌های استخراج متون می‌توان در کاوش داده‌ها^۲، صنعت گردشگری و نیز سیستم‌های خودکار ماشین‌های بدون سرنشین، جهت بهبود سیستم مسیریاب آن‌ها استفاده نمود. ارائه یک سیستم نقشه‌خوان مناسب که بتواند دقت (بیشترین مقدار از متون همراه با کم‌ترین مقدار از خطوط خیابان‌ها و نمادهای گرافیکی) را در کمترین زمان و با حداقل دخالت کاربر بهبود دهد، هدف همه محققان در این زمینه است.

تشخیص متون از نقشه‌های شهری به چند دلیل، کاری چالش‌برانگیز است. اول این که کیفیت تصویر نقشه‌ها معمولاً در طی مراحل اسکن و فرآیندهای فشرده‌سازی تصویر کاهش می‌یابد. دومین دلیل این است که برجسب‌های متنی در یک نقشه جغرافیایی می‌توانند اندازه‌ها و قلم‌های مختلفی

¹ Auto-pilot automobiles

² Data mining

داشته باشند و در پاره‌ای از موارد با یکدیگر یا با سایر خصیصه‌های موجود در نقشه مانند خطوط خیابان‌ها همپوشانی داشته باشند (در شکل (۱-۱) بیضی‌های قرمز رنگ همپوشانی متون با خطوط خیابان را نشان می‌دهند. دایره‌های آبی رنگ در شکل (۱-۲) مثالی از همپوشانی متون با یکدیگر و در شکل (۱-۳) نمونه‌ای از همپوشانی متون با گرافیک‌ها را نشان می‌دهند). سوم این که برچسب‌های متنی در یک نقشه معمولاً یک جهت خاص را دنبال نمی‌کنند [۴].



شکل ۱-۱: همپوشانی متون با خطوط خیابان روی بخشی از نقشه شهر تهران (نقشه اصلی از [۸] بدست آمده است)



شکل ۱-۲: همپوشانی متون با یکدیگر روی بخشی از نقشه شهر تهران (نقشه اصلی از [۸] بدست آمده است)



شکل ۱-۳: نمونه‌ای از همپوشانی متون با گرافیک‌ها روی بخشی از نقشه شهر تهران (نقشه اصلی از [۸] بدست آمده است)

برای استفاده از نرم‌افزار نویسه‌خوان نوری^۱ (OCR) خطوط متنی موجود در یک سند باید پیش از تشخیص، بخش‌بندی شوند. هنگامی که خطوط متنی در تصویر یک سند، موازی یکدیگر و در یک جهت هستند (اسناد تک‌جهته^۲، مانند آنچه در شکل (۱-۴) مشاهده می‌شود) روش‌های ساده برای بخش‌بندی متون تک‌جهته مناسب هستند؛ اما اسناد بسیار دیگری نیز وجود دارند که در آن‌ها خطوط متنی ممکن است در جهات مختلف چاپ شوند (اسناد چندجهته^۳، مانند آنچه در شکل (۱-۵) دیده می‌شود) یا این که خطوط متنی موجود در آن‌ها به شکل خمیده باشند [۹] (مانند آنچه در شکل (۱-۶) می‌بینید).

تکنیک‌های بسیاری برای استخراج خطوط متنی از اسناد تک‌جهته وجود دارد، اما کارهای منتشر شده روی اسناد دارای متون چندجهته و خمیده بسیار کم است. اکثر تکنیک‌های مطرح شده نیاز به دخالت زیاد کاربر دارند و تنها روی تعداد محدودی از اسناد قابل استفاده هستند؛ لذا نتیجه خوبی به همراه ندارند و طی فرآیند این سیستم‌ها، بسیاری از متون از بین می‌روند.

¹ Optical Character Recognition

² Single oriented

³ Multi oriented



شکل ۱-۴: بخشی از نقشه کوآلالمپور مالزی که نمایانگر یک سند تک‌جهته است و دو نمونه از متون آن با مستطیل مشخص شده است [۱۰].



شکل ۱-۵: بخشی از نقشه تهران که دارای خطوط چندجهته است و تعدادی از آن‌ها با مستطیل مشخص شده‌اند [۱۱].



شکل ۱-۶: بخشی از نقشه تهران که دارای متون خمیده است که با بیضی مشخص شده اند [۱۱].

۲-۱ هدف تحقیق

هدف از این تحقیق ارائه روشی خودکار است که در آن بتوان متن را از نقشه جغرافیایی به گونه‌ای استخراج کرد که حداکثر تعداد متون ممکن از نقشه استخراج شوند و خطوط جاده‌ها یا نمادهای گرافیکی موجود در نقشه که جزء متون محسوب نمی‌شوند، تا حد امکان حذف شوند. در نهایت کلمات به صورت جداگانه شناسایی و استخراج شده، در راستای افقی بچرخند و برای ورود به OCR آماده شوند. سعی ما بر این است که روش پیشنهادی به گونه‌ای باشد که هم برای متون لاتین و هم برای متون فارسی قابل استفاده باشد. هدف دیگری که در این پایان‌نامه مد نظر قرار گرفته، این است که روش ارائه شده برای همه قلم‌ها، اندازه قلم‌ها، رنگ‌های متون و جهات مناسب باشد.

در این پایان‌نامه تلاش شده است تا با استفاده از تکنیک‌های ساده، روشی جامع طراحی گردد تا اطلاعات جغرافیایی را برای تشخیص در سیستم OCR آماده سازد. همچنین در آن به سؤالات زیر پاسخ داده می‌شود:

۱- چگونه می‌توان نقشه‌ای با چند رنگ را بدون دخالت کاربر کوانتیزه کرد؟

۲- فایده کوانتیزاسیون رنگ چیست؟

۳- آیا روشی وجود دارد که برای متون زبان‌های مختلف، چه فارسی و چه لاتین، مناسب باشد؟

۴- چگونه می‌توان در یک نقشه، متون دارای قلم‌ها و اندازه‌های مختلف را که دارای رنگ‌ها و

جهات متفاوت هستند، از خطوط جاده و اطلاعات پس‌زمینه جدا نمود؟

۳-۱ مروری بر فصل‌های پایان‌نامه

در فصل دوم به بررسی سابقه موضوع استخراج متن و خطوط جاده از نقشه جغرافیایی پرداخته می‌شود. در این فصل در ابتدا به اطلاعاتی که از نقشه‌ها قابل استخراج هستند، اشاره می‌کنیم و در هر مورد بخشی از کارهای انجام شده را شرح می‌دهیم. سپس اهمیت موضوع استخراج متن، سابقه آن و کارهای انجام شده در این حوزه مورد بحث قرار می‌گیرند. در فصل سوم، کوانتیزاسیون رنگ، روش جابه‌جایی میانگین و روش نگاشت خود سامان‌ده تعریف شده‌اند. در فصل چهارم، نقش تکنیک‌های بیان شده در فصل قبل در این پایان‌نامه مورد بررسی قرار می‌گیرد؛ الگوریتم مطرح شده روی چند نقشه اجرا شده و نتایج به دست آمده در هر مرحله تحلیل می‌شود. در فصل پنجم، به‌طور مختصر، مروری بر کارهای انجام شده در این پایان‌نامه صورت می‌گیرد، نتایج جمع‌بندی می‌شوند و پیشنهادهایی برای ادامه تحقیق ارائه می‌گردد.

فصل دوم

پیش درآمد موضوع و مروری بر کارهای گذشته

۲- ۱ مقدمه

نرم افزار GIS مرجع اصلی اطلاعات جغرافیایی و یکی از قدرتمندترین سیستم‌های اطلاعات جغرافیایی است که می‌تواند ما را در موارد مختلفی از جمله نوع پوشش گیاهی یک منطقه، تراکم جمعیت، وضعیت راه‌های ارتباطی و ... آگاه سازد. برای فراهم ساختن اطلاعات مورد استفاده GIS باید ورود اطلاعات به نحو درستی انجام گیرد و امکان نقشه‌خوانی برای سیستم فراهم باشد.

تشخیص نقشه‌ها و متون در یک سیستم بر عهده OCR است. برای تشخیص نویسه^۱ در OCR، باید مکان نویسه‌ها از پیش شناسایی شده، نویسه‌ها بخش‌بندی و از پس‌زمینه جدا گردند.

پیشتر محققان گاه طی پژوهش‌های خود خطوط جاده و تقاطع‌ها را استخراج کرده‌اند و نیز گاهی به استخراج خطوط متنی پرداخته‌اند. استخراج خطوط خیابان کمک می‌کند تا خطوط روی نقشه‌های متنی استخراج شوند و بتوانیم با حذف آن‌ها، متون و نمادهای گرافیکی را در پس‌زمینه بیابیم. گرچه قطعاً استخراج خطوط خیابان‌ها به نوبه خود کاربردهای دیگری مانند تولید داده‌های برداری نیز دارد که بحث درباره موارد استفاده آن از محتوای این پایان‌نامه خارج است. در این فصل سعی بر آن داریم تا مروری بر تاریخچه اقدامات صورت گرفته در زمینه استخراج اطلاعات از نقشه‌ها انجام دهیم.

از آن جایی که نقشه‌ها حاوی اطلاعات متفاوتی هستند، اطلاعات قابل استخراج از نظر گرافیکی را به چهار دسته خطوط خیابان، تقاطع خیابان، ناحیه مورد نظر^۲ و متن تقسیم کرده و در هر مورد بخشی از کارهای انجام شده را بررسی می‌کنیم. به دلیل تمرکز این پایان‌نامه روی مبحث استخراج متون، بخش مرتبط با آن به تفصیل بررسی خواهد شد.

¹ Character

² Region of interest

۲-۲ استخراج خطوط خیابان

نقشه‌های جغرافیایی به آسانی در دسترس و شامل اطلاعات غنی جاده‌ای هستند. هر چند، تبدیل اطلاعات جاده به قالب برداری به علت کیفیت تصویر، خصیصه‌های همپوشان و ... اغلب امری چالش‌برانگیز است.

چیانگ و نوبلاک [۱۲] در سال ۲۰۱۱ میلادی رهیافتی کلی برای استخراج داده‌های برداری^۱ جاده از نقشه‌های جغرافیایی پیشنهاد کردند. آن‌ها یک رهیافت برداری‌سازی کلی ارائه می‌دهند که از خواص جغرافیایی عمومی جاده‌ها برای پردازش غیریکنواخت نقشه‌های شهری^۲ بهره‌برداری می‌کند؛ حال آن که نیازمند حداقل دخالت کاربر می‌باشد. آن‌ها کار خود را در چندین فاز انجام می‌دهند. در فاز اول هندسه جاده را استخراج می‌کنند. برای این کار ابتدا به استخراج بانظارت پیکسل‌های جاده می‌پردازند. در گام اول در راستای این استخراج بانظارت، ابتدا به کوانتیزه کردن رنگ با استفاده از الگوریتم‌های جابه‌جایی میانگین^۳، برش میانه^۴ و K میانگین می‌پردازند. سپس کاربر برچسب‌زنی‌های لازم را انجام می‌دهد و بعد از آن به شناسایی خودکار رنگ‌های جاده و استخراج پیکسل‌های جاده می‌پردازد که در آن از خاصیت خط مرکزی^۵ و خاصیت همسایگی استفاده شده است. گام دوم شناسایی خودکار خطوط مرکزی جاده و گام سوم استفاده از الگوریتم ردیابی الگوی موازی مسیر تکی است. فاز دوم ردیابی تقاطع خیابان است که در آن در اولین گام به تولید حباب‌های تقاطع خیابان^۶ برای برچسب‌زنی خطوط اعوجاج یافته، پرداخته می‌شود. شناسایی و ردیابی نماینده‌های خطوط خیابان، گام بعدی این فاز است و به‌روزرسانی الگوهای تقاطع خیابان، آخرین اقدام انجام گرفته در این

¹ Vector data

² Raster maps

³ Mean-Shift algorithm

⁴ Median-Cut

⁵ The Centerline Property

⁶ Road-intersection blobs

استفاده می‌شوند و شبکه‌های جاده‌ای معمولاً از یکدیگر قابل تشخیص هستند، می‌توان نقاط تقاطع را به صورت اثر انگشت نقشه‌ها^۱ در نظر گرفت. آن‌ها در روش خود، ابتدا از یک الگوریتم بخش‌بندی خودکار استفاده می‌کنند تا پیکسل‌های پس‌زمینه را بر پایه تفاوت در سطح روشنایی حذف نمایند. بعد از اینکه پیکسل‌های پیش‌زمینه را بدست آوردند که شامل اطلاعاتی از نقشه شهری اصلی است، لایه جاده را از سایر لایه‌ها جدا می‌کنند تا نقاط تقاطع جاده‌ها را استخراج کنند. قالب جاده به صورت اتوماتیک روی نقشه شهری ورودی بررسی و عرض خیابان ردیابی می‌شود. اگر لایه جاده در قالب دو خطی باشد، الگوی موازی برای خطوط جاده ردیابی می‌شود. آن‌ها سپس الگوریتم جداسازی متن/گرافیک را با استفاده از عملگرهای مورفولوژیکی^۲ اعمال کرده‌اند تا نویز را حذف کنند و لایه جاده را بازسازی نمایند. با لایه جاده استخراج شده، آن‌ها تنها نقاط برجسته^۳ را به عنوان نماینده تقاطع خیابان، بر پایه تغییرات سطح روشنایی هر پیکسل لایه جاده ردیابی می‌کنند. اتصال هر نقطه برجسته، سپس توسط بررسی ساختارهای خطی همسایگی روی لایه خیابان محاسبه می‌شود تا تعیین کند که آیا این نقطه تقاطع، یک نقطه تقاطع خیابان واقعی است یا نه. آن‌ها همچنین جهت تقاطع خیابان‌ها را در هر نقطه از تقاطع به عنوان نتیجه‌ای فرعی محاسبه می‌کنند. در فاز بعدی به ردیابی خودکار تقاطع خیابان‌ها می‌پردازند. نقشه‌های شهری معمولاً شامل بسیاری از اشیاء مانند نویسه‌ها، ساختمان‌ها، خیابان‌ها، رودخانه‌ها و حتی خطوط کانتور در نقشه‌های مکان‌نگاری^۴ هستند و این امری مهم است که بتوانیم در نقشه‌ها بین اشیاء و پس‌زمینه تمایز قائل شویم. آن‌ها نقشه‌های ورودی را به دو دسته اصلی، بسته به راهی که تولید می‌شوند، طبقه‌بندی می‌کنند. گام اول این مرحله طبقه‌بندی، بخش‌بندی خودکار است. چپانگ و همکارانش برای جداسازی پیش‌زمینه از نویزهای اضافه ورودی، از یک تکنیک عمومی به نام بخش‌بندی استفاده می‌کنند که از یک آستانه مشخص

¹ Maps' fingerprint

² Morphological operators

³ Salient points

⁴ Topographic maps

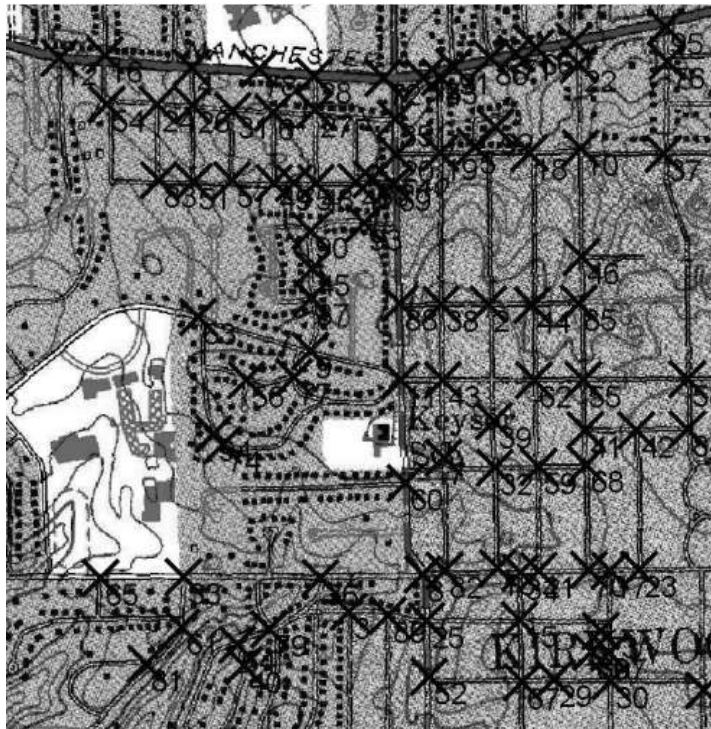
برای بخش بندی پیکسل‌های پیش‌زمینه و پس‌زمینه بهره می‌برد. گام دوم پیش‌پردازش^۱ استخراج لایه خیابان است. این بخش نقشه دودویی را که شامل لایه‌های مختلف اطلاعاتی به عنوان ورودی و خروجی لایه خیابان است، دریافت می‌کند. لایه‌های خیابان معمولاً دو خاصیت هندسی دارند که آن‌ها را از سایر لایه‌ها قابل تشخیص می‌کند: (۱) خطوط خیابان در طول یک فاصله کوچک (یعنی چندین متر در یک بلوک خیابان) مستقیم هستند (۲) برخلاف برجسب‌های لایه‌های ساختمان که ممکن است اشیای کوچک متصل بسیاری داشته باشند، خطوط خیابان به صورت شبکه‌های جاده‌ای به یکدیگر متصل هستند. لایه‌های خیابان معمولاً اشیای متصل کمی دارند یا نهایتاً یک شیء متصل بزرگ در آن‌ها دیده می‌شود که قسمتی از لایه خیابان است. برنامه جداسازی متن/گرافیک، سپس اشیای کوچکی را که با بخش‌هایی از خطوط خیابان در تماس هستند، حذف می‌کند. عملگرهای ریخت‌شناسی^۲ نیز خطوط جاده شکسته شده را مجدداً متصل می‌کنند. برای این کار به بررسی قالب خط دوگانه و ردیابی الگوی موازی می‌پردازند؛ اگر پیکسل هدف در خط جاده افقی یا عمودی باشد، می‌توان دو پیکسل پیش‌زمینه در جهت خط جاده را در فاصله عرض خیابان^۳ و پیکسل پیش‌زمینه دیگری را مطابق با خط جاده موازی در فاصله عرض خیابان یافت. جداسازی متن و گرافیک اقدام بعدی است که باید در این روش انجام گیرد. آن‌ها الگوریتم شرح داده شده در کار کائو و گروه همکارانش [۷] را اعمال می‌کنند. در این روش نمی‌توان از ایجاد خطوط شکسته جاده، اجتناب کرد. بنابراین باید در ابتدا اشیایی را که با آن‌ها تماس پیدا می‌کنند، حذف نمود و سپس با استفاده از عملگرهای ریخت‌شناسی این عناصر را متصل نمود. در ادامه، عملگرهای ریخت‌شناسی با استفاده از تبدیل hit or miss پیاده‌سازی می‌شوند. اگر ماسک‌ها با پیکسل‌های زیرین تطبیق داشته باشند، یک "hit" است و اگر ماسک با پیکسل‌های زیرین تطبیق نداشته باشد، یک "miss" است. در مرحله بعد از یک عملگر فرسایش کلی استفاده می‌شود تا خطوط خیابان را باریک کند و پیش از استفاده از

¹ Pre-processing

² Morphological operators

³ Road width

عملگرهای ریخت‌شناسی، جهات مشابه با جهات اصلی را شامل شود. می‌توان از یک عملگر باریک‌سازی در این قسمت استفاده نمود تا عرض خیابان‌ها در حدود یک پیکسل شود. گام بعدی ردیابی نمایندگی‌های تقاطع خیابان‌ها هستند. یک نقطه برجسته نقطه‌ای است که در آن بیش از یک خط با تانژانت‌های متفاوت به هم می‌رسند که به آن نقطه تقاطع^۱ می‌گویند. آخرین گام نیز فیلتر کردن تقاطع‌ها، استخراج اتصال تقاطع‌ها و جهات خیابان‌ها است. هنگامی که هیچ دانش اولیه‌ای روی نقشه‌ها وجود نداشته باشد، این رهیافت به ۹۲٪ دقت و ۷۷٪ فراخوانی می‌رسد [۱۳]. شکل (۲-۲) استخراج تقاطع خیابان توسط کار چیانگ و همکارانش در [۱۳] را نشان می‌دهد.



شکل ۲-۲: استخراج تقاطع خیابان توسط روش چیانگ و همکارانش (نقشه مکان‌نگاری USGS، ۲ m/pixel، سنت لوئیس) [۱۳]

در این رهیافت دو فرض کلی نیز وجود داشت: اول این که پیکسل‌های پس‌زمینه باید با استفاده از تفاوت سطح روشنایی از پیکسل‌های پیش‌زمینه قابل جداسازی باشند که این پیکسل‌های پیش‌زمینه

^۱ Crossing point

شامل جاده، ساختمان و سایر نمادها مانند لایه‌های نویسه هستند. دوم این که گرچه رهیافت فوق با هیچ دانش اولیه‌ای در مقیاس نقشه کار نمی‌کند، ممکن است در مورد نقشه‌های با وضوح کم، به درصد پایینی از دقت و فراخوانی برسد.

۲-۴ استخراج ناحیه مورد نظر^۱

در سال ۲۰۱۱ میلادی تبسم و شریف‌الدین دو روش برای استخراج ناحیه مورد نظر (ROI) در نقشه جغرافیایی پیشنهاد کردند [۱]. ناحیه مورد نظر روی شبکه جاده‌ای و آبراه متمرکز شده است. روش اول بر پایه رنگ است که در آن خوشه‌بندی K میانگین را اعمال می‌کنند. تصویر رنگی شامل چندین شیء همپوشان با رنگ، اندازه و شکل متفاوت است. فرآیند استخراج شبکه حمل و نقل با ماکزیمم کردن خیابان‌های متصل و استخراج آبراه‌ها انجام می‌گیرد. بعد از این که تصویر به سیستم وارد شد، از فضای رنگ RGB به Lab برده می‌شود. سپس خوشه‌بندی K میانگین در این فضای جدید انجام می‌شود و در ادامه اجزای متصل تصویر، برچسب زده می‌شوند. در گام بعدی شبکه جاده‌ای مورد پردازش قرار می‌گیرد. به این ترتیب که از یک آستانه‌گذاری^۲ مناسب برای حذف نویز استفاده می‌شود و سپس جاده‌ها بازسازی می‌شوند. در مرحله بعد، شبکه جاده‌ای متصلی که دارای بیشترین اندازه است، شناسایی می‌شود. روش دیگر استفاده از انطباق الگو^۳ است که جامع‌تر از تکنیک مطرح شده پیشین است. در این روش برای تطبیق الگو بر حسب یک ویژگی خاص که می‌خواهیم آن را ردیابی کنیم، از یک ماسک پیچش^۴ (الگو) استفاده می‌شود. خروجی پیچش در جایی که ساختار تصویر منطبق با ساختار ماسک است و مقادیر بزرگ تصویر در مقادیر بزرگ ماسک ضرب می‌شوند، در بالاترین مکان خواهد بود. این روش شبکه‌ها و آبراه‌ها را از نقشه استخراج می‌کند. روش اول برای نقشه‌هایی که تغییرات رنگ در آن‌ها کم است و روش دوم برای نواحی‌ای که باریک‌تر از حدی

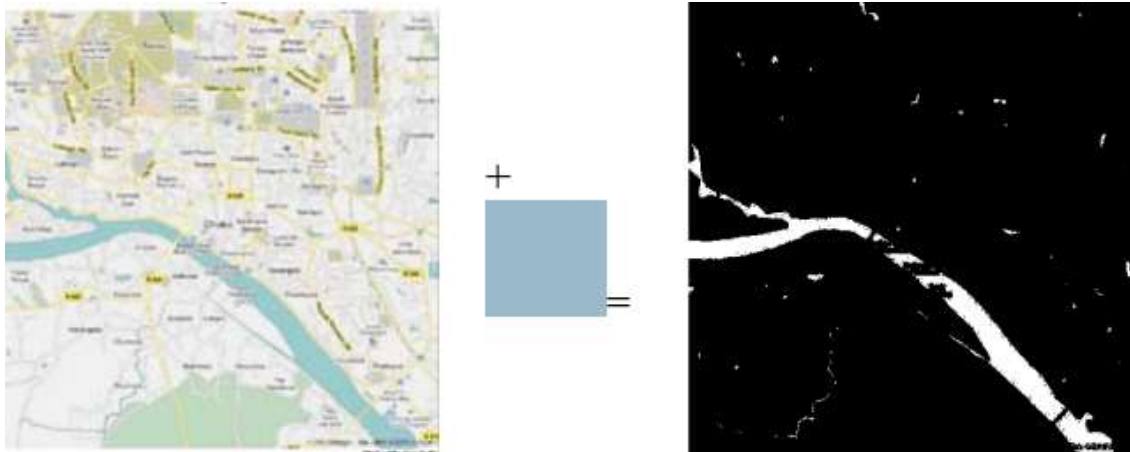
¹ Region Of Interest

² Thresholding

³ Template based matching

⁴ Convolution

مشخص هستند، مناسب نیستند و پاسخ مطلوبی از اعمال آن‌ها حاصل نمی‌شود. شکل (۲-۳) مثالی از استخراج آبراه را روی یک نقشه توسط روش تبسم و شریف الدین نشان می‌دهد.



شکل ۲-۳: استخراج آبراه با استفاده از الگو در روش تبسم و شریف الدین [۱]

۲-۵ استخراج متن

گوتو و آسو یک روش محلی خطی بر پایه تکنیک شناسایی جهت خطوط متنی در نوشته‌های انگلیسی و چینی ارائه کرده‌اند. محدودیت اصلی روش آن‌ها این است که نمی‌تواند از عهده متون دارای نویسه‌های با اندازه‌های مختلف برآید [۱۴].

فلچر و کاستوری [۱۵] در مقاله خود توسعه و پیاده‌سازی یک الگوریتم جدید را برای جداسازی خودکار یک رشته متنی که مستقل از تغییرات قلم متن، اندازه و جهت رشته‌ها است، مورد بررسی قرار داده‌اند. کارهایی که در این مقاله صورت گرفته، در ادامه به ترتیب بیان شده است: در ابتدا اجزای متصل^۱ تصویر تولید شده، سپس از یک فیلتر مساحت/نسبت^۲ استفاده می‌شود. گروه‌بندی اجزایی که در یک خط قرار گرفته‌اند و گروه‌بندی منطقی رشته‌ها به کلمات و عبارات، اقدامات بعدی هستند که به ترتیب باید انجام گیرند؛ در نهایت جداسازی رشته متنی صورت می‌پذیرد. الگوریتم دو تصویر تولید

^۱ Connected components

^۲ Area/ ratio filter

می‌کند؛ یکی شامل متن و دیگری شامل گرافیک. این تصاویر ممکن است توسط سیستم‌های نویسه‌خوان و شناسایی گرافیک، مطلوب تشخیص داده شوند. در این مقاله از تبدیل هاف^۱ برای گروه‌بندی نویسه‌ها استفاده گردیده و رشته‌های متنی استخراج شده‌اند. از آنجایی که تبدیل هاف تنها خطوط مستقیم را ردیابی می‌کند، روش آن‌ها قابل اعمال به رشته‌های خمیده نیست [۵].

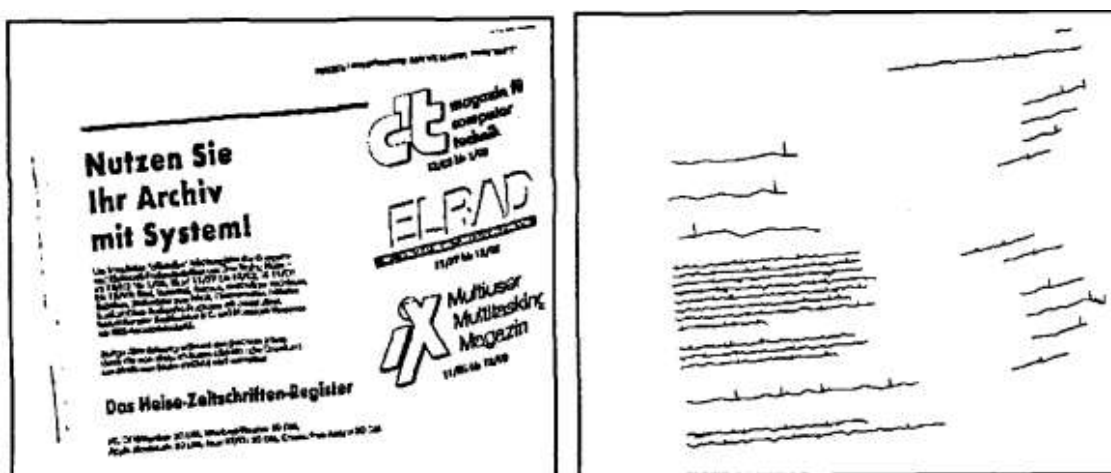
چن و ونگ یک روش کامل برای استخراج و تشخیص رشته‌های عددی دستی روی نقشه‌ها مطرح نموده‌اند [۱۶]. الگوریتم استخراج نویسه می‌تواند نویسه‌های کج و نویسه‌هایی را که با هم تماس دارند، به اجزای منحصر به فردش بخش‌بندی کند. الگوریتم شناسایی بر پایه ویژگی نیز می‌تواند نویسه‌های عددی با هر اندازه، موقعیت و جهت را تشخیص دهد. در روش مطرح شده ابتدا نویسه‌ها استخراج می‌شوند، سپس عملیات تشخیص صورت می‌گیرد؛ در این تشخیص سوراخ‌ها، نقاط مورد نظر و محل‌های خطی، اشکال متقارن و نقاط تقاطع شناسایی می‌شوند. در تکنیک مطرح شده آن‌ها، تبدیل هاف و مجموعه‌ای از مشخصه‌های مستقل قلم و اندازه نیز جهت تشخیص رشته‌های عددی در نقشه‌ها به کار رفته‌اند، اما این روش را نمی‌توان در مورد نویسه‌های الفبایی به کار برد.

در روشی دیگر که توسط لیچر و هونز [۱۷] برای استخراج و بخش‌بندی پیشنهاد شده است، در ابتدا لنگرهای خطی در تصاویر اسناد ایجاد می‌شوند و سپس خطوط متنی با بسط این لنگرهای^۲ خطی تولید می‌شوند. این روش مستقل از جهت متون است و می‌تواند از عهده متون با اندازه‌ها و مدل‌های مختلف قلم برآید. سه مرحله اصلی این پردازش، تولید عناصر متصل، تحلیل همسایگی‌ها، تولید خطوط متنی و بلوک‌هاست. بعد از تحلیل، عناصر متصل به گروه‌های متنی و غیرمتنی تقسیم می‌شوند. سپس عناصر متنی به نویسه‌ها، کلمات، خطوط و بلوک‌ها گروه‌بندی می‌شوند. عناصر غیرمتنی هم به عنوان بلوک‌های غیرمتنی تقسیم‌بندی می‌شوند. مثالی از استخراج خطوط متنی

¹ Hough transform

² Anchor

توسط روش لیچر و هونز در شکل (۲-۴) نشان داده شده است.



شکل ۲-۴: استخراج خطوط متنی توسط روش لیچر و هونز [۱۴]

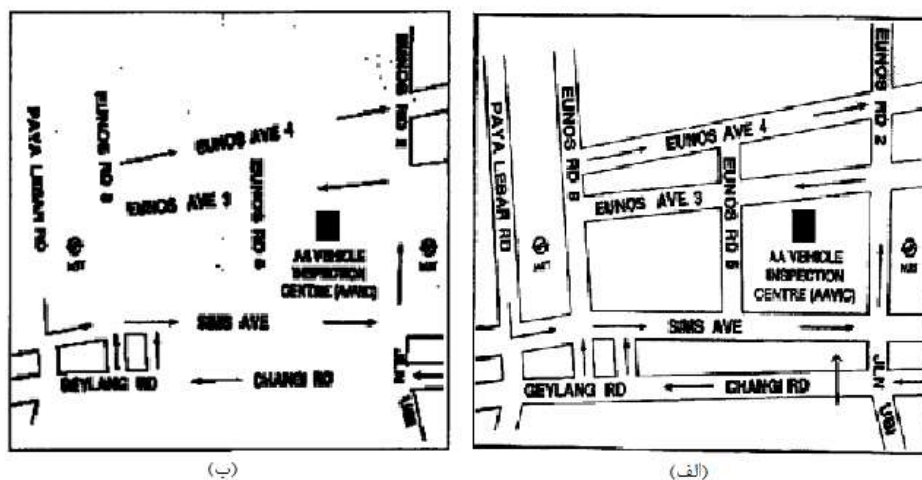
سینها و چادهوری [۱۸] در تکنیک پیشنهادی خود بر روی خطوط هندی "بنگلا"^۱ و "دوانگاری"^۲ تمرکز کردند. در این دو نوع خط هنگامی که تعداد دو یا بیشتری از نویسه‌ها در کنار هم قرار می‌گیرند و یک کلمه را تشکیل می‌دهند، یک خط افقی به نام "Matra" یا "Shirorekha" با قسمت بالایی آن‌ها تماس پیدا می‌کند؛ اما چون این روش بر پایه خصیصه Matra عمل می‌کند، بر روی خطوط لاتین قابل اجرا نیست؛ زیرا این ویژگی در خطوط لاتین وجود ندارد.

سینها و چادهوری در تکنیکی دیگر [۱۸] در سال ۲۰۰۳ میلادی روشی را بر پایه رهیافت پایین به بالا^۳ برای شناسایی خطوط متنی چندجهته انگلیسی ارائه کردند که در ابتدا عناصر متصل را برچسب می‌زند، سپس آن‌ها را به گروه‌های کلمات گروه‌بندی می‌کند. این روش قادر است متون را به سیاه روی پس‌زمینه سفید یا سفید روی پس‌زمینه سیاه بخش‌بندی کند. سپس از تکنیک‌های استخراج خط استفاده می‌شود. رهیافت شناسایی خط بر پایه ردیابی خط کرسی بنا شده است. اگر

¹ Bangla
² Devangari
³ Bottom-up approach

یک خط متن به صورت افقی باشد، استخراج خط کرسی آن بسیار ساده است، زیرا خط فرضی از میان ماکزیمم تعداد پایین‌ترین نقاط از عناصر خط کرسی عبور می‌کند. اما اگر یک خط متن به صورت افقی نباشد، ردیابی خط کرسی دشوار خواهد بود و برای آن از یکی از دو روش زیر استفاده می‌شود: الف) ردیابی خط کرسی برای گروه شامل سه عنصر ب) ردیابی خط کرسی برای یک گروه با چهار عنصر یا بیشتر. در نهایت متون به صورت سیاه روی صفحه‌ای سفید نمایش داده می‌شوند. دقت کلی این روش در این نوع تشخیص ۹۸/۳٪ است. بیشترین خطاها هنگامی رخ می‌دهند که یک یا دو نویسه پایین‌رونده و بالارونده وجود داشته باشد. بنابراین این روش برای زبان‌هایی مانند فارسی که دارای نویسه‌های بالارونده و پایین‌رونده زیادی هستند، مناسب نخواهد بود. یکی از مهم‌ترین مزایای این روش نیز انعطاف‌پذیری آن است که از قلم و اندازه نویسه‌های متون مستقل است.

تن و ان‌جی یک سیستم را با استفاده از اهرام^۱ برای استخراج رشته‌های متنی توسعه داده‌اند. روش آن‌ها این فرض را در نظر می‌گیرد که متون با گرافیک‌ها در تماس نیستند و با آن‌ها همپوشانی ندارند. بنابراین روش خوب و جامعی به شمار نمی‌رود [۱۹]. شکل (۲-۵) نتیجه اعمال هرم در روش تن و ان‌جی را روی یک نقشه نشان می‌دهد.



شکل ۲-۵: اعمال روش هرم در کار تن و ان‌جی (الف) نقشه اصلی (ب) متون استخراج شده [۱۹]

^۱ Pyramid

ولازکز و لواچکین در سال ۲۰۰۴ میلادی روشی را برای جداسازی و تشخیص متن در نقشه‌های مکان‌نگاری رنگی بر پایه ردیابی رشته‌های مستقیم پیشنهاد کردند [۲۰]. آن‌ها ابتدا نقشه را به همه رشته‌ها شامل آن‌هایی که با سایر نمادها در تماس هستند، سرکش‌ها و نویسه‌ها بخش‌بندی کرده‌اند. بخش‌بندی کلی نویسه‌های نقشه به این صورت انجام می‌شود که همه المان‌ها باید با مختصات و خصیصه‌ها بازبینی شوند و سپس به لایه موضوعی مربوطه فرستاده شوند. سپس تشخیص بر پایه OCR با شبکه عصبی مصنوعی^۱ (ANN) برای تعریف مختصات، اندازه و جهت رشته‌های نویسه‌های الفبایی عددی در هر مورد ارائه شده در نقشه، اعمال شده است. در مرحله اول باید تصویر رنگی به یک تصویر دودویی تبدیل شود و نویسه‌های دارای تماس و همپوشانی جداسازی و شناسایی شوند. بعد از آن چهار خط مستقیم یا تعدادی از خمیدگی‌ها که به عنوان یک تابع اولیه توسط ANN سازمان‌دهی شده‌اند، محاسبه می‌شوند. سپس نویسه‌ها برون‌یابی^۲ می‌شوند تا آن نمادهایی که متصل هستند، جدا شوند. در نهایت نویسه‌های جدا شده، مجدداً به ANN وارد می‌شوند تا نهایتاً شناسایی شوند. تکنیک آن‌ها با نویسه‌های در اندازه‌های مختلف، انواع قلم‌ها و جهت‌ها سر و کار دارد؛ اما نمی‌تواند روی رشته‌های خمیده کار کند.

در [۹] دزیلیگنی و همکارانش روشی را برای تشخیص تغییرناپذیر با چرخش بودن رشته‌های نویسه روی نقشه‌ها، پیشنهاد کردند. برای حصول نتیجه‌ای خوب برای GIS باید لایه‌های مختلف از هم جدا شوند. همان‌طور که لایه‌های متفاوت در طول اجرای این برنامه استخراج می‌شوند، باید یکی یکی از نقشه اصلی کم شوند؛ بنابراین در هر مرحله نقشه‌ای ساده‌تر برای لایه‌های باقیمانده ایجاد می‌شود که نیازمند به کار بردن قوانین بین لایه‌ای سطح بالایی است. برای GIS ضروری است که پیوندهای توپولوژیکی بین اشیای لایه‌های مختلف به صورت اتوماتیک خلق شوند. روش‌های تشخیص

^۱ Artificial Neural Network

^۲ Extrapolate

استفاده شده، بر پایه تطبیق مستقیم هستند. این روش‌ها سریع نبوده و برای تشخیص اسناد دارای متون تک‌قلم مناسب هستند. نتایج این کار برای کاربردهای صنعتی کافی به نظر می‌رسد؛ اما نیاز دارد تا روی مثال‌های بیشتری پاسخ مناسبی بدهد. علاوه بر این، زمان آن نیز در مقایسه با نرم‌افزار تجاری OCR اندکی طولانی‌تر است و نتایج آن برای کاربردهای عملی مناسب نیست.

در [۷] تکنیکی برای جداسازی متن و گرافیک در نقشه‌ها پیشنهاد شده است. از آن‌جایی که اجزای تشکیل‌دهنده نویسه‌ها معمولاً بخش کوتاهی در مقایسه با گرافیک‌ها هستند، اجزای تشکیل‌دهنده خط با ویژگی‌های عرض خط ترکیب می‌شوند تا بخش‌هایی که توسط تقاطع‌ها بازسازی می‌شوند، تجزیه گردند. در ابتدا به پیش‌پردازش و خوشه‌بندی اولیه پرداخته می‌شود. پیش‌پردازش شامل سه مرحله است: جداسازی زیرلایه، حذف عناصر گرافیکی توپر و حذف خطوط تیره. در گام بعدی نویسه‌های تماسی ردیابی و بازشناسی می‌شوند. اجزای تشکیل‌دهنده عناصری که به نویسه‌ها برخورد می‌کنند، معمولاً در مقایسه با گرافیک‌ها کوچک‌ترند؛ از همین رو از باریک‌سازی عناصر استفاده شده است. پس از باریک‌سازی، فرآیند تجزیه اجرا می‌شود تا نقاط تقاطع به آسانی در تصویر گرافیکی باریک شده، ردیابی شوند. بعد از تجزیه، از فیلتر اندازه در خوشه‌بندی اولیه استفاده می‌شود تا اجزای تشکیل‌دهنده متن از خطوط گرافیکی جدا شده و در انتها به خوشه‌بندی نویسه‌ها پرداخته شود. دقت تشخیص نویسه‌های تکی و تماسی بعد از استفاده از این تکنیک به ترتیب تا ۹۳/۴٪ و ۶۴/۸٪ بهبود یافته است.

روی و گروه همکارانش [۲۱] روشی را برای استخراج متون لاتین چندجهته بر پایه اطلاعات پیش‌زمینه و پس‌زمینه مطرح کردند. بیشتر محققان توجهشان را تنها روی بخش‌های پیش‌زمینه متمرکز می‌کنند و بخش پس‌زمینه را نادیده می‌گیرند (در اسناد نوشتاری، بخش پس‌زمینه آن نواحی سفیدی از تصویر باینری هستند که پیش‌زمینه سیاه را کاملاً احاطه می‌کنند). اطلاعات پس‌زمینه نقش مهمی را در روش پیشنهادی ایفا می‌کنند و الگوریتم را به سوی استخراج خطوط مجزا از اسناد

شامل خطوط چندجهته و خمیده هدایت می‌کند. برای استخراج بخش پس‌زمینه، از مفهوم مخزن آب^۱ استفاده شده است. مخزن آب یک تشبیه است تا ناحیه گودال از یک عنصر را شرح بدهد. اصل مخزن آب به صورت زیر است: اگر آب از یک طرف یک عنصر بریزد، نواحی گود از بخش پس‌زمینه عنصر که آب در آن‌جا ذخیره می‌شود، به عنوان مخزن عنصر در نظر گرفته می‌شود. استفاده از اطلاعات پس‌زمینه و محاسبات آن توسط مفهوم مخزن آب، یکی از خصیصه‌های منحصر به فرد از تکنیک پیشنهادی آن‌ها است. در طرح پیشنهادی در ابتدا عناصر مجزا ردیابی شده‌اند و با استفاده از فاصله بین عنصری آن‌ها، اندازه و اطلاعات مکانی به سه دسته نویسه، گروه‌بندی می‌شوند. با یکی کردن این سه دسته یک گراف تخمینی ایجاد می‌شود تا دسته‌های بزرگتری را شامل شود. با استفاده از اطلاعات پس‌زمینه بین نویسه‌ای جهات نویسه‌های اضافی از یک دسته بزرگتر انتخاب می‌شود و بر پایه این جهات، دو ناحیه کاندیدا از خوشه شکل می‌گیرد. در نهایت با کمک این نواحی کاندیدا، خطوط مجزا استخراج می‌شوند. عیب این روش این است که در آن فرض شده است که فاصله بین دو خط از یک سند، بزرگتر از فاصله بین نویسه‌های کلمات است. اما در واقع گاهی فاصله بین دو کلمه از دو متن متفاوت بسیار کوچک است و لذا روش پیشنهادی در این موارد خطاهایی را ایجاد می‌کند. دومین عیب آن این است که اگر نویسه‌ها شکسته باشند، روش روی آن‌ها کار نمی‌کند و بخش شکسته نمی‌تواند در پردازش متصل گردد. در این گونه موارد انتخاب عنصر همسایگی نیز کمکی نمی‌کند و استفاده از مفهوم مخزن آب نیز نمی‌تواند ناحیه نماینده را به طور مناسب مشخص کند و خطا اتفاق می‌افتد. اشکال سوم این روش این است که اگر عناصر همپوشان زیادی با هم در یک کلمه موجود باشند، روش پیشنهادی ممکن است به خوبی کار نکند. از جمله برتری‌های این روش به برخی از روش‌های موجود نیز می‌توان به انعطاف‌پذیری آن اشاره کرد؛ زیرا این طرح مستقل از قلم، اندازه، مدل و جهت متون است. شکل (۲-۳) نتایج استخراج خط متنی را توسط روش روی و همکارانش در

¹ Water reservoir concept

یک روزنامه، سند معنایی و یک نقشه نشان می‌دهد.

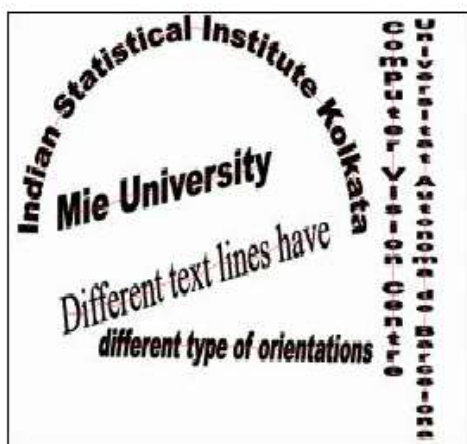
در تحقیقی دیگر، روی و گروه همکارانش تکنیکی را برای بخش‌بندی نویسه‌های تماسی دارای چند اندازه و جهت با استفاده از برنامه‌ریزی پویا پیشنهاد کرده‌اند [۶]. هنگامی که دو یا چند نویسه با یکدیگر تماس پیدا می‌کنند، حفره‌ای بزرگ در قسمت پس‌زمینه ایجاد می‌شود. با استفاده از اطلاعات بدنه محدب^۱ از بعضی نقاط اولیه برای بخش‌بندی رشته تماسی به بخش‌های اولیه استفاده می‌شود. این بخش‌های اولیه ایجاد شده برای رسیدن به بهینه‌ترین بخش‌بندی، یکی می‌شوند. بعد از این مرحله، برنامه‌ریزی پویا با استفاده از شباهت کلی نویسه‌ها به عنوان تابع هدف اعمال می‌شود. برای پیدا کردن شباهت نویسه‌ها از کلاسه‌بند SVM^۲ استفاده شده است. برای در نظر گرفتن رشته‌های همپوشان چندجهته، خصیصه‌هایی در SVM به کار گرفته شده‌اند که با جهت‌گیری نویسه تغییر نمی‌کنند. از رهیافت بر پایه حلقه دایروی و حلقه محدب همراه با اطلاعات زاویه‌ای پیکسل‌های کانتور نویسه‌ها، برای ایجاد خصیصه تغییرناپذیر با چرخش استفاده شده است. این الگوریتم برای مراقبت از رشته نویسه در محیط نویزی کارآمد است. روش پیشنهادی در مورد بخش‌بندی رشته‌های تماسی دارای چندین اندازه و جهت نیز نتایج خوبی را بدست می‌دهد. شکل (۲-۶) نتایج استخراج خطوط متنی را در کار روی و همکارانش نشان می‌دهد.

لی و گروه همکارانش در [۳] تکنیک‌هایی برای تشخیص متون از نقشه‌های دودویی ارائه داده‌اند. این قبیل تکنیک‌ها که روی نقشه‌های دودویی کار می‌کنند، نمی‌توانند نقشه‌های اسکن شده را به سادگی پردازش کنند. چون نقشه‌های اسکن شده، اغلب در نتیجه فشرده‌سازی و نویز ناشی از فرآیند اسکن خراب می‌شوند، تولید نقشه دودویی با اشکال مواجه شده و پردازش آن نیز فرآیندی زمان‌بر است [۵]. در [۳] لی و همکارانش در مورد برخی از پیشرفت‌های حاصل شده در زمینه روش‌های

¹ Convex Hull

² Support Vector Machine

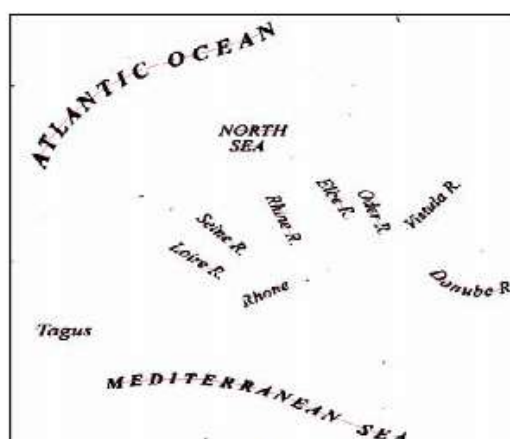
جداسازی متن و گرافیک برای مناسب ساختن این روش‌ها در نقشه‌ها و در مورد روشی برای جداسازی متن از گرافیک با استفاده از روش گروه‌بندی نویسه‌ها بحث می‌کنند. در مقاله آن‌ها، ابتدا جداسازی توسط آنالیز اجزای متصل صورت می‌گیرد. سپس به ترتیب از بخش‌بندی رنگ، جداسازی اجزای طویل مستقیم و بخش‌بندی اسکلت استفاده می‌شود. پس از آن نویسه‌ها به گروه‌های کلمات دسته‌بندی می‌شوند.



(ب)



(الف)



(ج)

شکل ۲-۶: نتایج استخراج خط متنی توسط روش روی و همکارانش (روی الف) یک روزنامه (ب) یک سند معنایی (ج) یک نقشه [۲۱]

در [۳] لی و گروه همکارانش تکنیکی دیگر برای استخراج متون یکپارچه^۱ و خطوط از یک نقشه مکان‌نگاری ارائه کرده‌اند. از آنجایی که در نقشه USGS^۲ اسامی و خطوط خیابان‌ها همه به صورت سیاه چاپ شده‌اند، جداسازی لایه‌های سیاه با یک فیلتر رنگی نقطه‌آغازی برای پردازش تصویر مهیا می‌کند. لایه متنی سیاه نه تنها شامل اسامی خیابان‌ها است، بلکه شامل مقدار ارتفاع و اسامی همسایگی‌ها، حوزه‌های سیاسی، مدارس، بیمارستان‌ها، سایر مناطق ساختمانی، مشخصه‌های فیزیوگرافیک و مکان‌نگاری نیز هست. آن‌ها سپس یک طبقه‌بندی اولیه روی هر جزء بر طبق خصیصه‌های هندسی و چگالی پیکسل‌های سیاه بر روی خطوط، نویسه‌ها و نمادها انجام داده‌اند. خطوط با استفاده از نرم‌افزار Arcscan برداری شده‌اند و با استفاده از عرض خط^۳، طول خط^۴ و معیار جفت خط^۵ تصحیح گردیده‌اند و سرانجام محدودیت‌هایی روی آن‌ها اعمال شده تا پوشش خیابان‌ها را به صورت بلوک‌های شهری شکل دهند. سپس پوشش دو خطی خیابان به یک خط مرکزی خیابان کاهش می‌یابد. در گام بعدی نویسه‌ها به صورت "جعبه‌های کلمات" گروه‌بندی می‌شوند. روش پیشنهادی آن‌ها برای استخراج متن و خطوط، نیازمند قرار دادن دقیق یک جعبه روی رشته متنی و مشخص کردن توابع بردارهای خطوط بیرونی روی جعبه است. نتیجه برونیابی این بردارهای درون جعبه به یک برنامه نویسه‌خوان نوری (OCR) تک‌قلم آزمایشی منتقل می‌شود که به طور خاص جهت قلم استفاده شده در برچسب خیابان‌ها آموزش داده شده است. در این جا دو نوع پردازش انجام می‌شود. پردازش اولیه هر پیکسل را فقط یک بار بررسی می‌کند. سپس اجزای اصلی کاندید شده برای اسامی خیابان‌ها و پوشش آن‌ها مشخص می‌شوند. پردازش محلی شامل دو مرحله است: گروه‌بندی نویسه‌ها و بهبود ساختار خط در حین گروه‌بندی نویسه‌ها که بر پایه گراف همسایگی استوار است. در این جا رابطه فضایی بین اجزای متصل مورد بررسی قرار می‌گیرد. در مرحله دوم هدف بهبود پوشش

¹ Integrated

² United States Geological Survey

³ Line width

⁴ Line length

⁵ Line-pairing criteria

خط خیابان‌هایی است که به دلیل همپوشانی با متن قطع شده‌اند. در مرحله بعد باید از OCR که بر پایه تطبیق الگو استوار است، استفاده کرد تا علاوه بر تبدیل نقشه بیتی^۱ جعبه‌های کلمات به برجسب‌های الفبایی عددی ASCII، اجازه بهبود بیشتری در بخش‌بندی خط و متن را نیز بدهد.

در [۲۲] هانگ و لین سیستمی را برای خواندن اتوماتیک نقشه‌های شهری تایوان پیشنهاد داده‌اند. آن‌ها از یک الگوریتم پردازش مخلوط^۲ (رهیافت پایین به بالا و بالا به پایین^۳) استفاده می‌کنند تا نقشه‌ها را تفسیر کنند. این رهیافت که توسط یک دانش اولیه هدایت می‌شود، تناقض بین اشیای استخراج شده را حل می‌کند. آن‌ها از یک نمایش دارای چهار سطح، برای تفسیر نقشه‌های شهری رنگی استفاده می‌کنند که این سطوح به شرح زیرند: (۱) سطح پیکسل (۲) سطح برداری (۳) سطح رشته‌های بلوک‌ها، جاده‌ها، نمادها و نویسه‌ها (۴) سطوح معنایی. دو راه ارتباطی بین هر دو سطح همسایگی وجود دارد. رهیافت بالا به پایین از سیستمی بر پایه "مدلسازی" ایجاد می‌گردد که نیاز به یک شاهد برای تشخیص یک شیء با یک نمایش ذهنی از آن به عنوان یک اولویت دارد. سیستم عملیات زیر را مدیریت می‌کند: الف) ساختن ارتباط بین پیکسل‌ها برای شکل‌دهی بردارهای خط و نویسه ب) گروه‌بندی بردارهای خط جاده‌ها و نویسه‌ها به رشته‌های نویسه ج) تشخیص نمادها و رشته‌های نویسه د) پیدا کردن پیوندهای معنایی بین جاده‌ها و رشته‌های نویسه‌ها یا بین نمادها و رشته‌های نویسه. رهیافت پایین به بالا به این شرح است که چون یک شیء از چندین اولیه در سطح پایین‌تر ساخته می‌شود، رهیافت پایین به بالا به رشته پردازش‌هایی مرتبط است که از قانون‌های مرتبط با گرافیک‌های اولیه با سطوح کمتر استفاده می‌کنند تا یک تفسیر صحیح برای شیء سطح بالا داشته باشد. گام اول این رهیافت بخش‌بندی رنگ است که برای جداسازی هر لایه رنگی از نقشه

¹ Bitmap

² Mix- processing algorithm

³ Top-down approach

اسکن شده شهری مورد استفاده قرار می‌گیرد و شامل تیز کردن^۱، هموار کردن^۲ و شکافتن^۳ است. گام دوم کم کردن تصویر چمنزار است. در نقشه‌های مورد بررسی هانگ و لین، چمنزار به رنگ سبز مخلوط با نقطه‌های سیاه بود که این نقطه‌ها بعضی نویزها را در پردازش لایه سیاه ایجاد می‌کردند. برای این پردازش از عملگر گسترش برای پر کردن سوراخ‌ها استفاده می‌شود و سپس لایه سبز از تصویر لایه سیاه کم می‌شود. گام سوم یافتن جهت نقشه است که اطلاعات مهمی برای استخراج و تشخیص نویسه‌ها و نمادها دارد و در آن از یک تبدیل هاف برای پیدا کردن خطوط مستقیمی که از طول کل نقشه می‌گذرند و هم فاصله‌اند، استفاده شده است. چهارمین گام استخراج و تشخیص نماد است که در آن تطبیق بر پایه یافتن نمادها به کار برده شده است؛ بنابراین معیار فاصله به کار رفته هاسدروف^۴ است که برای مقایسه تصویر نماد با مدل‌های نمادین از پیش ذخیره شده استفاده می‌شود. گام پنجم استخراج رشته نویسه مرتبط با هر نماد است که با استفاده از آن می‌توانیم رشته‌های نویسه‌ای را پیدا کنیم که معنی نمادها را شرح دهند. اگر رشته‌های نویسه به دقت از تصویر نقشه استخراج شوند، می‌توانند توسط نرم‌افزار تجاری OCR شناسایی شوند. گام ششم استخراج جاده است. برای استخراج جاده نیاز است تا موقعیت جاده‌ها در ابتدا شناسایی شوند و سپس رشته‌های نویسه‌های مرتبط با مفروضات زیر پیدا شوند. (۱) نویزهای به جای مانده از باقیمانده نمادها باید پیش از اعمال روش استخراج خطوط جاده حذف شوند. (۲) سفید و زرد رنگ‌های خاصی هستند که برای نمایش جاده‌ها در نقشه‌ها استفاده می‌شوند. (۳) رشته‌های نویسه مرتبط با جاده‌ها معمولاً روی نوارهای خطوط جاده قرار داده شده‌اند. فرآیند استخراج خیابان ممکن است شامل ۵ مرحله باشد: الف) استخراج لایه رنگی (ب) حذف شکاف‌ها (ج) یافتن اسکلت جاده‌ها (د) یافتن اتصالات (ه) ردیابی جاده‌ها. گام هفتم استخراج رشته نویسه است که در آن یک پنجره تعریف شده است که اندازه آن کمی

¹ Sharpening

² Smoothing

³ Splitting

⁴ Hausdroff

بزرگتر از اندازه نویسه است و سپس این پنجره همراه با جاده از روی نقشه حذف می‌شود. در حین ردیابی جاده، آن‌ها پنجره را حرکت می‌دهند و چگالی پیکسل سیاه را محاسبه می‌کنند. اگر چگالی پیکسل سیاه درون پنجره بزرگتر از آستانه مشخصی باشد، در آنجا یک نویسه وجود دارد. در مرحله بعدی باید پایداری اجرا شود. اگر بیشترین تعداد عناصر نشان داده شوند، در ابتدا پایداری درونی و داخلی یک شیء توسط یک چسبندگی ذاتی توصیف می‌شود. پایداری درونی تنها با در نظر گرفتن هویت شیء، هنگام نادیده گرفتن اطلاعات مربوط به هویت همسایگی‌ها سر و کار دارد. سپس پایداری بیرونی و کلی که همسایگی اشیاء را در نظر می‌گیرد، استفاده می‌شود. در این تکنیک درصد ردیابی اشیاء از ۷۰٪ تا ۹۰٪ افزایش می‌یابد [۲۲]. این سیستم نیازمند این است که در فرآیند بالا به پایین استخراج اشیاء تعمیم یابد تا اشیاء قابل اطمینانی بدست بیاید و فرآیندهای منطقی بیشتری را جمع کند تا اشیای استخراج شده با موفقیت اصلاح شوند.

پودرکس و همکارانش [۲۳] رهیافتی خودکار برای استخراج و تشخیص اسامی شهرها و مکان‌ها ارائه کردند که مبتنی بر بخش‌بندی تصویر و تحلیل اجزای متصل است. مراحل مختلف اعمال فیلتر، پایداری رشته‌ها و نویسه‌های محتمل را تضمین می‌کند. نواحی متنی شناسایی شده به عنوان ورودی OCR استفاده می‌شوند و کلمات تشخیص داده شده، تحلیل و اصلاح می‌گردند. اقداماتی که باید به ترتیب صورت بگیرد، استخراج و تشخیص اسامی شهرها، بخش‌بندی نقشه، تحلیل اجزا، تحلیل رشته، OCR، پس‌پردازش و یکی کردن رشته‌ها است. مزیت مقاله این است که در آن هیچ فرض اولیه‌ای در مورد قلم، اندازه و جهت وجود ندارد؛ اما اشکال این رهیافت در این است که در آن روی نقشه‌های شهری و استخراج اسامی خیابان‌ها کار نشده است.

چیانگ و نوبلاک در [۵] روشی جامع برای تشخیص متن در نقشه‌ها ارائه کرده‌اند که می‌توان این رهیافت را جامع‌ترین رهیافت در زمینه متون لاتین دانست. رهیافت ارائه شده در این مقاله نیاز به

تلاش کم‌کاربر جهت تشخیص برجسب‌های متنی در نقشه‌های غیریکنواخت دارد. دو مرحله اصلی این رهیافت (۱) استخراج لایه متنی و (۲) تشخیص برجسب‌های متنی است. مرحله استخراج لایه متنی یک تکنیک بانظارت است که نواحی نمونه متنی را تحلیل می‌کند تا رنگ‌هایی را شناسایی کند که در یک نقشه با جداسازی لایه‌های متنی منحصر به فرد، متون را نمایش می‌دهند (یعنی مجموعه‌ای از پیکسل‌های متنی با رنگ یکسان). مرحله شناسایی برجسب متنی بر پایه کار پردازش آن‌ها در [۱۲] صورت می‌گیرد. این مرحله متون چندجهته، دارای اندازه‌های مختلف و نیز دارای خمیدگی را بررسی می‌کند و نیاز به آموزش قلم خاصی ندارد. در استخراج لایه متنی در ابتدا به کوانتیزاسیون رنگ پرداخته می‌شود. نقشه‌های شهری معمولاً شامل تعداد زیادی رنگ به علت فرآیند اسکن یا فشردگی هستند. در این مرحله تعداد رنگ‌های درون نقشه به تعداد محدودی از رنگ‌ها کاهش می‌یابد. در مرحله برجسب‌زنی، کاربر نقشه کوانتیزه را برجسب می‌زند. یک ناحیه مستطیلی انتخاب می‌کند که حداقل باید یک رشته با دو نویسه را پوشش دهد. اگر رشته غیرافقی باشد، کاربر می‌تواند آن را بچرخاند. در گام بعدی رنگ‌های متنی با استفاده از مثال‌های متنی به صورت خودکار شناسایی می‌شوند. ابتدا این مثال‌های متنی به مجموعه‌ای از تصاویر تجزیه می‌شوند؛ طوری که هر عکس تجزیه شده تنها شامل یک رنگ از مثال متنی باشد. در مرحله بعد عملگر ریخت‌شناسی RLSA، که ترکیبی از اپراتور بستن و به دنبال آن اپراتور فرسایش است، اعمال می‌گردد تا تصاویر تجزیه‌شده‌ای که شامل پیکسل‌های متنی هستند، تعیین شوند. برای حذف اشیاء غیرمتنی از یک رهیافت بر پایه اندازه نویسه در مثال متنی، استفاده می‌شود. اندازه متوسط نویسه‌ها در یک مثال متنی محاسبه می‌شود و سپس اجزای متصل استخراج شده از لایه متنی که کوچکتر از نصف یا بزرگتر از دو برابر اندازه متوسط هستند، فیلتر می‌شوند. در مرحله تشخیص برجسب از یک تکنیک خودکار استفاده می‌شود که شامل سه جزء است: (۱) الگوریتم گسترش شرطی^۱ (CDA) (۲) الگوریتم

^۱ Conditional Dilatation Algorithm

ردیابی جهت رشته تکی^۱ (SSOD) (۳) نویسه‌خوان نوری (OCR). در ابتدا CDA یک لایه متنی را به موزاییک‌های همپوشان تقسیم می‌کند، رشته‌های متنی مجزا را در هر موزاییک قرار می‌دهد و رشته‌های متنی شناسایی شده را یکی می‌کند. هنگامی که رشته‌های مجزا در مکان خود قرار داده می‌شوند، SSOD جهت رشته‌ها را ردیابی می‌کند. در نهایت رشته‌ها برای ورود به محصول OCR، جهت تشخیص نویسه‌های آن‌ها به صورت افقی چرخانده می‌شوند. دو اسکن باید در این اقدام صورت بگیرد. در اولین اسکن شرط اتصال نویسه^۲، شرط اندازه نویسه^۳، شرط گسترش نویسه^۴ و سپس شرط خمیدگی رشته^۵ مورد بررسی قرار می‌گیرد. در اسکن دوم هر کاندید گسترش با استفاده از شروط یکسان با اسکن اول بررسی می‌شود. ما به اسکن دوم نیازمندیم تا نماینده‌ی گسترش را مجدداً بررسی کنیم. بعد از این مرحله خروجی CDA مورد بررسی قرار می‌گیرد. بعد از این که CDA در صورت عدم وجود کاندید دیگری برای گسترش متوقف شد، هر عنصر متصل از نتایج گسترش به عنوان یک رشته متنی شناسایی می‌شود. CDA یک نقشه شهری را به موزاییک‌های همپوشان تقسیم می‌کند و هر موزاییک را برای شناسایی برچسب متنی تکی پردازش می‌کند. در مرحله بعدی جهت رشته‌های تکی ردیابی می‌شود و سپس با استفاده از OCR نویسه‌ها تشخیص داده می‌شوند. از اشکالات وارد بر این روش می‌توان به موارد زیر اشاره کرد: (۱) کیفیت پایین تصویر نقشه‌های تست، باعث کاهش کیفیت لایه‌های متنی از قبیل نویسه‌های شکسته یا وجود اشیای غیرمتنی در لایه متنی می‌شود. (۲) شباهت بین نمادها منجر به اشتباه می‌شود. نرم‌افزار Strabo که برای پیدا کردن جهت نویسه‌ها در مرحله استخراج لایه متنی مورد استفاده قرار می‌گرفت و توسط نویسندگان مقاله طراحی شده بود، نمی‌تواند جهات درست را برای رشته‌هایی که به طور قابل توجهی خمیده هستند، ردیابی کند؛ در

¹ Single String Orientation Detection algorithm

² Character connectivity condition

³ Character size condition

⁴ Character expandability condition

⁵ String curvature condition

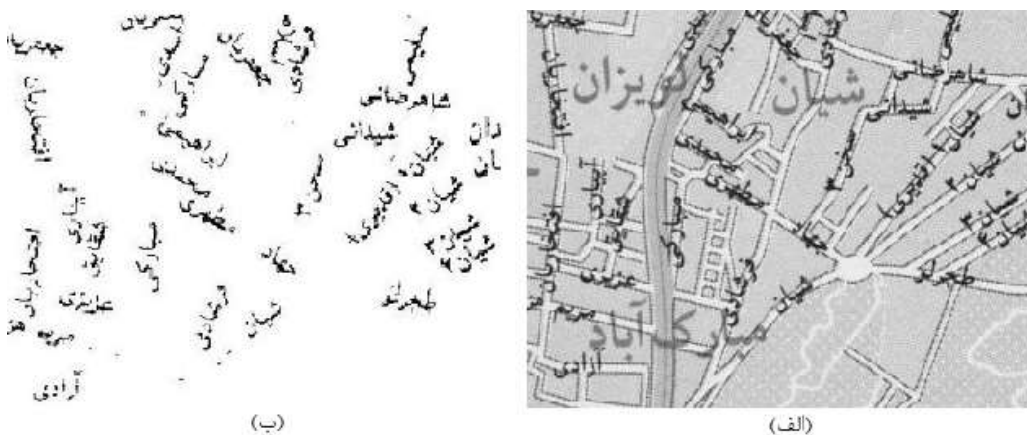
نتیجه نرم‌افزار OCR نمی‌تواند همه نویسه‌ها را در رشته‌های خمیده تشخیص بدهد. ۳) CDA ممکن است نویسه‌های با فاصله زیاد را در چند رشته متنی، گروه‌بندی کند. ۴) نرم‌افزار OCR نمی‌تواند بعضی از نویسه‌های پیکسل‌پیکسل شده را یکنواخت را تشخیص بدهد. ۵) CDA ممکن است نویسه‌ها را با اشیای غیرمتنی گروه‌بندی کند. ۶) از آن جایی که FineReader یک نسخه دستکاپی OCR است که در این مقاله مورد استفاده قرار گرفته است، نتایج حاصل از تشخیص در آن تنها شامل اطلاعات محدودی هستند.

در زمینه استخراج متون فارسی نیز پیشتر آقایان کبیر، غفاری و کنگرلو تحقیقاتی انجام داده بودند. آن‌ها در [۲۴] روشی جدید برای جداسازی متن فارسی از تصویر اسکن شده نقشه‌های شهری ارائه کرده‌اند. ایشان برای حل مشکل ناشی از همپوشانی و تلاقی متون با نمادها و گرافیک‌های موجود در نقشه ابتدا تصویر خاکستری را با استفاده از روش اتسو^۱ به یک تصویر دودویی تبدیل می‌کنند تا عناصر متصل به وجود آیند. سپس عناصر متصل با اندازه‌ای کمتر از یک آستانه مشخص از این تصویر دودویی حذف می‌شوند. عناصر متصل براساس ویژگی‌های اندازه مساحت (تعداد پیکسل‌های عناصر متصل) و نسبت طول به عرض کوچکترین مستطیل محیطی عناصر متصل، مورد تحلیل قرار می‌گیرند. در اینجا فرض شده است که نوشته‌ها دارای رنگ تیره هستند و تیره‌ترین عنصر به عنوان آستانه برای استخراج لایه متنی در نظر گرفته می‌شود؛ سایر عناصر نیز جزء گرافیک‌ها محسوب می‌شوند. با استفاده از فیلتر اندازه مساحت و با این فرض که عناصر متصل با اندازه مساحت بالا جزء گرافیک‌ها هستند، عناصر متصل کوچک به عنوان متن دسته‌بندی می‌شوند. فیلتر نسبت دید^۲، مستطیل محیطی کمینه را برای هر مؤلفه همبند بدست می‌آورد. نسبت طول به عرض این مستطیل، نسبت دید مؤلفه همبند خواهد بود. با این ویژگی می‌توان عناصر متصل با آستانه ۱۵ و بالاتر را گرافیک در نظر گرفت. زیرا، متون دارای نسبت دید کوچکتر از این آستانه می‌باشند. در بخش بعدی

¹ Otsu

² Aspect ratio

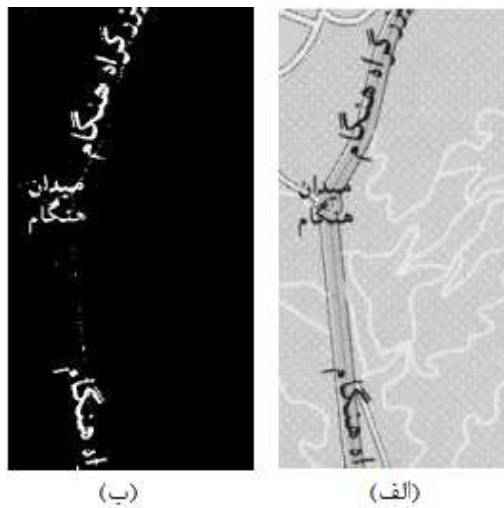
فیلتر اندازه مساحت و نسبت دید با یکدیگر ترکیب می‌شوند. عناصر متصل، زمانی متن تشخیص داده می‌شوند که از هر دو فیلتر به عنوان متن عبور کنند و هنگامی به عنوان گرافیک تشخیص داده می‌شوند که از هر دو فیلتر به عنوان گرافیک عبور نمایند. سپس بین متون و گرافیک‌ها تناظری پیدا می‌شود. میانگین شدت روشنایی پیکسل‌های متن متناظر تصویر دودویی بدست آمده با تصویر خاکستری محاسبه می‌شود. بدین طریق دو آستانه جداساز برای تفکیک متن و گرافیک ایجاد می‌شود. این روش کلمات را با دقت ۶۰٪ و زیرکلمات را با دقت ۶۵٪ به درستی آشکارسازی می‌کند [۲۴]. شکل (۷-۲) نتیجه استخراج لایه متنی از یک نقشه سطح خاکستری را توسط روش [۲۴] نشان می‌دهد.



شکل ۷-۲: نتیجه استخراج متن از یک نقشه توسط روش دکتر کبیر و همکارانش. الف) تصویر سطح خاکستری ب) لایه متنی استخراج شده [۲۴]

در کاری دیگر [۲۵] دکتر کبیر و همکارانش روشی را برای بازنمایی بر پایه تبدیل فاصله برای آشکارسازی متن در تصاویر نقشه‌های شهری ارائه کرده‌اند. آن‌ها ابتدا تصویر سطح خاکستری را به یک تصویر دودویی تبدیل می‌کنند. در تصویر دودویی بدست آمده تعداد پیکسل‌های متن و خطوط گرافیکی کمتر از تعداد پیکسل‌های پس‌زمینه است. با کمک این ویژگی می‌توان پس‌زمینه تصویر را از پیش‌زمینه آن جدا کرد. برای محاسبه بازنمایی تصویر پیکسل‌های پس‌زمینه با صفر و پیکسل‌های پیش‌زمینه با یک برچسب زده می‌شوند. سپس عناصر متصل هشت همسایگی برای تصویر دودویی

بدست می‌آیند. برای این کار از یک تبدیل فاصله با معیار اقلیدوسی استفاده می‌شود. فاصله اقلیدوسی هر پیکسل در مؤلفه همبند با هشت پیکسل مرزی آن محاسبه می‌شود. این هشت پیکسل مرزی محل برخورد چهار اشعه در چهار راستای عمودی، افقی و مورب ۴۵ و ۱۳۵ درجه هستند. از کوچکترین این فاصله‌ها برای بازنمایی تصویر استفاده شده است. با افزایش مقادیر هیستوگرام، مجموعه‌ای از تصاویر پایه را بدست می‌آوریم. در بهترین حالت خطای جداسازی متن ۲۲/۷٪ و خطای جداسازی گرافیک ۵۶٪ شده است [۲۵]. شکل (۲-۸) نتیجه استخراج متن با استفاده از یک تبدیل فاصله که توسط دکتر کبیر و همکارانش ارائه شده بود را نشان می‌دهد.



شکل ۲-۸: استخراج متن از نقشه با استفاده از تبدیل فاصله (الف) یک تصویر سطح خاکستری (ب) متن بازیابی شده [۲۵]

۲-۶ جمع‌بندی و نتیجه‌گیری

گرچه پیشتر کارهایی در زمینه جداسازی متن و گرافیک انجام شده است، اما هیچ یک از آنها روشی جامع و خودکار نبودند. برچسب‌های متنی در نقشه‌ها از راستای خطوط گرافیکی خیابان‌ها و کوچه‌ها تبعیت می‌کنند. بنابراین چون خیابان‌ها و کوچه‌ها جهات مختلفی را دنبال می‌کنند، برچسب‌های متنی هم دارای جهات‌های مختلفی هستند. از طرفی به علت فشردگی اطلاعات گرافیکی در این نوع نقشه‌ها، گاهی متون و خطوط گرافیکی با یکدیگر همپوشانی دارند که این امر جداسازی

آن‌ها را دشوار می‌سازد. از طرف دیگر همه متون در یک نقشه دارای یک رنگ مشخص یا یک قلم با اندازه مشخص نیستند. رنگ متون بسته به مکانی که به آن اشاره دارند، ممکن است از برجسب‌های متنی خیابان متفاوت باشد. اندازه عرض قلم هم بسته به عرض کوچه یا خیابان می‌تواند در یک نقشه مقادیر متفاوتی داشته باشد. در مورد قلم مورد استفاده در نقشه‌ها هم باید خاطر نشان کرد که همه نقشه‌ها دارای یک استاندارد مشخص برای نوع قلم نیستند و قلم مورد استفاده ممکن است از نقشه‌ای به نقشه دیگر متفاوت باشد.

اکثر کارهای قبلی انجام شده در زمینه استخراج متن، بر روی متون لاتین صورت پذیرفته است. فرض تمامی این روش‌ها، جدا بودن حروف چاپی زبان انگلیسی از یکدیگر است. فرض دیگر در انجام کارهای قبلی، جدا بودن تعداد قابل توجهی از متون تصویر از گرافیک‌ها است. بیشتر کارهای انجام شده محدود به یک قلم، اندازه، رنگ یا جهت مشخص بودند و نمی‌توانستند از عهده نقشه‌ها یا حتی متون چندجهته و دارای قلم، رنگ و اندازه مختلف برآیند. علاوه بر این دخالت کاربر در بیشتر این کارها زیاد بود.

در بحث نقشه‌های تهیه شده از خیابان‌های سطح شهر، رنگ از اهمیت بسیاری برخوردار است. در این قبیل نقشه‌های جغرافیایی، متون، خطوط خیابان‌ها، ساختمان‌های مهم مانند بیمارستان‌ها، مدارس، مساجد و ... با رنگ‌های متفاوتی نمایش داده می‌شوند. لذا بخش‌بندی و کوانتیزه کردن رنگ‌ها می‌تواند به جداسازی لایه‌های مختلف نقشه و نیز به استخراج متون از آن‌ها کمک کند.

جامع‌ترین روشی که تاکنون در زمینه استخراج متن ارائه گردیده است، روش مربوط به کار [۵] چیانگ و نوبلاک است که در زمینه متون لاتین انجام گرفته است. به نظر می‌رسد که این روش به علت پیچیدگی متون فارسی نمی‌تواند به خوبی از عهده استخراج این متون برآید. همان‌طور که در توضیح روش مذکور نیز بیان شد، این روش به صورت نیمه خودکار است و نیاز به دخالت کاربر دارد.

در زبان فارسی بعضی از حروف کلمات به یکدیگر متصل می‌شوند و حتی شکل حروف نیز موقع اتصال به یکدیگر تغییر می‌نماید. در نقشه‌های شهری، متون و گرافیک‌ها بسیار به هم نزدیک هستند. به خصوص در نقشه‌های شهر تهران، اسامی خیابان‌ها و کوچه‌ها بسیار به هم نزدیک است و نمی‌توان فرض نمود که خطوط و منحنی‌های گرافیکی تصویر از متن جدا هستند [۲۴].

در مورد زبان فارسی هم همان‌گونه که پیشتر بیان شد، تنها در [۲۴] و [۲۵] دکتر کبیر و همکارانش روش‌هایی را ارائه کرده‌اند. اما هر دوی این روش‌ها روی تصاویر سطح خاکستری کار می‌کند و از خاصیت رنگ بهره نمی‌برد؛ علاوه بر این، روش‌های پیشنهادی آن‌ها با این فرض کار می‌کنند که متون دارای رنگ‌های تیره هستند و از تیره‌ترین رنگ به عنوان آستانه برای دودویی کردن تصویر استفاده می‌کنند؛ در حالی که در بعضی نقشه‌ها ممکن است متون دارای رنگ‌های دیگر مانند آبی، قرمز و ... نیز باشند. به همین دلیل نتیجه حاصل از کار آن‌ها روی نقشه‌های دارای متون رنگی مطلوب نیست؛ علاوه بر این درصد دقت^۱ و فراخوانی^۲ بیان شده در این روش‌ها هم نشان از ضعف آن‌ها در استخراج متون دارد.

در این پایان‌نامه سعی بر آن داریم تا روش خودکار جداسازی متون فارسی و لاتین از نقشه‌های جغرافیایی را بهبود دهیم. در این رهیافت از روشی بر پایه الگوریتم جابه‌جایی میانگین و نگاشت خود سامان‌ده برای کوانتیزاسیون رنگ استفاده می‌شود که در هیچ یک از کارهای پیشین چنین الگوریتمی برای کوانتیزاسیون مورد استفاده قرار نگرفته است. وجود نقطه‌ها، سرکش‌ها، مدها، بالارونده‌ها و پایین‌رونده‌ها، ساختار زبان فارسی را از زبان لاتین متمایز می‌کند. لذا روش‌های موجود قابل اعمال به نقشه‌های لاتین، قابل استفاده به صورت مستقیم روی نقشه‌های فارسی نیستند. اندک روش‌های فارسی موجود نیز روی تصاویر سطح خاکستری کار می‌کنند و با توجه به این که بیشتر نقشه‌های

¹ Precision

² Recall

امروزی رنگی هستند، از قابلیت رنگ در نقشه‌ها استفاده نمی‌شود. لذا به دنبال ارائه روشی خودکار هستیم که مشکلات روش‌های موجود را تا حدودی برطرف کند و درصد دقت و فراخوانی بالاتری داشته باشد.

فصل سوم

تئوری الگوریتم‌های مورد استفاده

۳-۱ مقدمه

در این فصل از پایان نامه سعی بر آن داریم تا مسأله پیش‌رو و تئوری مربوط به الگوریتم‌های مورد استفاده را شرح دهیم. روش‌هایی که در طرح پیشنهادی ما مورد استفاده قرار گرفته‌اند، عبارتند از: جابه‌جایی میانگین و روش نگاشت خود سامان‌ده که خود کوانتیزاسیون فضای رنگ را مطرح می‌کنند. بنابراین ما در ادامه به شرح هر یک از سه مفهوم جابه‌جایی میانگین، نگاشت خودسامان‌ده و کوانتیزاسیون می‌پردازیم، تاریخچه مختصری از پیدایش هر یک از این مفاهیم و کاربردهای آن‌ها را بیان می‌کنیم.

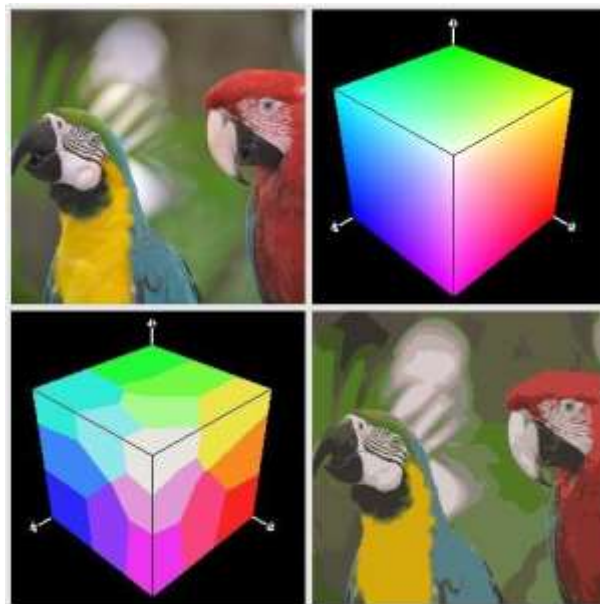
۳-۲ کوانتیزاسیون

یک تصویر دیجیتال معمولاً شامل ده‌ها هزار رنگ است. بنابراین، بیشتر دستگاه‌های پردازش تصویر در ابتدا نیازمند یک طرح کاهش رنگ پیش از اجرای عملیات پیچیده بخش‌بندی هستند [۲۶]. با اعمال یک روش کوانتیزاسیون استاندارد به یک مدل رنگ، هر محور در فضای تصویر به قسمت‌هایی تقسیم‌بندی می‌شود [۲۷].

۳-۲-۱ کوانتیزاسیون رنگ

کوانتیزاسیون در پردازش تصویر یک تکنیک فشرده‌سازی دارای اتلاف است. نمایشگرهای ویدئویی، تصاویر رنگی را با تغییر مقدار سه رنگ اولیه قرمز، سبز و آبی در فضای RGB برای هر پیکسل نشان می‌دهند. در یک تصویر کوانتیزه، هر یک از رنگ‌های اولیه معمولاً با وضوح هشت بیت نشان داده می‌شود. نمایش رنگی کامل رقمی از ۲۴ بیت استفاده می‌کند تا رنگ هر پیکسل را در صفحه تصویر مشخص کند. هزینه‌ای که حافظه با سرعت بالا برای پشتیبانی نمایش کامل رنگ‌های یک تصویر روی نمایشگر با وضوح بالا تحمیل می‌کند، بسیاری از کاربردها را غیرعملی می‌سازد. یک رهیافت که توسط نمایشگرهای موجود مورد استفاده قرار می‌گیرد، تعداد محدودی از بیت‌ها (مثلاً هشت بیت) را برای تشخیص رنگ هر پیکسل مهیا می‌کند. هر یک از این $2^8 = 256$ مقدار به عنوان یک شاخص در یک

جدول رنگ از پیش تعریف شده توسط کاربر، استفاده می‌شوند. هر ورود به جدول شامل یک مقدار 2^4 بیتی است که عناصر قرمز، سبز و آبی را مشخص می‌کنند. در این روش به کاربر اجازه داده می‌شود تا یک زیرمجموعه کوچک از رنگ‌ها به نام یک "الگو" را از محدوده کامل $2^4 = 16777216$ رنگ انتخاب کند [۲۸]. شکل (۱-۳) نمونه‌ای از عمل کوانتیزاسیون رنگ روی یک تصویر واقعی و تأثیر آن در فضای رنگ سه‌بعدی را نشان می‌دهد.



شکل ۱-۳: نمونه‌ای از عمل کوانتیزاسیون رنگ روی یک تصویر واقعی و در فضای رنگ سه‌بعدی [۲۹]

کوانتیزاسیون رنگ فرآیند کاهش تعداد رنگ‌های متمایز استفاده شده در یک تصویر و نمایش رنگی آن با استفاده از تعداد محدودی از رنگ‌ها است. تصویر حاصل باید تا حد امکان از لحاظ دیداری شبیه به تصویر اصلی باشد. هدف اصلی کوانتیزاسیون رنگ کاهش فضای ذخیره‌سازی در حافظه، تسریع زمان ارسال تصاویر و انجام پردازش روی آن‌ها است. با استفاده از یک الگوریتم کوانتیزاسیون مناسب و فشرده‌سازی با تلافی، تصویر می‌تواند با کیفیت یکسان از نظر دیداری و با حجم کمتری نسبت به حالت اولیه ذخیره شود [۳۰]. برای مثال یک تصویر رقمی با 512×512 پیکسل در قالب غیرفشرده 24 bit/pixel دارای حجم 768 کیلوبایت است. حال اگر بتوان تصویر را به 256 رنگ

تبدیل کرد، حجم آن به یک سوم حجم اولیه کاهش می‌یابد.

الگوریتم‌های کامپیوتری برای اجرای کوانتیزاسیون رنگ روی نقشه بیتی از دهه ۱۹۷۰ مورد بررسی قرار گرفته است. تصاویر رنگی معمولاً شامل هزاران رنگ هستند و نمایش، ذخیره‌سازی و پردازش آن‌ها امری چالش برانگیز می‌باشد. به همین دلیل کوانتیزاسیون رنگ (کاهش رنگ) معمولاً به عنوان یک گام پیش‌پردازش برای بسیاری از کاربردهای پردازش تصویر و گرافیک مورد استفاده قرار می‌گیرد. در گذشته کوانتیزاسیون رنگ به دلیل محدودیت‌های نمایش در سخت‌افزارهایی که نمی‌توانستند ۱۶ میلیون رنگ ممکن را در تصاویر ۲۴ بیتی نمایش دهند، ضروری بود. کاربردهای کوانتیزاسیون رنگ (۱) فشرده‌سازی^۱ تصویر (۲) بخش‌بندی^۲ تصویر (۳) تحلیل^۳ تصویر (۴) بازیابی تصویر بر پایه محتوا^۴ است [۳۱].

فرآیند کوانتیزاسیون رنگ اساساً از دو فاز تشکیل شده است. طراحی الگو (انتخاب یک مجموعه کوچک از رنگ‌های غالب که رنگ‌های تصویر اصلی را نمایش می‌دهند) و نگاشت پیکسل (تخصیص هر پیکسل ورودی به یکی از الگوهای رنگ). در بیشتر کاربردها پیکسل‌های ۲۴ بیتی در تصویر اصلی به هشت بیت یا کمتر کاهش می‌یابد. از آنجایی که تصاویر طبیعی معمولاً شامل تعداد زیادی از رنگ‌ها هستند، نمایش این تصاویر با یک الگوی با اندازه محدود مشکل است. کوانتیزاسیون رنگ به صورت عمده می‌تواند در دو گروه طبقه‌بندی شود: روش‌های مستقل تصویر که یک الگوی (ثابت) کلی را بدون توجه به تصویری خاص تعیین می‌کنند و روش‌های وابسته به تصویر که یک الگوی عادی (وفقی) را بر پایه توزیع رنگ تصاویر تعیین می‌کنند. علی‌رغم سریع بودن، روش‌های مستقل از تصویر معمولاً نتایج ضعیفی به دنبال دارند. چون محتوای تصویر را به حساب نمی‌آورند. بنابراین بیشتر مطالعات در زمینه روش‌های وابسته به تصویر است که می‌کوشند تا به تعادل بهتری بین کارآمدی

¹ compression

² Segmentation

³ Analysis

⁴ Content-based image retrieval

محاسباتی و کیفیت دیداری خروجی کوانتیزاسیون رنگ برسند [۳۱].

روش‌های کوانتیزاسیون رنگ وابسته به تصویر، در دهه‌های گذشته بسیار توسعه یافته‌اند و می‌توانند به دو خانواده طبقه‌بندی شوند: روش‌های پیش‌طبقه‌بندی^۱ و روش‌های پس‌طبقه‌بندی^۲. روش‌های پیش‌طبقه‌بندی بیشتر بر پایه تحلیل آماری توزیع رنگ‌ها در تصاویر بنا شده‌اند. روش‌های پیش‌طبقه‌بندی تقسیم‌کننده با یک خوشه تکی که شامل همه پیکسل‌ها است، شروع می‌کنند. خوشه اولیه به صورت بازگشتی^۳ تقسیم می‌شود تا K خوشه حاصل شود. روش‌های تقسیم‌کننده معروف شامل برش میانه، Octree، روش بر پایه واریانس، برش مرکز^۴ و ... هستند [۳۱]. از سوی دیگر روش‌های متراکم پس‌طبقه‌بندی با N خوشه یگانه (N تعداد کل پیکسل‌ها است) که هر یک شامل یک پیکسل هستند، شروع می‌شوند. این خوشه‌ها به طور مکرر ادغام می‌شوند تا K خوشه باقی بماند. در مقایسه با روش‌های پیش‌طبقه‌بندی که الگو را تنها یک بار محاسبه می‌کنند، روش‌های پس‌طبقه‌بندی ابتدا یک الگوی اولیه را تعیین می‌کنند و سپس به طور مکرر بهبود می‌یابند. چون این روش‌ها با بهینه‌سازی تکراری یا تصادفی درگیر هستند، می‌توانند نتایج خوشه‌بندی بهتری را در مقایسه با روش‌های پیش‌طبقه‌بندی، البته به بهای افزایش زمان محاسبه، حاصل کنند [۳۱]. روش‌های خوشه‌بندی برای کوانتیزاسیون تصویر شامل K میانگین، کمترین حداکثر^۵، یادگیری رقابتی^۶، C میانگین فازی^۷ و نگاشت خود سامان‌ده هستند که ما از روش نگاشت خود سامان‌ده در رهیافت پیشنهادی خود استفاده کرده‌ایم. جدای از دو مرحله اصلی، مخصوصاً در مورد همگرایی، تکنیک کوانتیزاسیون رنگ باید برای محدود کردن تعداد رنگ‌ها طی پیش‌پردازش، تعداد مناسبی از رنگ‌های

¹ Preclustering

² Postclustering

³ Recursively

⁴ Center-cut

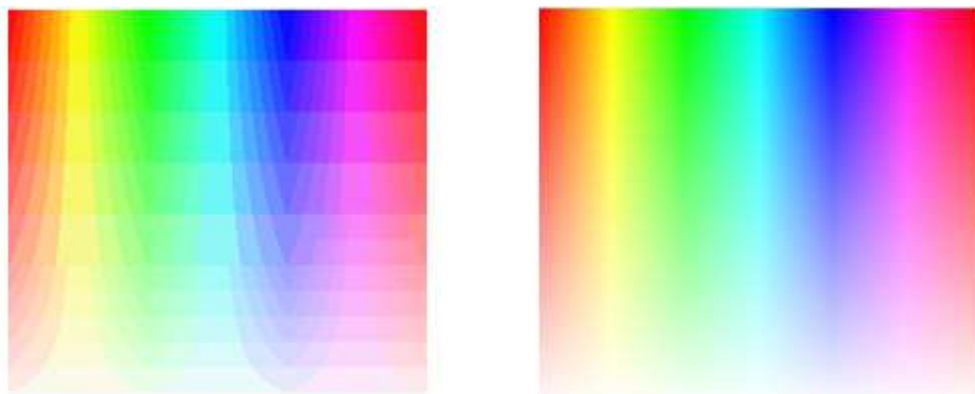
⁵ MinMax

⁶ Competitive learning

⁷ Fuzzy C-Means

غالب تصویر را تعیین کند. در این مورد تکنیک کوانتیزاسیون رنگ به طور بدون نظارت به تعدادی رنگ، همگرا می‌شود. اگر ما تنها رنگ‌های تصویر را در نظر بگیریم، در یک فضای رنگ سه بعدی که هر بردار یک عنصر رنگ سه‌گانه^۱ تصویر را نشان می‌دهد، مسأله کوانتیزاسیون رنگ یک مسأله خوشه‌بندی برداری بدون نظارت خواهد بود [۳۲].

همه روش‌های فوق، الگوهای رنگی را با نمایش خوشه‌بندی کلی در فضای رنگ ایجاد می‌کنند. به عبارت دیگر، آن‌ها یک کوانتیزاسیون رنگ را اجرا می‌کنند که همه رنگ‌های تصویر را به شمار می‌آورد. بنابراین الگوهای رنگ یک کتاب رمز^۲ را نشان می‌دهند که در آن همه رنگ‌های تصویر به صورت بهینه نگاشت داده می‌شوند [۳۲]. تصویر (۲-۳) نمونه‌ای از عمل کوانتیزاسیون رنگ در صفحه تصویر و الگوی تصویر نتیجه را نشان می‌دهد.



(ب) تصویر نتیجه (رنگ‌ها به ۸ بیت کاهش یافته‌اند)

(الف) تصویر اصلی (۲۴ بیت)

شکل ۲-۳: عمل کوانتیزاسیون در صفحه تصویر [۳۳]

۳-۲-۲ کاربرد

در روزهای نخست پیدایش کامپیوترهای شخصی برای مبدل‌های ویدئو رایج بود که تنها ۲، ۴، ۱۶ یا نهایتاً ۲۵۶ رنگ به خاطر محدودیت‌های حافظه ویدئویی پشتیبانی شوند. ترجیح بر آن بود که

¹ Triple

² Codebook

وضوح تصویر افزایش یابد؛ به عبارت دیگر پیکسل‌های بیشتر (وضوح بالاتر) نسبت به تنوع رنگ بیشتر وجود داشته باشند. کوانتیزاسیون رنگ کمک کرد تا این مصالحه توسط نمایش تصاویر با کیفیت بالا در حالت‌های ۱۶ و ۲۵۶ رنگ با کاهش کمی در کیفیت تصاویر، برقرار شود. بسیاری از نمایشگرهای تصویر، به صورت خودکار کوانتیزاسیون رنگ را اجرا می‌کنند و تصاویر رنگی با تعداد رنگ زیاد را در یک حالت محدود شده ویدئویی با ۲۵۶ رنگ نمایش می‌دهند. کامپیوترهای مدرن اکنون می‌توانند میلیون‌ها رنگ را در یک لحظه نمایش دهند، یعنی بیش از محدوده‌ای که حتی چشم انسان قادر به دیدن آن باشد.

صفحات وب تنها تا ۲۵۶ رنگ را پشتیبانی می‌کند. بعضی از جستجوگران وب تصاویر را به الگوی خاصی محدود می‌کنند که به عنوان رنگ‌های وب شناخته می‌شوند و منجر به کاهش کیفیت در مقایسه با الگوهای بهینه می‌شوند. تصاویر PNG با رنگ ۲۴ بیت می‌توانند با کاهش اندازه فایل، بدون کاهش زیادی که چشم بتواند آن را حس کند، در کاربردهای کوانتیزاسیون رنگ استفاده شوند؛ چون فایل‌های PNG از بیت بر پیکسل‌های کمتری برای تصاویر استفاده می‌کنند.

علاوه بر این‌ها، همان‌گونه که پیشتر نیز ذکر کردیم، کوانتیزاسیون رنگ موجب کاهش حافظه مورد نیاز برای ذخیره‌سازی، تسریع در ارسال تصاویر و بالا بردن سرعت پردازش تصاویر در پردازشگرهای مختلف می‌شود.

۳-۳ الگوریتم جابه‌جایی میانگین^۱

جابه‌جایی میانگین یک تکنیک تحلیل فضای ویژگی غیرپارامتریک و تکرارشونده است که هر داده را به سمت ماکزیمم محلی تابع چگالی جابه‌جا می‌کند [۳۴]. روش جابه‌جایی میانگین را نخستین بار فوکوناگا و هاستلر در سال ۱۹۷۵ مطرح کردند [۳۵]. چنگ فرمول‌بندی جابه‌جایی میانگین را

^۱ Mean-shift algorithm

عمومیت بخشید و کاربرد آن را در خوشه‌بندی و بهینه‌سازی نشان داد [۳۶]. پس از آن جابه‌جایی میانگین به طور وسیع در ردیابی اشیاء، بخش‌بندی تصویر، تشخیص الگو، خوشه‌بندی، فیلتر کردن و ترکیب اطلاعات^۱ به کار رفت [۳۴].

الگوریتم جابه‌جایی میانگین یک روش خوشه‌بندی بر پایه خوشه‌بندی بدون نظارت است که در آن تعداد و شکل خوشه‌های داده در ابتدا نامعلوم است. فرآیند بخش‌بندی بر پایه راهبرد یکی کردن بعضی نواحی است که به نتیجه تصویر فیلتر شده اعمال می‌شود. تعداد نواحی در تصویر بخش‌بندی شده توسط کمترین تعداد پیکسل‌ها در یک ناحیه تعیین می‌شود. نواحی دارای تعداد پیکسل کمتر از این مقدار مشخص، حذف خواهند شد و با نواحی همسایه یکی می‌شوند [۳۷].

جابه‌جایی میانگین یک روش تکرار شونده است و ما آن را با یک تخمین از x شروع می‌کنیم. یک تابع کرنل $K(x_i - x)$ داده شده است. این تابع، وزن نقاط مجاور برای دوباره تخمین زدن میانگین را تعیین می‌کند. معمولاً از یک کرنل گاوسی روی یک فاصله برای تخمین جاری استفاده می‌شود:

$$K(x_i - x) = e^{-c\|x_i - x\|^2}$$

میانگین وزن دار چگالی در پنجره توسط K تخمین زده شده است [۳۶]:

$$m(x) = \frac{\sum_{x_i \in N(x)} K(x_i - x)x_i}{\sum_{x_i \in N(x)} K(x_i - x)} \quad (1-3)$$

که در آن $N(x)$ همسایگی x است و مجموعه‌ای از نقاط است که در آن $K(x) \neq 0$.

تفاوت $m(x) - x$ در الگوریتم فوکوناگا و هاستنلر، جابه‌جایی میانگین نامیده می‌شود [۳۵]. الگوریتم جابه‌جایی میانگین اکنون $m(x) \leftarrow x$ تنظیم می‌کند و تخمین را تا همگرایی $m(x)$ تکرار می‌کند.

گرچه الگوریتم جابه‌جایی میانگین در بسیاری موارد کاربرد دارد، هنوز هم اثباتی محکم برای همگرایی این الگوریتم با استفاده از یک کرنل کلی در فضای ویژگی با ابعاد بالا وجود ندارد [۳۸].

¹ Information Combining Strategy

علیاری قصابه^۱ همگرایی الگوریتم جابه‌جایی میانگین را در یک بعد با یک تابع پروفایل مشتق‌پذیر، محدب و به شدت کاهشی نشان می‌دهد [۳۹].

داده‌ها را در یک مجموعه محدود S تعبیه شده در فضای اقلیدوسی n بعدی X در نظر بگیرید. K را یک کرنل مسطح در نظر بگیرید که در تابع مشخصه توپی λ^{-2} در X است [۳۶]:

$$K(x) = \begin{cases} 1 & \text{if } \|x\| \leq \lambda \\ 0 & \text{if } \|x\| > \lambda \end{cases} \quad (2-3)$$

در هر تکرار از الگوریتم، $m(s) \leftarrow s$ بر روی همه $s \in S$ در یک زمان اجرا می‌شود. اولین سؤال این است که چگونه تابع چگالی داده شده یک مجموعه پراکنده از نمونه‌ها را تخمین می‌زند؟ یکی از ساده‌ترین رهیافت‌ها، ساده‌سازی داده‌ها است. برای مثال با کانالو آن با یک کرنل ثابت به عرض h داریم [۳۹]:

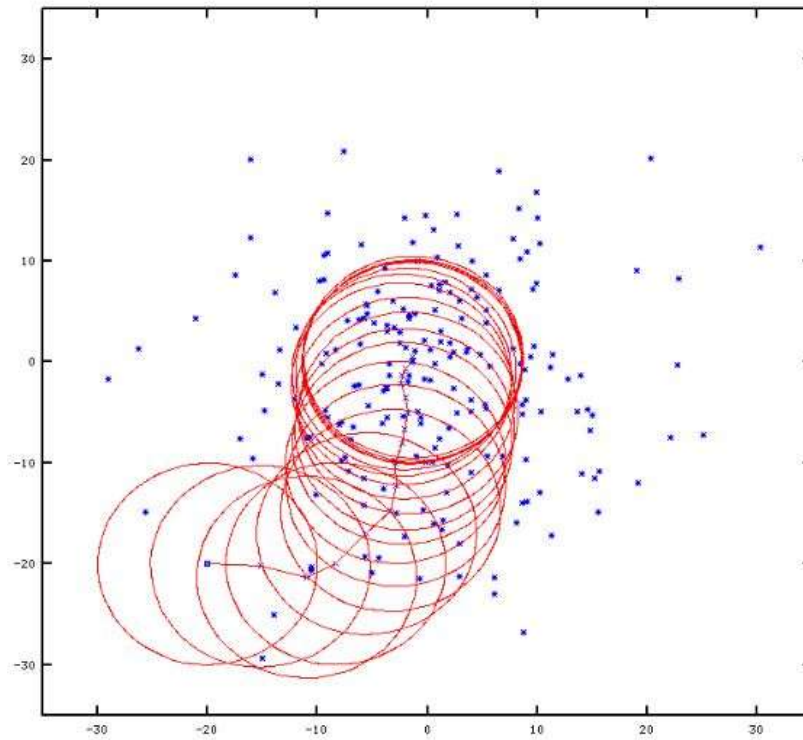
$$f(x) = \sum_i K(x - x_i) = \sum_i K\left(\frac{\|x - x_i\|^2}{h^2}\right) \quad (3-3)$$

که x_i ها نمونه‌های ورودی هستند و $K(r)$ تابع کرنل یا پنجره پارزن می‌باشد. h تنها پارامتر در الگوریتم است و پهنای باند نامیده می‌شود. این رهیافت تخمین چگالی کرنل یا تکنیک پنجره پارزن نامیده می‌شود. هنگامی که $f(x)$ را از معادله بالا محاسبه می‌کنیم، می‌توانیم ماکزیمم محلی آن را با استفاده از گرادیان یا برخی از تکنیک‌های دیگر بهینه‌سازی پیدا کنیم. مشکل این رهیافت این است که برای ابعاد بالاتر از نظر محاسباتی، ارزیابی $f(x)$ روی فضای جستجوی کامل انجام نمی‌گیرد. در

¹ Aliyari Ghassabeh

² Ball

عوض، جابه‌جایی میانگین از یک متغیر که در مقالات بهینه‌سازی به عنوان گرادیان کاهش با چندین شروع مجدد شناخته شده است، پیروی می‌کند و با شروع از یک حدس ماکزیمم محلی y_k را محاسبه می‌کند. شکل (۳-۳) چگونگی یافتن نقطه همگرایی را در الگوریتم جابه‌جایی میانگین نشان می‌دهد.



شکل ۳-۳: چگونگی یافتن نقطه همگرایی در الگوریتم جابه‌جایی میانگین [۴۰]

۳-۳-۱ انواع کرنل‌ها

تعریف کرنل: X را یک فضای اقلیدوسی n بعدی R^n در نظر بگیرید و x را با x_i علامت‌گذاری کنید. نرم x یک عدد غیرمنفی است [۳۶].

$\|x\|^2 = x^T x$ یک تابع $K: X \rightarrow R$ است و بیان می‌کند که برای بودن یک کرنل اگر یک

پروفایل وجود داشته باشد $K: [0, \infty) \rightarrow R$ مانند $K(x) = K(\|x\|^2)$ که در آن [۳۶]

• K غیر منفی است.

$K(a) \geq K(b)$ اگر $a < b$ غیرافزایشی است:

• K تکه‌ای پیوسته است و $\int_0^\infty K(r)dr < \infty$

دو پروفایل کرنلی که معمولاً برای جابه‌جایی میانگین استفاده می‌شوند، به صورت زیر هستند [۳۶]:

$$K(x) = \begin{cases} 1 & \text{if } x \leq \lambda \\ 0 & \text{if } x > \lambda \end{cases} \quad \text{کرنل مسطح} \quad (۴-۳)$$

کرنل گاوسی

$$K(x) = e^{-\frac{\|x\|^2}{2\sigma^2}} \quad (۵-۳)$$

که σ پارامتر انحراف معیار است و به عنوان پهنای باند پارامتر h کار می‌کند.

۳-۲ خوشه‌بندی الگوریتم جابه‌جایی میانگین

الگوریتم جابه‌جایی میانگین یک تکنیک خوشه‌بندی غیرپارامتریک است که نیاز به دانش اولیه‌ای از تعداد خوشه‌ها ندارد و شکل دسته‌ها را محدود نمی‌کند.

N نقطه داده $x_i, i=1, \dots, n$ روی یک فضای d بعدی R^d داده شده است. تخمین چگالی کرنل

چند متغیره حاصل شده با کرنل $K(x)$ و شعاع پنجره h به صورت زیر است [۴۱]:

$$f(x) = \frac{C_{k,d}}{nh^d} \sum_{i=1}^n K\left(\left\|\frac{x-x_i}{h}\right\|^2\right) \quad (۶-۳)$$

که $C_{k,d}$ یک ثابت نرمالیزاسیون است که تضمین می‌کند جمع $K(x)$ ها برابر با یک است. نماهای

تابع چگالی در صفرهای تابع گرادیان $\nabla f(x) = 0$ قرار دارد.

$$\nabla f(x) = \frac{2c_{k,d}}{nh^{d+2}} \sum_{i=1}^n (x - x_i) K' \left(\left\| \frac{x - x_i}{h} \right\|^2 \right) \quad (7-3)$$

اگر $g(s) = -K'(s)$ خواهیم داشت [41]:

$$\begin{aligned} \nabla f(x) &= \frac{2c_{k,d}}{nh^{d+2}} \sum_{i=1}^n (x_i - x) g \left(\left\| \frac{x - x_i}{h} \right\|^2 \right) = \\ &= \frac{2c_{k,d}}{nh^{d+2}} \sum_{i=1}^n g \left(\left\| \frac{x - x_i}{h} \right\|^2 \right) \left[\frac{\sum_{i=1}^n x_i g \left(\left\| \frac{x - x_i}{h} \right\|^2 \right)}{\sum_{i=1}^n g \left(\left\| \frac{x - x_i}{h} \right\|^2 \right)} - x \right] \end{aligned} \quad (8-3)$$

اولین عبارت متناسب با تخمین چگالی در x است که با کرنل $G(x) = C_{g,d} g(\|x\|^2)$ محاسبه شده

است و عبارت دوم که در زیر آمده است، همان جابه‌جایی میانگین است [41]:

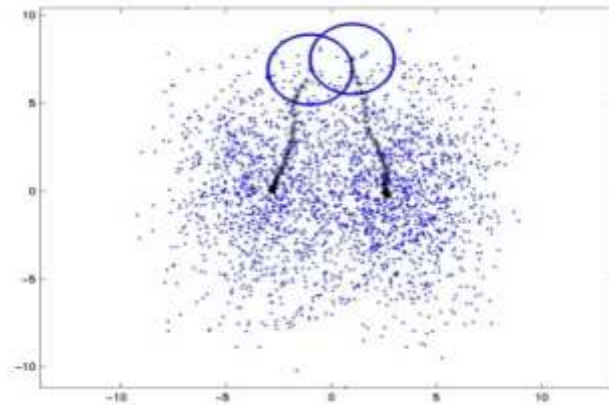
$$m_h(x) = \frac{\sum_{i=1}^n x_i g \left(\left\| \frac{x - x_i}{h} \right\|^2 \right)}{\sum_{i=1}^n g \left(\left\| \frac{x - x_i}{h} \right\|^2 \right)} - x \quad (9-3)$$

جابه‌جایی میانگین همیشه داده‌ها را به سمت جهت ماکزیمم افزایش در چگالی، هدایت می‌کند.

تضمین می‌شود که الگوریتم به یک نقطه که در آن گرادیان تابع چگالی صفر است، همگرا می‌شود.

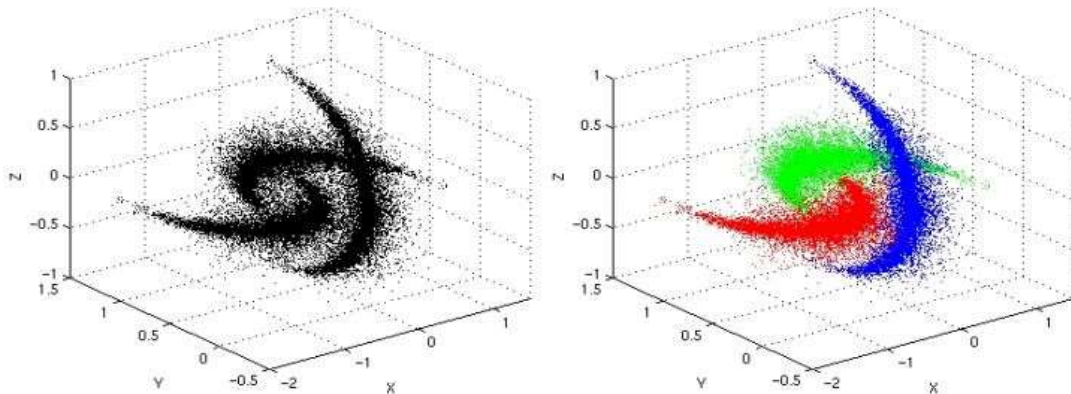
فرآیند پیدا کردن قله چگالی در فرآیند جابه‌جایی میانگین که مربوط به داده‌های دو خوشه می‌باشد،

در شکل (3-4) نمایش داده شده است [41]:

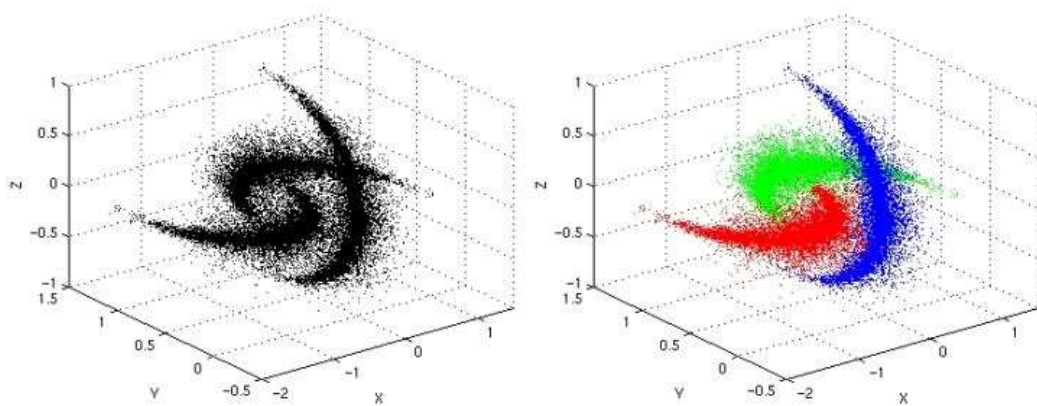


شکل 3-4: پیدا کردن قله جابه‌جایی میانگین [41]

مجموعه تمام محل‌هایی که به قلّه‌های یکسانی همگرا می‌شوند، حوزه جذب آن قلّه را تعریف می‌کنند. نقاطی که در حوزه جذب یکسانی هستند به خوشه یکسانی مرتبط هستند. شکل (۳-۵) دو مثال از خوشه‌بندی جابه‌جایی میانگین را در داده‌های سه بعدی نشان می‌دهد [۴۱]. شکل (۳-۶) نتیجه اعمال الگوریتم جابه‌جایی میانگین روی تصویر یک دلقک را نشان می‌دهد.



شکل (الف): مثال ترکیبی از سه کلاس جدایی‌پذیر غیرخطی (۳۲۶۴۰ نقطه)



شکل (ب): مثال واقعی از ۱۴۸۲۶ نقطه در فضای رنگ LUV

شکل ۳-۵: خوشه‌بندی جابه‌جایی میانگین [۴۱]



شکل ۳-۶: نتیجه اعمال روش جابه‌جایی میانگین روی تصویر یک دلقک [۴۲]

۳-۴ نگاشت خود سامان‌ده^۱ (SOM)

۳-۴-۱ تعریف نگاشت خود سامان‌ده

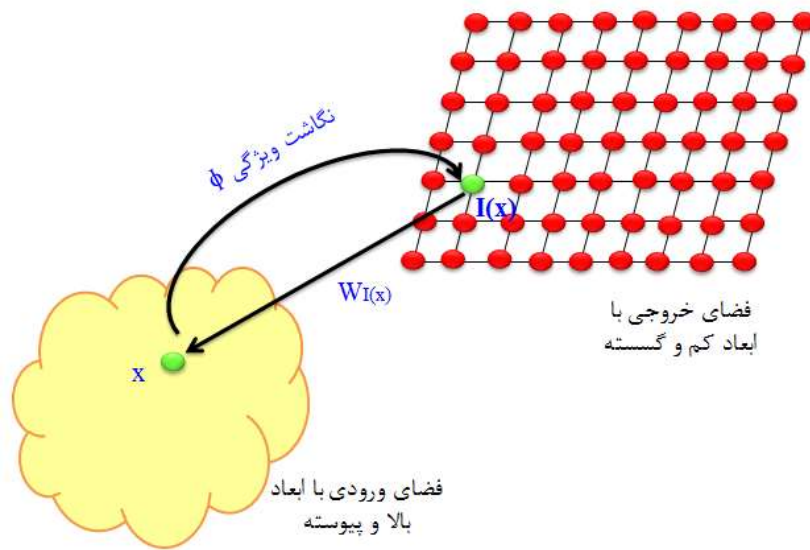
یک نگاشت خود سامان‌ده (SOM) یک شبکه عصبی مصنوعی است که با استفاده از یادگیری بدون نظارت^۲ آموزش می‌بیند تا درکی با ابعاد کم و گسسته^۳ به اسم نگاشت^۴ از فضای ورودی برای داده‌های آموزشی فراهم کند. شکل (۳-۷) نگاشت از فضای با ابعاد بالا به فضای با ابعاد کم را نشان می‌دهد. نگاشت خود سامان‌ده از سایر شبکه‌های عصبی که از یادگیری برپایه تصحیح خطا استفاده می‌کنند، متفاوت است؛ به این معنا که این نگاشت از یک تابع همسایگی برای نگهداشتن خواص توپولوژیکی فضای ورودی استفاده می‌کند. در SOM هدف خوشه‌بندی بدون نظارت کاربرد است. در این نگاشت هیچ اطلاعاتی از داده‌ها نداریم و تنها می‌خواهیم به خوشه‌بندی کور داده‌ها براساس ویژگی‌های مشترک آن‌ها بپردازیم.

^۱ Self Organizing Map

^۲ Unsupervised learning

^۳ Discrete

^۴ Map



شکل ۳-۷: نمایش نگاشت از فضای با ابعاد بالا به فضای با ابعاد کم

شبکه عصبی مصنوعی نخستین بار توسط پروفسور تئوو کوهن^۱ در سال ۱۹۸۲ معرفی شد که آن را گاهی نگاشت کوهن^۲ یا شبکه کوهن^۳ نیز می‌نامند [۴۳]. شبکه کوهن خلاصه‌ای از محاسباتی را فراهم می‌آورد که بر پایه کارهای بیولوژیکی مدل‌های عصبی دهه ۱۹۷۰ است [۴۴] که توسط ویل شو و ون در مالسبرگ در سال ۱۹۷۶ بیان شده بود و نیز مدل‌های مورفوژنی که از نظر تاریخی به دهه ۱۹۵۰ و آلن ترینگ برمی‌گردد [۴۵]. این نگاشت می‌تواند رابطه بین همسایگی‌ها را در داده‌های ورودی حفظ کند [۴۶].

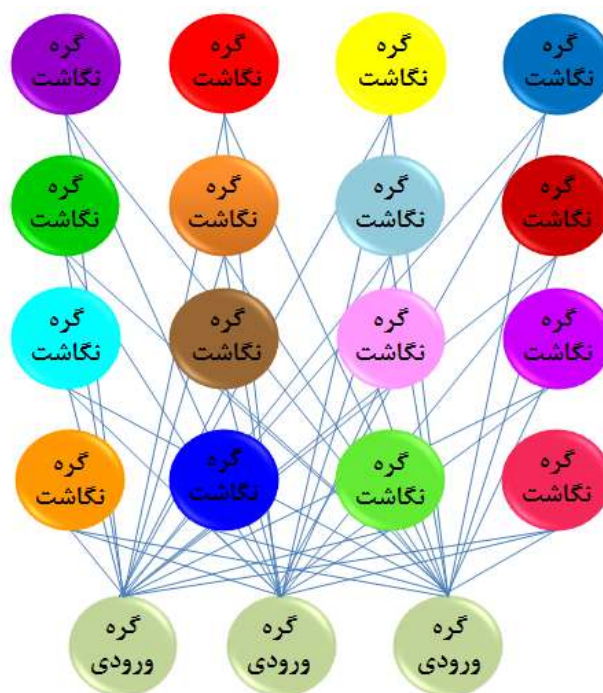
۳-۴-۲ ساختار نگاشت خود سامان‌ده

یک نگاشت خود سامان‌ده شامل عناصری است که گره یا نرون نامیده می‌شوند. تعداد نرون‌های لایه ورودی در یک SOM برابر با ابعاد بردار ورودی یا ویژگی‌های داده است و تعداد نرون‌های لایه خروجی برابر با تعداد دسته‌هایی (خوشه‌هایی) است که فضا به این دسته‌ها کوانتیزه می‌شود. این

¹ Teuvo Kohonen
² Kohonen map
³ Kohonen network

شبکه به اصطلاح یک شبکه عصبی مصنوعی کاملاً متصل^۱ نامیده می‌شود [۴۷]. طبق آنچه در شکل (۸-۳) دیده می‌شود، از هر نرون لایه ورودی به تمام نرون‌های لایه خروجی با استفاده از یک وزن اتصالی وجود دارد. بنابراین برای هر نرون لایه خروجی یک بردار وزن به ابعاد تعداد نرون‌های لایه ورودی وجود دارد. تعداد نرون‌های لایه خروجی برابر با تعداد دسته‌ها است.

مثل بیشتر شبکه‌های عصبی مصنوعی، SOMها به دو طریق عمل می‌کنند: آموزش و نگاشت. آموزش، نگاشت را با استفاده از مثال‌های ورودی می‌سازد. این شبکه شبیه شبکه‌های عصبی ساده است با این تفاوت که در SOM بر خلاف شبکه‌های دیگر، توپولوژی لایه خروجی می‌تواند یک بعدی یا دو بعدی باشد. در این شبکه زمانی که یک نرون برنده می‌شود، نرون‌های همسایه علاقمندند به‌روزرسانی شوند و این تمایل با فاصله از نرون برنده نسبت معکوس دارد [۴۷].



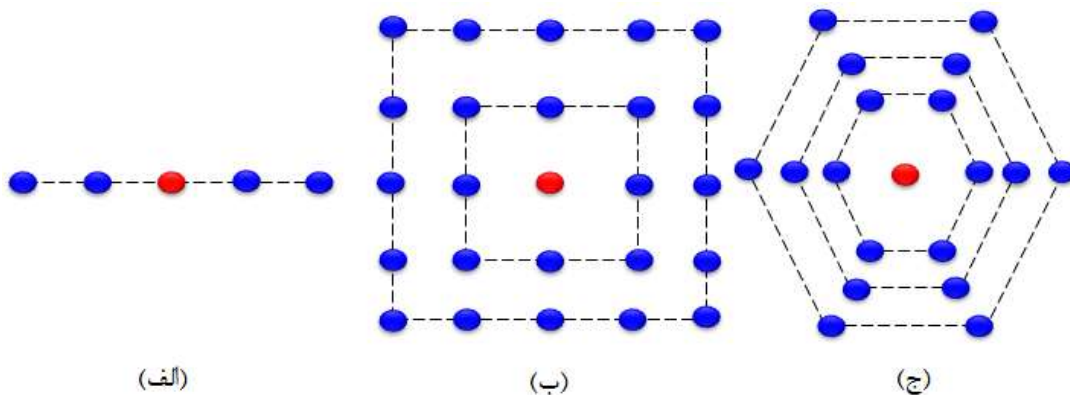
شکل ۸-۳: ساختار شبکه SOM

در هر SOM یک وزن مرتبط با هر نرون وجود دارد که بعدی برابر با بُعد بردارهای داده ورودی و

^۱ Fully connected artificial neural network

موقعیتی در فضای نگاشت دارد. آرایش معمول گره‌ها، به صورت خطی در فضای یک بعدی یا دو بعدی در یک شبکه مربعی^۱ یا شش ضلعی^۲ است. شکل (۳-۹) این آرایش‌ها را در شبکه SOM نشان می‌دهد.

ساختار طبیعی این شبکه‌ها به صورت پیش‌خور^۳ و کاملاً متصل^۴ است و در آن‌ها هیچ حلقه‌ای^۵ وجود ندارد. هر گره موجود در هر لایه کاملاً متصل به گره‌های موجود در لایه بعدی است. هر اتصالی که بین دو گره در این شبکه وجود دارد، وزنی دارد که به طور تصادفی بین صفر و یک است. تنظیم این وزن‌ها مهم‌ترین بخش فرآیند یادگیری است. SOMها لایه مخفی^۶ ندارند و در آن‌ها اطلاعات به صورت مستقیم از ورودی به خروجی منتقل می‌شوند. در SOMها گره‌هایی که نزدیک به هم قرار دارند، از نظر موقعیت در صفحه تصویر و میزان حرکت به سمت نرون برنده، نسبت به گره‌هایی که از هم دور هستند، شباهت بیشتری به یکدیگر دارند.

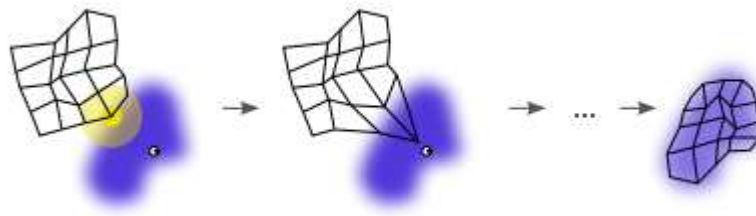


شکل ۳-۹: تعریف‌های همسایگی: الف) خطی، ب) مربعی، ج) شش ضلعی. همسایگی در اطراف یک نرون برنده (نرون قرمز، نرون برنده و نرون‌های آبی همسایگی‌ها را نشان می‌دهند).

¹ Square
² hexagonal
³ Feedforward
⁴ Completely connected
⁵ Loop
⁶ Hidden layer

۳-۴-۳ الگوریتم یادگیری

نگاشت خود سامان‌ده در ابتدا برای تجسم داده‌های با ابعاد بالا و برای تولید گرافی مشابه داده‌های ورودی مطرح شد. این نگاشت معمولاً شامل آرایه‌ای دو بعدی است که در آن هر گره به یک بردار ویژگی مرتبط است [۴۸]. آموزش از یادگیری رقابتی استفاده می‌کند. هنگامی که یک مثال آموزش به شبکه تغذیه می‌شود، فاصله اقلیدوسی آن با همه بردارهای وزن محاسبه می‌شود. گره‌های خروجی برای رسیدن به بهترین نتیجه با هم رقابت^۱ می‌کنند. نرونی که بردار وزن آن بیشتر شبیه ورودی است، بهترین واحد تطبیق^۲ (BMU) نامیده می‌شود و این گره برنده مرکز یک همسایگی می‌شود. گره‌های خروجی در همسایگی گره برنده، هر یک سهمی از میزان تحریک^۳ گره برنده را خواهند داشت. وزن‌های BMU و نرون‌های نزدیک به آن در شبکه SOM به سوی بردار ورودی تنظیم می‌شوند. اندازه تغییر، با زمان و با فاصله از BMU در شبکه کاهش می‌یابد.



شکل ۳-۱۰: شرح آموزش یک نگاشت خود سامان‌ده [۴۹].

در شکل (۳-۱۰) حباب آبی توزیع داده‌های آموزشی و شبکه سفید وضعیت کنونی داده‌های آموزشی را نشان می‌دهد. در ابتدا گره‌های SOM به صورت تصادفی در فضای داده قرار می‌شوند. گره‌ای که با رنگ زرد نشان داده شده است، نزدیک داده آموزشی قرار دارد. این گره به سمت داده آموزشی حرکت می‌کند. به همین ترتیب همسایه‌های آن نیز تمایل دارند به سمت داده آموزشی

¹ Competition

² Best Matching Unit

³ Motivation

حرکت کنند. میزان تمایل آن‌ها با فاصله از داده آموزشی نسبت عکس دارد. بعد از چندین تکرار شبکه تمایل دارد تا توزیع داده‌ها را تخمین بزند.

در شبکه SOM هنگامی که نرونی در دوره t ام برنده می‌شود، یک تابع $D(t)$ تعریف شده است، که مشخص می‌کند به چه شعاع از همسایگی، وزن‌های نرون‌های همسایه نرون برنده به‌روزرسانی می‌شود [۴۷]. شکل (۱۱-۳) نرون‌های لایه خروجی یک بعدی را برای یک مثال نشان می‌دهد.



شکل ۱۱-۳: نرون‌های لایه خروجی یک بعدی

که اگر $D(t)=1$ باشد، زمانی که نرون B برنده شود، وزن‌های ورودی به A و B به‌روزرسانی می‌شوند. در حالی که توپولوژی خروجی دو بعدی باشد، تابع $D(t)$ شعاع همسایگی نرون خروجی را به عنوانی تابعی از زمان (تکرار) می‌کند. معمولاً شعاع ابتدا گسترده است که این باعث می‌شود، شبکه به صورتی کلی‌تر و عمومی‌تر^۲ رفتار کند، در ادامه به تدریج شعاع همسایگی کوچکتر شده و باعث می‌شود، شبکه رفتار محلی‌تری داشته باشد [۴۷].

۳-۴-۴ شبکه کوهنن

الگوریتم یادگیری نگاشت‌های خود سامان‌ده در حالت کلی مبتنی بر انتخاب نرون برنده و حرکت نرون مذکور و برخی از همسایگانش به سوی داده ورودی مورد نظر است. الگوریتم یادگیری نگاشت‌های خود سامان‌ده را می‌توان به مراحل زیر خلاصه نمود:

¹ Epoch

² Global

۱. مقداردهی اولیه: در این مرحله وزن هر نرون به صورت یک عدد تصادفی انتخاب می‌شود، سپس یک الگوی ورودی $X=(x_1, x_2, \dots, x_d)$ به شبکه اعمال می‌گردد. نرخ یادگیری اولیه و اندازه همسایگی نیز باید مشخص شوند.

۲. تعیین نرون برنده: در این مرحله براساس معیار تشابه شبکه، نرون برنده مشخص می‌شود. معیارهای تشابه مختلفی را می‌توان در نگاشت‌های خود سامان‌ده به کار گرفت، اما معمول‌ترین آن‌ها معیار فاصله اقلیدوسی است. رابطه فاصله اقلیدوسی مطابق رابطه زیر می‌باشد [۴۷]:

$$\text{Euclidean Distance: } D(W, X) = \|W - X\| = \sqrt{\sum_{i=1}^d (W_i - x_i)^2} \quad (۱۰-۳)$$

در این مرحله ورودی $X=(x_1, x_2, \dots, x_d)$ به طور همزمان با تمامی نرون‌های موجود در شبکه مقایسه می‌گردد. نرون برنده، نرونی است که کمترین فاصله را از تمامی الگوهای داده ورودی، داشته باشد.

۳. تعیین نرون‌های همسایه: بعد از مشخص شدن نرون برنده مجموعه‌ای از نرون‌های همسایه نرون برنده که باید مقادیرشان تغییر نمایند، مشخص می‌گردند. تغییر مقادیر مربوط به نرون‌های همسایه در حالت کلی به دو صورت انجام می‌پذیرد. در حالت اول یک شعاع همسایگی معین در اطراف سلول برنده انتخاب می‌گردد. در این روش تمامی نرون‌هایی از شبکه که در فاصله معین از نرون برنده می‌باشند، با یک ضریب ثابت به سمت ورودی حرکت می‌نمایند. در روش دوم، تمامی نرون‌های موجود در شبکه با ضریبی نابرابر به سمت ورودی حرکت می‌نمایند. این ضریب نابرابر به صورتی است که در نرون برنده حداکثر مقدار را داشته و با دور شدن از نرون برنده مقدارش کاهش پیدا می‌کند.

۴. اصلاح وزن‌ها: در انتها وزن‌های مربوط به نرون برنده و همسایگانش می‌بایست براساس ورودی شبکه اصلاح گردند. این تغییرات براساس رابطه زیر صورت می‌گیرد [۴۷]:

$$W_j(t+1) = W_j(t) + \eta(t)(x_n - W_j(t)) \quad (11-3)$$

نرخ یادگیری $\eta(t)$ معمولاً با گذشت زمان به طور خطی کاهش می‌یابد:

$$\eta(t) = e_d \eta_0 \left(1 - \frac{t}{T}\right) \quad 1 \leq t \leq T \quad (12-3)$$

$$0 \leq \eta(t) \leq \eta(t-1) \leq 1 \quad (13-3)$$

که η_0 نرخ یادگیری اولیه است، T تعداد کل بردارهای ویژگی، t نشان‌دهنده t امین ارائه همه بردارهای ویژگی و e_d تحریک جانبی وابسته به موقعیت بردار وزن انتخاب شده و بردار وزنی است که باید به روزرسانی شود [۴۷].

۳-۵ جمع‌بندی و نتیجه‌گیری

در این فصل به تشریح الگوریتم‌های مورد استفاده در این پایان‌نامه پرداختیم. در روش پیشنهادی خود از دو الگوریتم جابه‌جایی میانگین و نگاشت خود سامان‌ده استفاده می‌کنیم تا فضای رنگ تصویر را کوانتیزه کنیم.

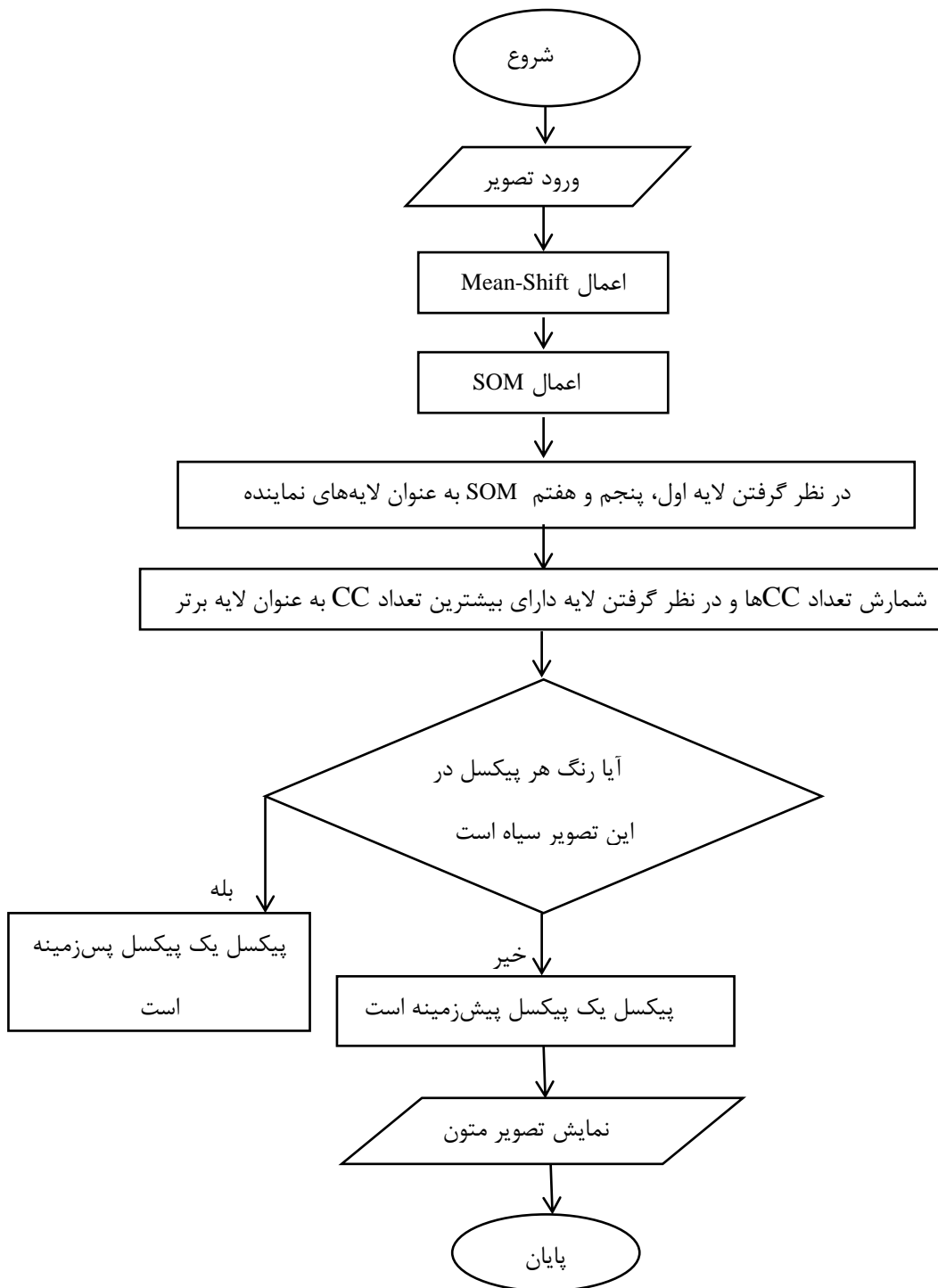
فصل چهارم

روش پیشنهادی و پیاده‌سازی آن

در این پایان‌نامه تلاش شده است تا روشی خودکار جهت جداسازی متون فارسی و انگلیسی از نقشه‌های جغرافیایی ارائه گردد که کارایی بهتری نسبت به روش‌های موجود داشته باشد. وجود نقطه‌ها، سرکش‌ها، مدها، بالارونده‌ها و پایین‌رونده‌ها، ساختار زبان فارسی را از زبان لاتین متمایز می‌کند. بنابراین روش‌های موجود در مورد نقشه‌های لاتین، به صورت مستقیم روی نقشه‌های فارسی قابل استفاده نیستند. تعداد روش‌های موجود در زمینه استخراج متن فارسی از نقشه‌های جغرافیایی نیز بسیار کم بوده و دقت این روش‌ها قابل قبول نمی‌باشد (روش‌های [۲۴] و [۲۵]). علاوه بر این، در آن‌ها از ویژگی رنگ نقشه‌ها استفاده نمی‌شود. هدف ما ارائه روشی است که در آن بتوان متون، مخصوصاً متون فارسی، را از تصویر رنگی نقشه جغرافیایی به گونه‌ای استخراج کرد که کمترین خطا را به همراه داشته باشد به عبارت دیگر حداکثر تعداد کلمه‌های ممکن در نقشه استخراج شوند و تا حد ممکن خطوط جاده‌ها یا سایر نمادهای گرافیکی موجود در نقشه که غیرمتنی محسوب می‌شوند، فیلتر شوند. سعی ما بر این است که روش پیشنهادی برای نقشه‌های دارای متون لاتین، فارسی و نیز ترکیبی از این دو با قلم‌ها، رنگ‌ها، اندازه‌ها و جهات مختلف قابل استفاده باشد. همچنین تلاش می‌کنیم تا کلمات را به صورت جداگانه از لایه متنی استخراج کرده، آن‌ها را به صورت افقی چرخانده و برای ورود به OCR آماده سازیم.

در شکل (۴-۱) روندنمای^۱ مراحل انجام روش پیشنهادی برای استخراج لایه متنی دیده می‌شود. در این فصل مراحل پیاده‌سازی الگوریتم پیشنهادی خود را به ترتیب شرح می‌دهیم، نقش هر یک از الگوریتم‌های تعریف شده در فصل قبل را روی کار خود بررسی کرده و نتیجه اعمال آن‌ها را روی یک نقشه فارسی، یک نقشه لاتین و یک نقشه با متون دارای رنگ‌های مختلف به نمایش می‌گذاریم.

¹ Flowchart



شکل ۴-۱: روندنمای مراحل پیاده‌سازی الگوریتم جهت استخراج لایه متنی

شکل (۴-۲) بخشی از نقشه شهر تهران است که مراحل اجرای روش پیشنهادی برای نمونه بر روی آن گزارش می‌شود. در ابتدا سیستم نقشه ورودی را می‌گیرد. سپس روش جابه‌جایی میانگین روی آن اعمال می‌شود.



شکل ۴-۲: بخشی از نقشه شهر تهران [۸]

۴-۲ اعمال روش جابه‌جایی میانگین

در ابتدا جهت کاهش اثر نویز، هموارسازی^۱ رنگ موجود در یک ناحیه بر روی نقشه و حفظ ساختار نواحی متنی، روش جابه‌جایی میانگین را اعمال می‌کنیم. روش جابه‌جایی میانگین، رابطه بین رنگ‌ها در فضای رنگ (موقعیت پیکسل‌ها) و فضای تصویر را در نظر می‌گیرد و در صدد تغییر رنگ پیکسل‌های یک خوشه به رنگ میانگین آن خوشه بر می‌آید [۴۱]. چون فضای HSI (فام^۲، اشباع^۳ و روشنایی^۴) نمایش بهتری از فضای رنگ فراهم می‌کند، در خوشه‌بندی از این فضای رنگ استفاده شده است. اگر پیکسل واقع در مختصات (x,y) را با $P(x,y)$ نشان دهیم، S ، H و I اجزای رنگ پیکسل P می‌باشند.

¹ Smoothing

² Hue

³ Saturation

⁴ Intensity

در روش جابه‌جایی میانگین هر پیکسل مثل $P(x,y)$ در فضای δ بعدی x, y, h, s و i به صورت یک نقطه در نظر گرفته می‌شود. دو جزء اول مختصات پیکسل و سه جزء بعدی مؤلفه‌های رنگ پیکسل در فضای HSI می‌باشند. الگوریتم جابه‌جایی میانگین از هر نقطه از فضای ویژگی (۵ بعدی) در جهت افزایش چگالی نقاط حرکت نموده تا به نقطه‌ای از فضا که در آن همسایگی با حداکثر چگالی موجود است، برسد. این نقطه مُد محلی بوده و رنگ این نقطه به تمام پیکسل‌هایی که طی فرآیند جابه‌جایی میانگین به این نقطه می‌رسند، داده خواهد شد. به این ترتیب رنگ پیکسل‌های اطراف یک ناحیه به رنگ میانگین آن ناحیه تغییر می‌یابد.

شکل (۳-۴) نتیجه اعمال الگوریتم جابه‌جایی میانگین روی شکل (۲-۴) را نشان می‌دهد. همان‌طور که در شکل (۳-۴) مشاهده می‌شود، بسیاری از سایه‌های رنگ‌ها که در شکل (۲-۴) موجود بودند، از بین رفته‌اند و با خوشه‌های اصلی رنگ‌های اطراف خود یکی شده‌اند. همچنین، رنگ‌هایی که تعداد پیکسل‌های کمی به آن‌ها اختصاص یافته مانند کوچه‌ها و خیابان‌های سفید رنگ در خوشه‌های اطراف ادغام شده‌اند. این کاهش در تعداد رنگ‌ها به گونه‌ای است که به راحتی توسط چشم قابل مشاهده می‌باشد. بعضی از رنگ‌ها مانند متون سیاه رنگ یا ناحیه آبی رنگ با این که نسبت به خوشه‌های اطراف تعداد پیکسل‌های کمتری دارند، اما همچنان در تصویر حاصل از اعمال نگاشت خود سامان‌ده باقی مانده‌اند؛ علت این امر این است که مراکز این خوشه‌ها با مراکز خوشه‌های اطراف آن‌ها فاصله زیادی دارد.

پارامترهای این الگوریتم به گونه‌ای تنظیم شده‌اند که اگر کمتر از مقدار تعیین‌شده باشند، سرعت پردازش افزایش می‌یابد و انجام فرآیند زمان کمتری لازم خواهد داشت، اما تعدادی از نواحی نویزی و خوشه‌های دارای پیکسل‌های کم که تأثیری در یافتن متون ندارند، همچنان در لایه نتیجه باقی خواهند ماند. اگر مقدار پارامترها بیشتر از مقدار کنونی باشد نتیجه حاصل از آن تغییر خاصی با نتیجه کنونی نخواهد داشت و تنها زمان پردازش طولانی‌تر می‌شود. بنابراین انتخاب مناسب پارامترها در

کسب نتیجه مطلوب تأثیر مهمی دارد.



شکل ۳-۴: نتیجه اعمال الگوریتم جابه‌جایی میانگین روش شکل ۲-۴

تعداد تکرارهای الگوریتم جابه‌جایی میانگین به توزیع داده‌ها در فضای رنگ وابسته است که بعد از آن مراکز خوشه‌ها تغییر خاصی نمی‌کند. تعداد خوشه‌ها نیز به تعداد رنگ‌های موجود در نقشه بستگی دارد و از پیش تعیین نشده است. ما سعی می‌کنیم این تعداد را در الگوریتم SOM به ۸ مورد کاهش دهیم.

۳-۴ اعمال روش نگاشت خود سامان‌ده

نگاشت خود سامان‌ده یک شبکه عصبی مصنوعی با یادگیری بدون نظارت است. این شبکه شامل لایه‌های ورودی و خروجی است و لایه مخفی ندارد. تعداد نرون‌های لایه ورودی برابر با ابعاد بردار ویژگی است و تعداد نرون‌های لایه خروجی برابر با تعداد خوشه‌هایی است که فضا به آن‌ها کوانتیزه می‌شود.

الگوریتم یادگیری SOM گره برنده را می‌یابد و وزن آن را به‌روزرسانی می‌کند. در این زمان گره‌های همسایه گره برنده هم به تبعیت از آن به‌روزرسانی می‌شوند. به این معنا که SOM یک الگوی طبیعی‌تر ایجاد می‌کند و بعضی از رنگ‌های اصلی گره‌های همسایه، از دست می‌رود. در الگوریتم پیشنهادی ما ورودی SOM همان فام، اشباع و روشنایی در الگوریتم جابه‌جایی میانگین هستند. رنگ‌های با تکرار بالا، بارها خوانده می‌شوند و وزن‌ها را به طور مکرر به‌روزرسانی می‌کنند. این بدان معناست که رنگ‌های با تکرار بالا، روی گره‌های همسایه تأثیر بیشتری دارند؛ زیرا در هنگام به‌روزرسانی آن‌ها، همسایه‌هایشان نیز به روزرسانی می‌شوند. در نتیجه رنگ‌های با تکرار پایین به تدریج ناپدید شده و رنگ‌های نهایی، مشابه می‌شوند [۵۰].

استفاده از این نگاشت تصویر را به تعدادی لایه که برابر با تعداد خوشه‌های^۱ انتخابی است، کوانتیزه می‌کند. هر یک از الگوهای رنگ یک خوشه را نشان می‌دهند که در آن همه رنگ‌های تصویر به صورت بهینه نگاشت داده شده‌اند. در الگوریتم ما، تعداد خوشه‌ها هشت در نظر گرفته شده است؛ بنابراین هشت لایه خواهیم داشت که این لایه‌ها در شکل‌های (۴-۴) تا (۴-۱۱) نشان داده شده‌اند. کوانتیزاسیون رنگ در الگوریتم نگاشت خود سامان‌ده به گونه‌ای است که توسط چشم ملموس نیست. این کوانتیزاسیون باعث می‌شوند تشخیص متون راحت‌تر شود و بتوان از طریق رنگ، آن‌ها را از نمادهای گرافیکی و خطوط جاده جدا کرد. همان‌طور که از شکل‌های (۴-۴) تا (۴-۱۱) مشاهده می‌شود، تنها یکی از لایه‌های حاصل از SOM دارای عناصر متنی می‌باشد و آنچه در سایر لایه‌ها دیده می‌شود، تنها نمایانگر رنگ یا ناحیه‌ای خاص از شکل (۴-۳) حاصل از اعمال جابه‌جایی میانگین است.

^۱ Codebook



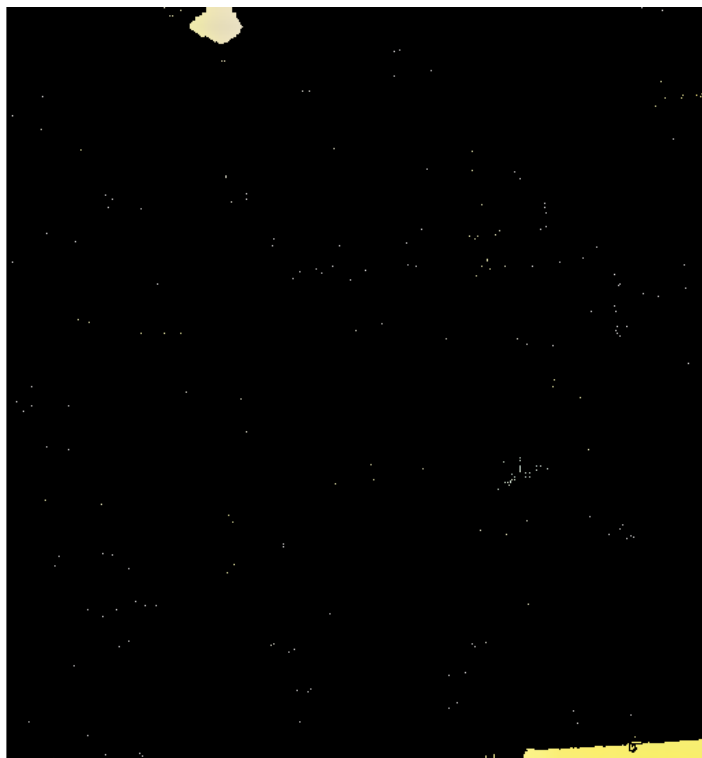
شکل ۴-۴: لایه اول حاصل از اعمال SOM روی شکل ۳-۴



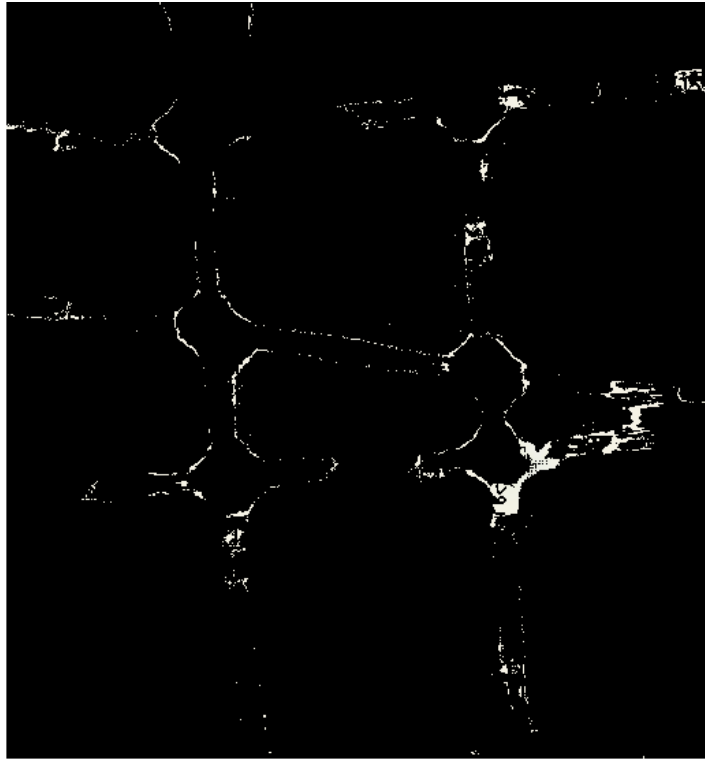
شکل ۴-۵: لایه دوم حاصل از اعمال SOM روی شکل ۳-۴



شکل ۴-۶: لایه سوم حاصل از اعمال SOM روی شکل ۳-۴



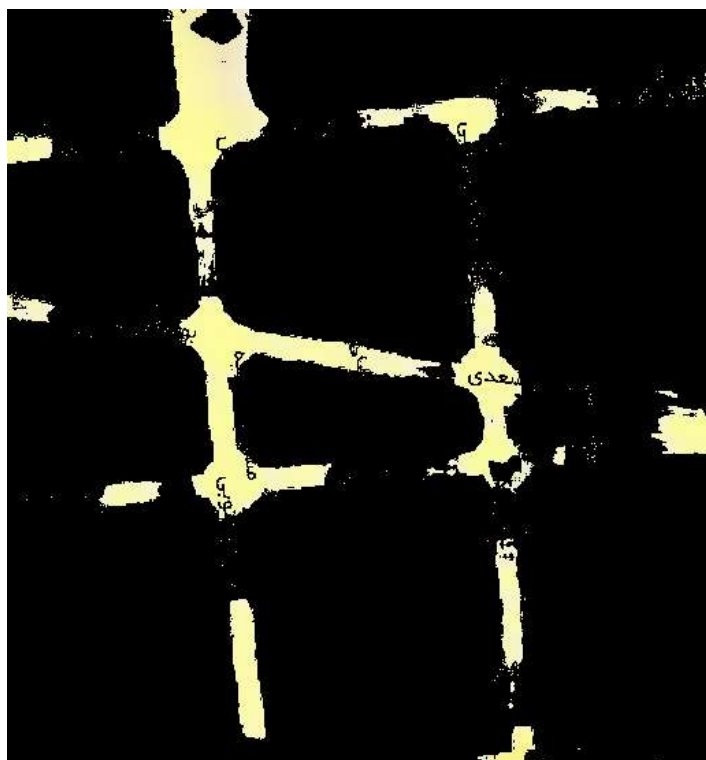
شکل ۴-۷: لایه چهارم حاصل از اعمال SOM روی شکل ۳-۴



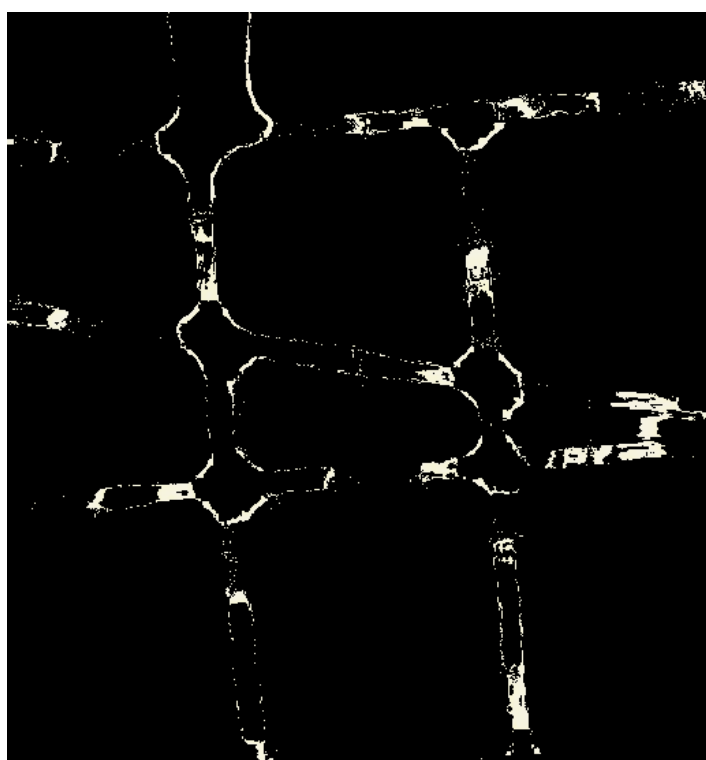
شکل ۴-۸: لایه پنجم حاصل از اعمال SOM روی شکل ۴-۳



شکل ۴-۹: لایه ششم حاصل از اعمال SOM روی شکل ۴-۳



شکل ۴-۱۰: لایه هفتم حاصل از اعمال SOM روی شکل ۳-۴

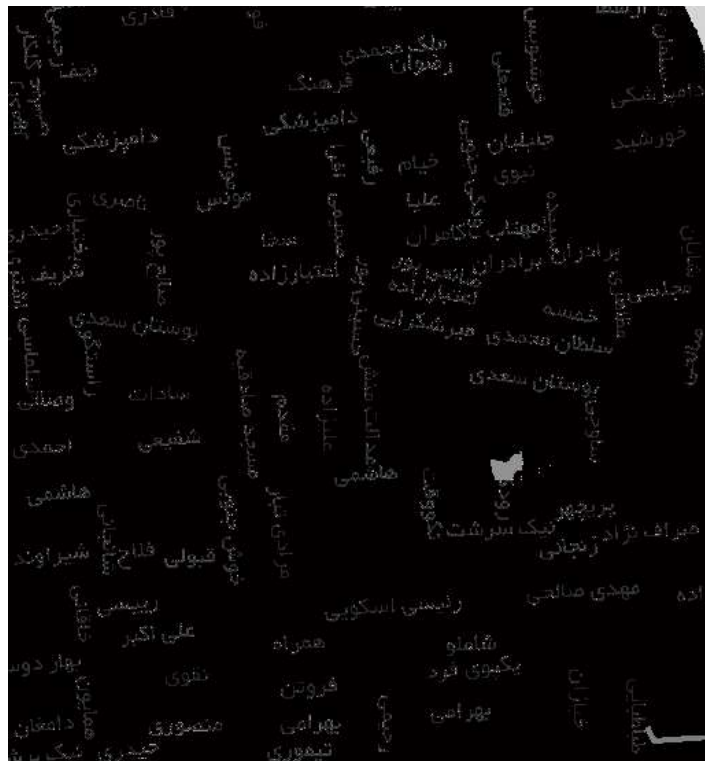


شکل ۴-۱۱: لایه هشتم حاصل از اعمال SOM روی شکل ۳-۴

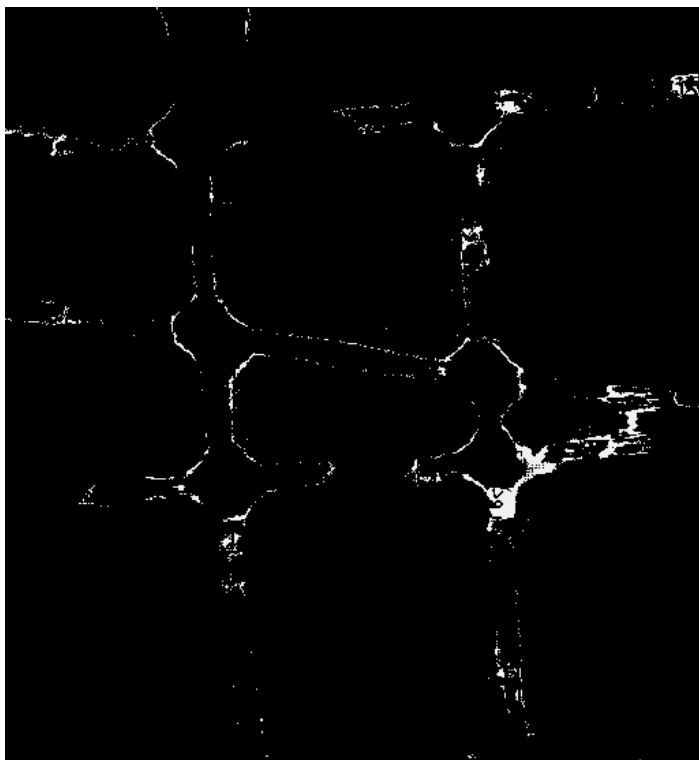
از آن جایی که هر دو روش جابه‌جایی میانگین و نگاشت خود سامان‌ده، تکنیک‌های بدون نظارتی هستند، بنابراین استفاده از آن‌ها نیازی به مداخله کاربر در انتخاب لایه متنی یا برجسب‌های متنی ندارد. بنابراین روش پیشنهادی ما یک روش خودکار است.

نتایج بررسی‌ها روی تعداد زیادی از نقشه‌ها نشان می‌دهد که در همه نقشه‌ها لایه‌های یک، پنج و هفت نماینده‌هایی برای لایه متنی هستند. لایه منتخب متنی باید به طریقی از بین این سه لایه نماینده انتخاب شود. برای این منظور ما از تحلیل اجزای متصل استفاده می‌کنیم. برای اجرای این تحلیل نیازمندیم تا لایه‌های نماینده را به لایه‌های سطح خاکستری تبدیل کنیم تا برای ورود به مرحله بعدی آماده شوند.

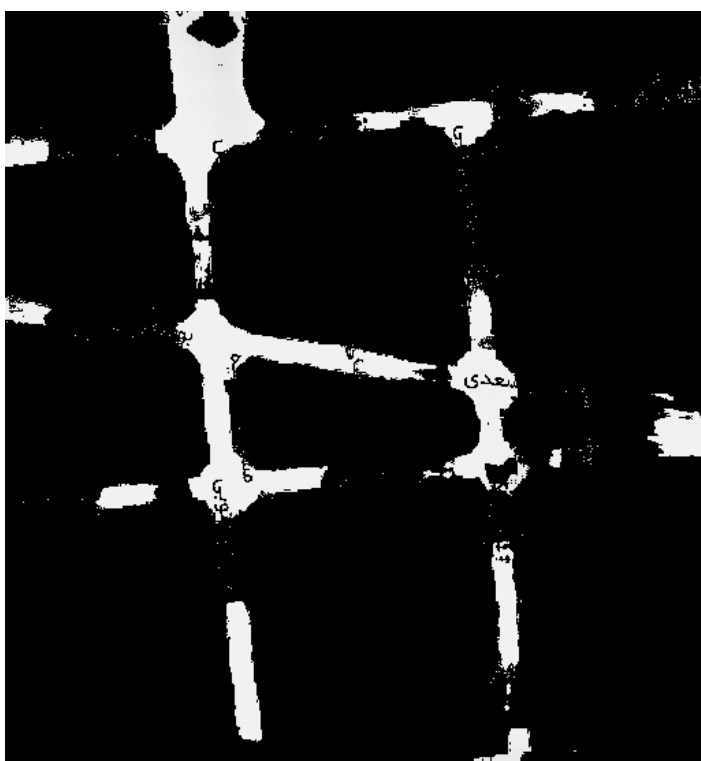
تصویر سطح خاکستری لایه‌های نماینده برای نقشه شکل (۲-۴) در شکل‌های (۴-۱۲) تا (۴-۱۴) نشان داده شده‌اند.



شکل ۴-۱۲: تصویر سطح خاکستری لایه اول SOM



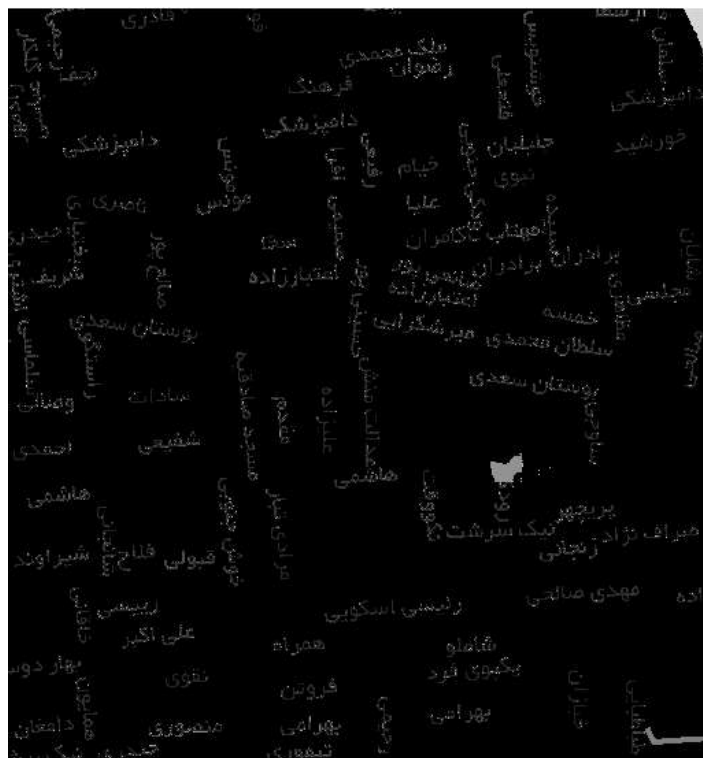
شکل ۴-۱۳: تصویر سطح خاکستری لایه پنجم SOM



شکل ۴-۱۴: تصویر سطح خاکستری لایه هفتم SOM

۴-۴ تحلیل اجزای متصل

در مرحله بعد در لایه‌هایی که به عنوان کاندیداهای متنی شناخته شده‌اند، تعداد عناصر متصلی را که مساحتی بزرگتر از پنج دارند، می‌شماریم. عدد پنج طی روش سعی و خطا و به این منظور در نظر گرفته شده است که معمولاً تعدادی از عناصر نویزی در کنار هم قرار می‌گیرند و به عنوان عنصر متصل شمارش می‌شوند. بنابراین ما از این قید استفاده می‌کنیم تا نویزها را در شمارش خود لحاظ نکنیم. اگر عددی کوچکتر از پنج به عنوان آستانه انتخاب شود، ممکن است لایه‌ای که دارای تعداد زیادی نویز است، به صورت اشتباه به عنوان لایه مطلوب در نظر گرفته شود. حال سیستم، لایه‌ای را که دارای بیشترین تعداد عناصر متصل است، به عنوان لایه متنی مورد نظر برای ورود به مرحله بعدی در نظر می‌گیرد. لایه منتخب پس از شمارش تعداد عناصر متصل در شکل (۴-۱۵) نشان داده شده است.



شکل ۴-۱۵: تصویر سطح خاکستری لایه منتخب

در تصویر سطح خاکستری هر یک از عناصر موجود در پیش‌زمینه بسته به رنگی که در تصویر اصلی داشته‌اند، دارای درجه‌ای از روشنایی خواهند بود و پس‌زمینه سیاه فاقد هرگونه اطلاعات متنی است. بنابراین، در گام بعدی برای یکسان‌سازی اطلاعات پیش‌زمینه و بهبود کیفیت نمایش آن‌ها، هر پیکسلی را که مقدار آن برابر با صفر است، یعنی سیاه است، به عنوان یک پیکسل پس‌زمینه^۱ در نظر می‌گیریم و آن را در تصویر لایه متنی با سیاه مشخص می‌کنیم. هر پیکسلی را که روشنایی دیگری جز سیاه دارد و مقدار آن غیرصفر است، به عنوان یک پیکسل پیش‌زمینه^۲ در نظر گرفته و در تصویر لایه متنی با آن را با مقدار یک و با رنگ سفید مشخص می‌کنیم. شکل (۴-۱۶) تصویر باینری حاصل شده را نشان می‌دهد که به عنوان لایه متنی توسط سیستم شناسایی شده است.



شکل ۴-۱۶: تصویر باینری لایه متنی

همان‌طور که از تصاویر فوق به خوبی مشخص است، روش پیشنهادی ما می‌تواند برچسب‌های

¹ Background pixel

² Foreground pixel

متنی را در جهات مختلف استخراج کند. ما روش را روی نقشه‌ای دیگر که حاوی متون با رنگ‌ها، اندازه‌ها و قلم‌های متفاوتی است، امتحان می‌کنیم. شکل (۴-۱۷) از سایت شهرداری تهران به دست آمده است و شکل‌های (۴-۱۸) تا (۴-۲۰) شکل‌های حاصل از مراحل اصلی انتخاب بهترین لایه متنی را نشان می‌دهند:



شکل ۴-۱۷: بخشی از نقشه شهر تهران [۵۱]

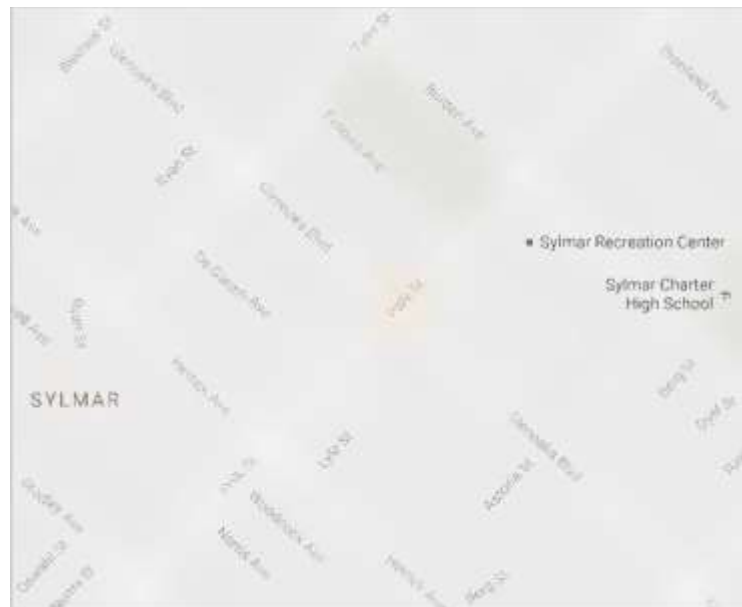


شکل ۴-۱۸: نتیجه اعمال روش جابه‌جایی میانگین روی شکل ۴-۱۷

تشخیص آن‌ها نیز به خوبی برآمده است. برای این که نشان دهیم، سیستم می‌تواند متون لاتین را نیز از نقشه‌ها استخراج کند، از شکل (۴-۲۱) که از سایت گوگل بدست آمده است و بخشی از نقشه شهر لس‌آنجلس آمریکا را نشان می‌دهد، استفاده می‌کنیم.



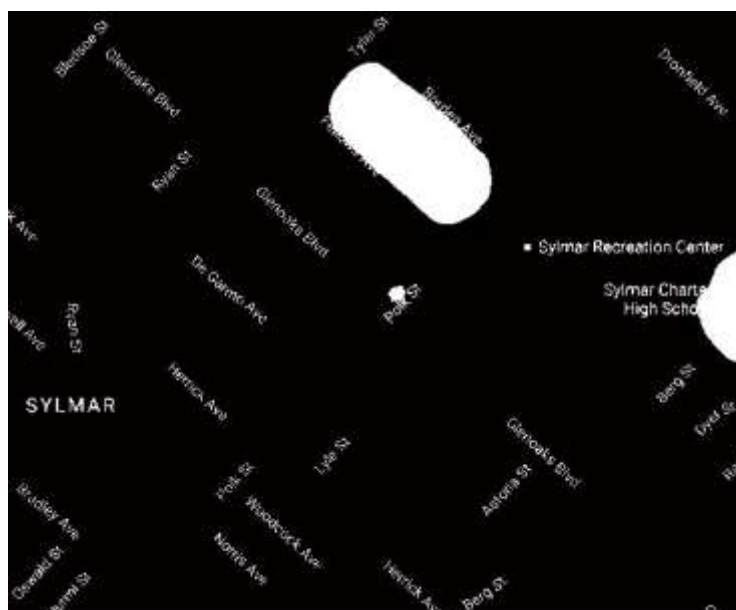
شکل ۴-۲۱: بخشی از نقشه شهر لس‌آنجلس [۱۱]



شکل ۴-۲۲: نتیجه اعمال روش جابه‌جایی میانگین روی شکل ۴-۲۱



شکل ۴-۲۳: لایه منتخب حاصل از اعمال روش نگاشت خود سامانده روی شکل ۴-۲۲



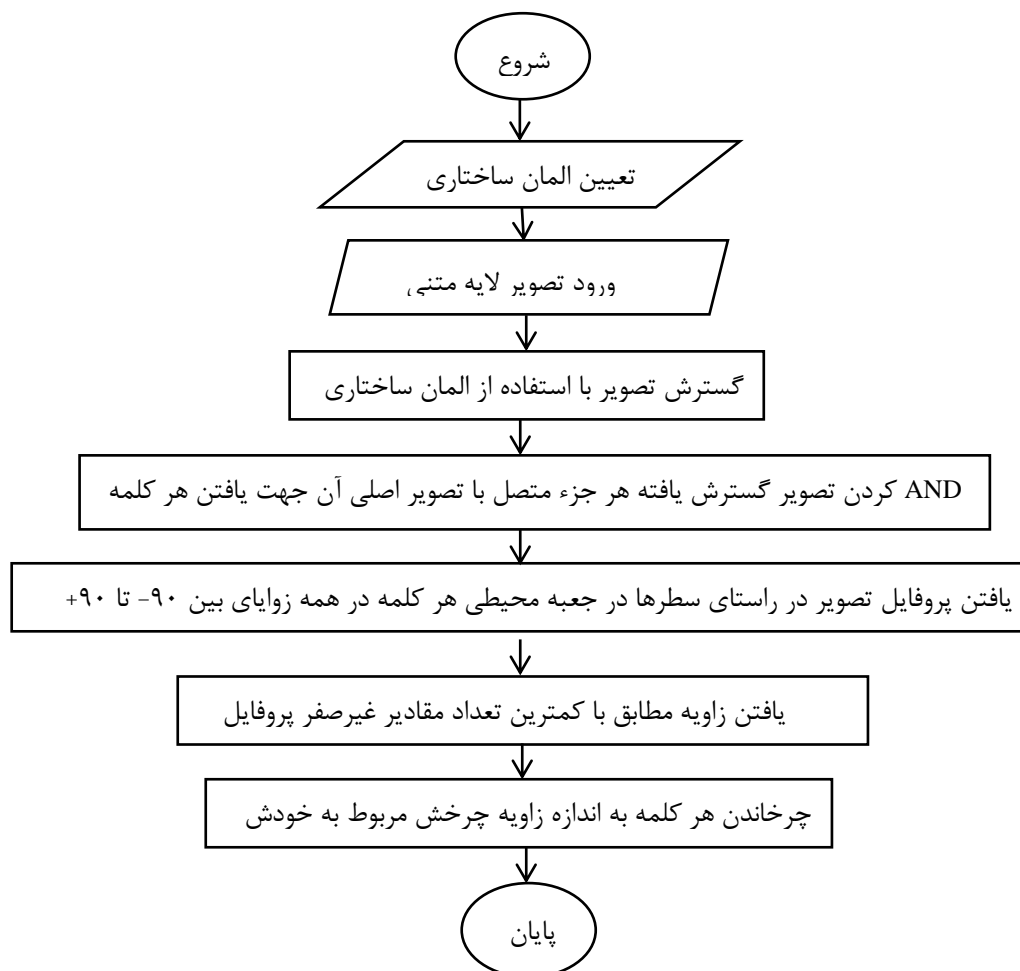
شکل ۴-۲۴: لایه متنی حاصل شده

همان طور که در شکل (۴-۲۴) ملاحظه می‌شود، الگوریتم پیشنهادی ما قابلیت استخراج متون لاتین را نیز دارد. در پاره‌ای از موارد ممکن است، برخی از عناصر گرافیکی در لایه متنی ظاهر شوند، مانند آنچه در شکل (۴-۲۴) دیده می‌شود. برای از بین بردن نواحی گرافیکی برجای مانده می‌توان

مجدداً با استفاده از تحلیل عناصر متصل، یک فیلتر مساحت تعریف کرد و عناصری را که اندازه‌ای بزرگتر از یک آستانه مشخص دارند، از تصویر نهایی حذف نمود.

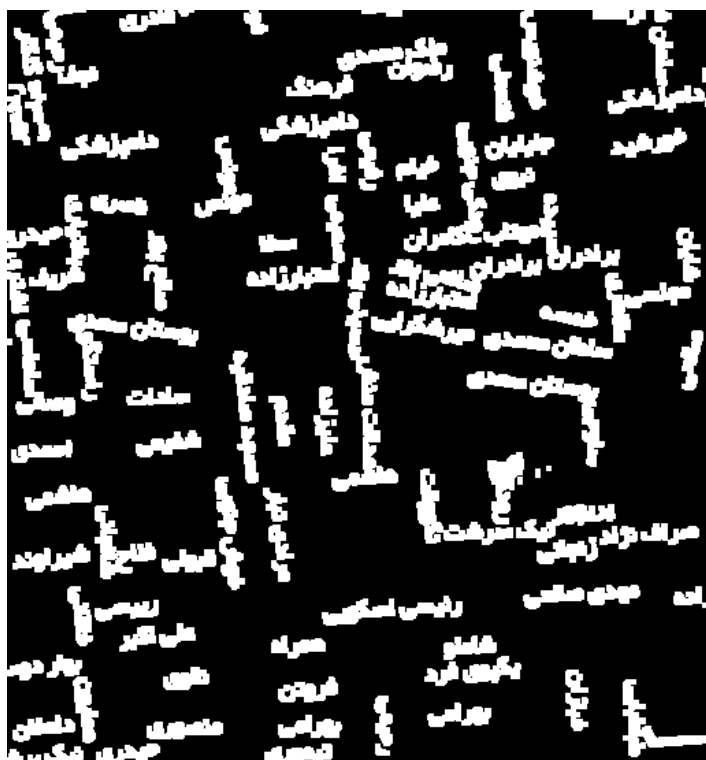
۴-۵ آماده‌سازی جهت ورود به OCR

مرحله بعدی آماده‌سازی متون برای ورود به نرم‌افزار تجاری OCR است. نرم‌افزار نویسه‌خوان نوری می‌تواند با استفاده از آموزش‌هایی که از قبل به آن داده شده است، کلمات را شناسایی کند. بنابراین کلمات موجود در لایه متنی باید به صورت جداگانه از نقشه جدا شده، به صورت افقی چرخیده و وارد OCR وارد شوند. شکل (۴-۲۵) روندنمای مراحل آماده‌سازی جهت ورود به OCR را نشان می‌دهد.



شکل ۴-۲۵: فلوچارت مراحل آماده‌سازی کلمات برای ورود به OCR

در ادامه الگوریتم پیشنهادی، با استفاده از یک المان ساختاری^۱ که به صورت یک ماسک^۲ چهار در چهار است و با استفاده از عملگر ریخت‌شناسی^۳ گسترش^۴، تصویر باینری حاصل از مرحله استخراج لایه متنی را گسترش می‌دهیم. علت این امر این است که اگر از تحلیل اجزای متصل روی تصویر لایه متنی استفاده شود، اجزای کلماتی که در نقشه وجود دارند، به صورت جداگانه در نظر گرفته می‌شوند. به طور مثال کلمه‌ای مانند کلمه "خیام" در شکل (۴-۱۶) به عنوان سه جزء در نظر گرفته می‌شود؛ حال آن‌که ما می‌خواهیم سیستم کل کلمه را به صورت یک جزء در نظر بگیرد. شکل (۴-۲۶) تصویر دودویی گسترش‌یافته را نشان می‌دهد.



شکل ۴-۲۶: تصویر دودویی گسترش‌یافته شکل ۴-۱۶

حال عناصر متصل موجود در تصویر گسترش‌یافته را در تصویر دودویی اصلی مرتبط با آن مؤلفه‌ها

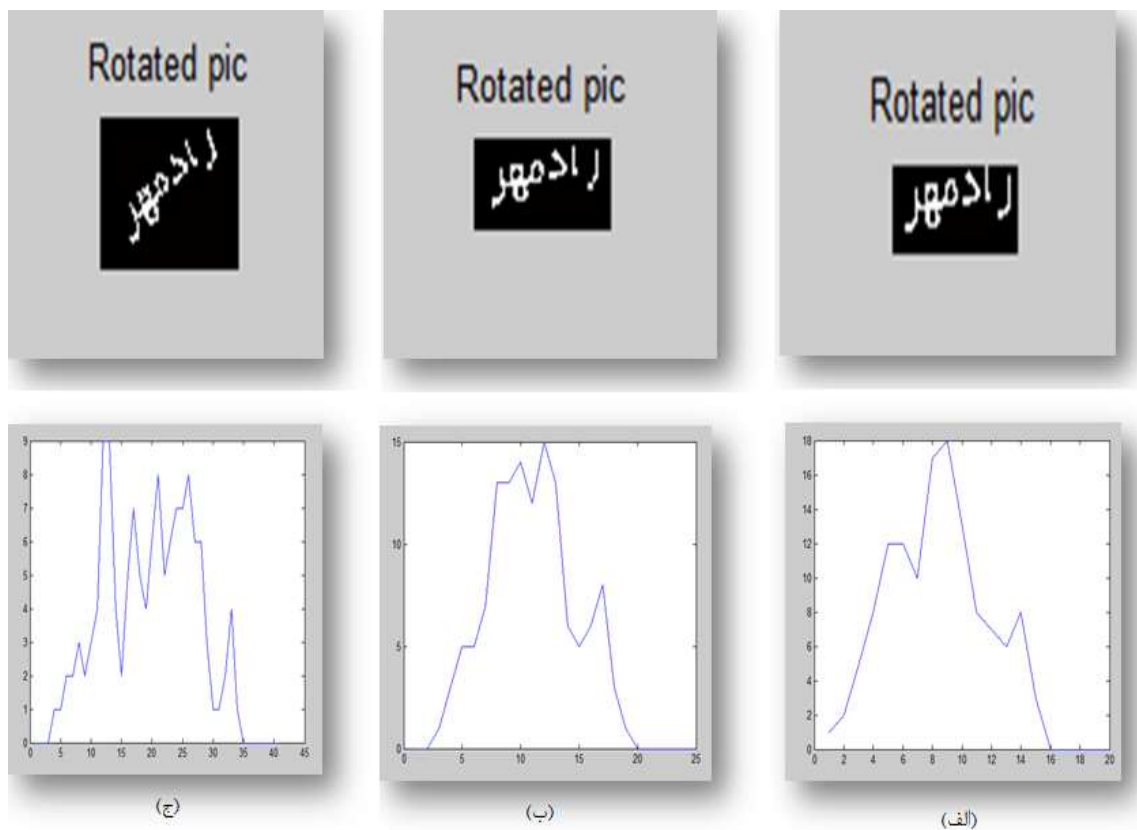
¹ Structure element

² Kernel

³ Morphological operator

⁴ Dilation

"AND" می‌کنیم تا تصویر هر کلمه به صورت جداگانه و واضح ظاهر شود. در گام بعدی از خاصیت جعبه محیطی^۱ استفاده می‌کنیم و جعبه محیطی اطراف هر کلمه را یافته، پروفایل تصویر را در راستای سطرها در جعبه محیطی هر کلمه در زوایای مختلف بین ۹۰- تا ۹۰+ می‌یابیم. همان طور که در شکل (۴-۲۷) دیده می‌شود، معمولاً اگر کلمه به صورت افقی باشد، عرض جعبه محیطی آن کمترین مقدار ممکن را خواهد داشت. در مورد (الف) در شکل (۴-۲۷) که کلمه تقریباً در جهت افقی است، پروفایل تصویر مقدار ۱۶ را دارد. در مورد (ب) چون شکل اندکی در جهت عمودی چرخیده، پروفایل تصویر مقدار ۲۰ را دارد و در مورد (ج) که میزان چرخش در جهت عمودی بیشتر است، مقدار پروفایل به ۳۵ رسیده است.

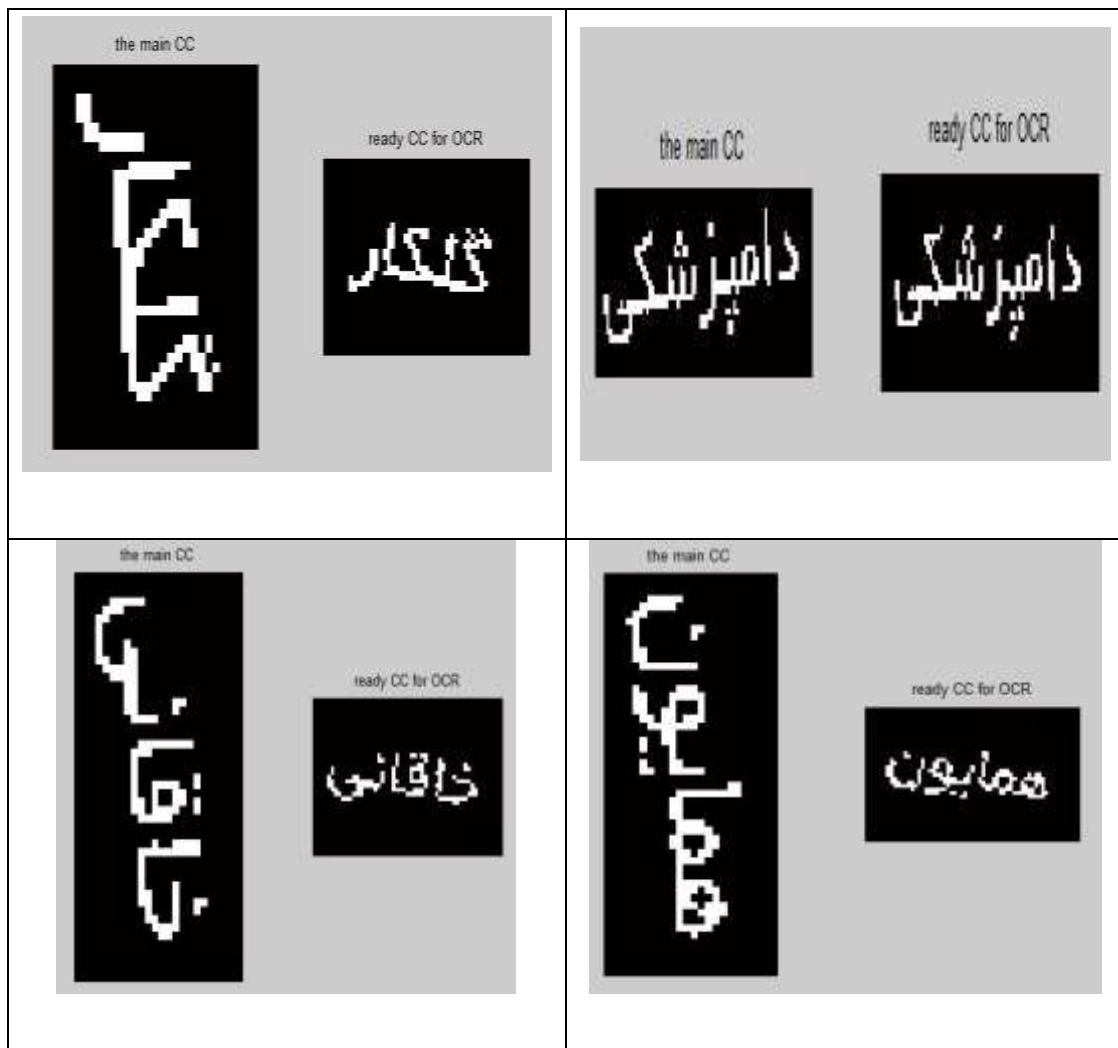


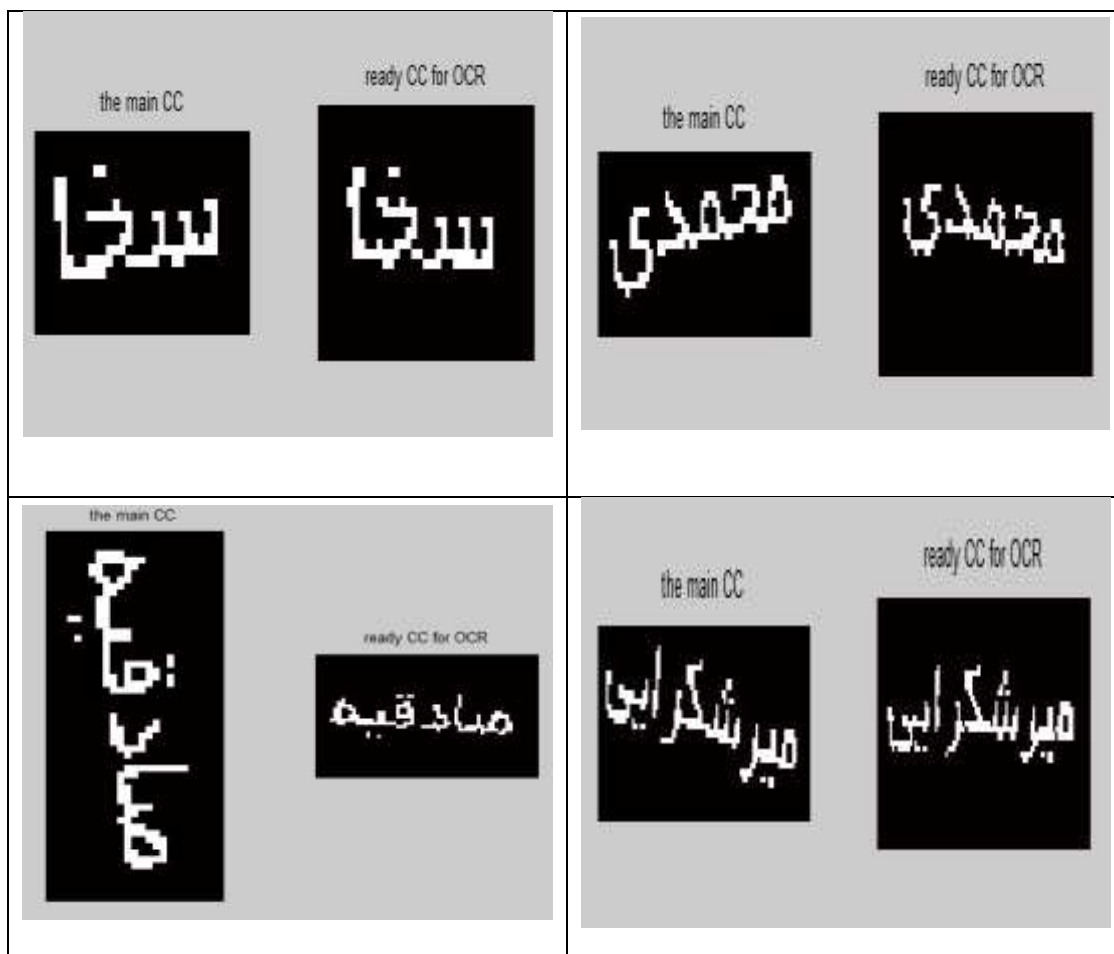
شکل ۴-۲۷: پروفایل تصویر در چند زاویه مختلف

ما هر زاویه‌ای را که در آن تعداد مقادیر غیر صفر پروفایل تصویر کلمه کمترین مقدار خود را داشته

^۱ Boundingbox

باشد، به عنوان زاویه چرخش مطلوب در نظر می‌گیریم. بنابراین، هر کلمه را به اندازه زاویه چرخش مطلوب خودش می‌چرخانیم تا در راستای افقی قرار بگیرد. شکل (۴-۲۸) نمونه‌هایی از کلمات استخراج شده آماده برای ورود به OCR از لایه متنی شکل (۴-۱۶) را نشان می‌دهد. در هر تصویر، بخش سمت چپ نشان‌دهنده تصویر دودویی و بخش سمت راست نشان‌دهنده تصویر چرخش یافته است.





شکل ۴-۲۸: نمونه‌هایی از کلماتی که توسط سیستم به صورت جداگانه شناسایی و چرخانده می‌شوند.

۴-۶ نتایج تجربی

در این تحقیق جهت ارزیابی روش پیشنهادی از ۴۰ نقشه که از سایت‌های گوگل^۱، Bing^۲، نقشه تهران^۳، کتاب اول^۴، شهرداری مناطق مختلف تهران (برای مثال منطقه دو^۵) و بعضی منابع دیگر بدست آمده‌اند، استفاده شده است. شکل (۴-۲۹) یک مورد از نقشه‌های مورد بررسی از هر منبع را نشان می‌دهد. همان‌گونه که می‌توان از موارد (الف) تا (ج) در شکل (۴-۲۹) مشاهده کرد، تراکم متون

¹ <https://www.google.com/maps>

² <https://www.bing.com/maps/>

³ <http://map.tehran.ir/>

⁴ <http://www.avval.ir/tehran-map>

⁵ <http://region2.tehran.ir/portals>

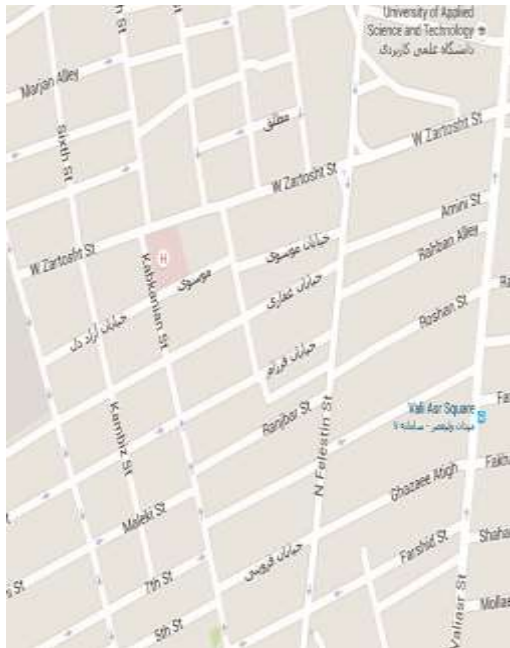
در نقشه‌های Bing، نقشه‌های گوگل و نقشه تهران از نقشه‌های کتاب اول، سایت شهرداری و اکثر نقشه‌های متفرقه کمتر است.



(ب)



(الف)



(د)



(ج)



(ی)



(ه)

شکل ۴-۲۹: مثالی از هریک از منابع نقشه مورد استفاده. الف) نقشه بدست آمده از Bing [۵۲] ب) نقشه بدست آمده از کتاب اول [۸] ج) نقشه بدست آمده از سایت شهرداری تهران [۵۱] د) نقشه بدست آمده از google maps [۱۱] ه) نقشه بدست آمده از Tehran maps [۵۳] ی) نقشه متفرقه‌ای که با جستجو در گوگل بدست آمده است.

برچسب‌های متنی در نقشه‌ها از راستای خطوط گرافیکی خیابان‌ها و کوچه‌ها تبعیت می‌کنند. بنابراین از آن جایی که خیابان‌ها و کوچه‌ها جهات مختلفی را دنبال می‌کنند، برچسب‌های متنی هم دارای جهت‌های متفاوتی هستند. از طرفی به علت فشردگی اطلاعات گرافیکی در این نوع نقشه‌ها، گاهی متون و خطوط گرافیکی با یکدیگر همپوشانی دارند که این امر جداسازی آن‌ها را دشوار می‌سازد. از طرف دیگر همه متون در یک نقشه، دارای یک رنگ مشخص یا یک قلم با اندازه مشخص نیستند. رنگ متون بسته به مکانی که به آن اشاره دارند، ممکن است از برچسب‌های متنی خیابان متفاوت باشد. اندازه عرض قلم هم بسته به عرض کوچه یا خیابان، می‌تواند در یک نقشه مقادیر متفاوتی داشته باشد. در مورد قلم مورد استفاده در نقشه‌ها هم باید خاطر نشان کرد که همه نقشه‌ها دارای یک استاندارد مشخص برای نوع قلم نیستند و نوع قلم ممکن است از نقشه‌ای به نقشه دیگر متفاوت باشد.

ما درصد دقت^۱ و فراخوانی^۲ را توسط فرمول‌های زیر محاسبه کرده‌ایم:

$$Precision = \frac{TP}{TP+FP} \quad (۱-۴)$$

$$Recall = \frac{TP}{TP+FN} \quad (۲-۴)$$

در این دو معیار TP^۳ (درست مثبت) به نواحی متنی اشاره دارد که سیستم می‌تواند آن‌ها را به درستی به عنوان متن تشخیص دهد. معیار FP^۴ (غلط مثبت) به نواحی غیرمتنی برمی‌گردد که سیستم به اشتباه آن‌ها را به عنوان متن شناسایی می‌کند و FN^۵ (غلط منفی) به نواحی متنی برمی‌گردد که سیستم به اشتباه نمی‌تواند آن‌ها را به عنوان متن شناسایی کند. ما هر کلمه کامل را به عنوان یک کاندید برای نواحی متنی و هر خط خیابان یا نماد گرافیکی را به عنوان یک ناحیه غیرمتنی می‌شماریم. اگر کلمه‌ای به صورت ناقص شناسایی شود و تنها بخش‌هایی از آن در لایه متنی موجود باشند (زیر کلمات)، ما آن‌ها را جزء معیار FN در نظر می‌گیریم و فرض می‌کنیم که سیستم موفق به شناسایی آن نشده است. نتایج آزمایشات ۹۰/۴٪ دقت و ۸۶/۶٪ فراخوانی را برای کلمات نشان می‌دهد. مسلماً اگر زیرکلمات را در شمارش خود محسوب کنیم، درصد دقت و فراخوانی افزایش می‌یابد. این نتایج تأیید می‌کند که رهیافت پیشنهادی ما دارای دقت خوبی در زمینه تشخیص متون فارسی است. باید توجه داشت که روش پیشنهادی تنها محدود به متون فارسی نیست و در مورد متون لاتین هم به خوبی جواب می‌دهد. همچنین این رهیافت مستقل از قلم، اندازه، جهت و رنگ متون است. جدول (۱-۴) ابعاد هر نقشه، منبع، زبان، درصد دقت و فراخوانی آن را روی هر ۴۰ نقشه به تفکیک نشان می‌دهد.

¹ Precision

² Recall

³ True Positive

⁴ False Positive

⁵ False Negative

جدول ۴-۱: منبع هر یک از ۴۰ نقشه مورد بررسی، ابعاد، زبان، درصد دقت و فراخوانی آن

شماره نقشه	منبع نقشه	ابعاد نقشه	زبان نقشه	درصد دقت	درصد فراخوانی
۱	کتاب اول	۵۳۱×۴۷۵	فارسی	۹۱/۴۳	۹۶/۹۷
۲	کتاب اول	۵۵۳×۵۱۵	فارسی	۹۷/۴۱	۱۰۰
۳	متفرقه	۴۵۵×۶۴۰	فارسی	۹۴/۳۲	۱۰۰
۴	Bing	۴۸۲×۷۱۸	لاتین	۹۳/۹۴	۸۸/۵۷
۵	گوگل	۴۴۱×۶۱۹	لاتین	۹۶/۴۳	۹۰
۶	کتاب اول	۳۵۱×۵۰۰	فارسی	۹۷/۳	۱۰۰
۷	شهرداری	۵۱۲×۶۴۷	فارسی	۸۱/۵۴	۹۶/۳۶
۸	متفرقه	۴۴۱×۶۳۱	ترکیب فارسی و لاتین	۹۲/۷	۹۵/۷
۹	شهرداری	۴۴۸×۶۴۴	فارسی	۷۵/۶۴	۹۸/۳۳
۱۰	شهرداری	۴۴۴×۵۸۶	فارسی	۸۰/۹۵	۸۵
۱۱	متفرقه	۳۶۲×۸۰۰	فارسی	۹۸/۴	۸۴/۴
۱۲	گوگل	۴۵۹×۵۷۳	ترکیب فارسی و لاتین	۹۷/۴۷	۱۰۰
۱۳	گوگل	۵۱۹×۷۱۶	ترکیب فارسی و لاتین	۹۲/۸۶	۱۰۰
۱۴	گوگل	۵۱۳×۶۶۶	ترکیب فارسی و لاتین	۸۹/۸	۱۰۰
۱۵	شهرداری	۴۵۲×۶۴۶	فارسی	۹۵/۷۸	۹۴/۴۴
۱۶	Tehranmaps	۳۷۱×۶۸۰	فارسی	۶۵	۷۲/۲۲
۱۷	گوگل	۴۶۷×۶۱۸	ترکیب فارسی و لاتین	۸۹/۸۷	۹۴/۶۷
۱۸	Tehranmaps	۳۸۲×۵۸۳	فارسی	۹۷/۵	۶۶/۱
۱۹	کتاب اول	۳۰۱×۴۰۰	فارسی	۸۸/۰۹	۹۲/۵
۲۰	کتاب اول	۴۱۰×۵۴۵	فارسی	۹۴/۴	۶۸
۲۱	متفرقه	۳۶۳×۵۵۰	فارسی	۹۵/۴۵	۹۹/۲۱
۲۲	کتاب اول	۳۲۱×۴۰۶	فارسی	۱۰۰	۹۴/۷۴

۸۰/۶۵	۷۳/۵۳	فارسی	۵۰۳×۶۴۰	شهرداری	۲۳
۱۰۰	۹۵/۶۵	فارسی	۴۴۵×۶۴۳	شهرداری	۲۴
۹۸/۳۹	۸۹/۷	فارسی	۵۲۱×۶۴۶	شهرداری	۲۵
۸۸/۳۳	۹۷/۲۵	فارسی	۵۱۴×۶۵۰	شهرداری	۲۶
۱۰۰	۹۶/۷۷	لاتین	۴۷۲×۵۷۴	گوگل	۲۷
۹۲/۰۶	۹۶/۶۷	فارسی	۴۱۳×۶۰۴	شهرداری	۲۸
۸۷/۶	۹۱/۸۷	فارسی	۵۱۹×۶۵۵	شهرداری	۲۹
۹۹/۱۷	۹۶/۷۷	فارسی	۲۹۲×۴۰۰	متفرقه	۳۰
۹۵/۴۵	۶۵/۶۲	فارسی	۲۹۲×۶۵۲	کتاب اول	۳۱
۹۷/۹۵	۹۷/۹۵	فارسی	۳۳۷×۳۶۴	متفرقه	۳۲
۸۱/۲۵	۸۳/۸۷	ترکیب فارسی و لاتین	۴۷۳×۶۴۳	گوگل	۳۳
۶۲	۸۷/۳۲	فارسی	۵۲۰×۶۴۳	شهرداری	۳۴
۹۸/۶۴	۷۳	فارسی	۴۴۵×۵۸۷	کتاب اول	۳۵
۹۵	۹۰/۴۸	فارسی	۵۰۸×۶۱۳	کتاب اول	۳۶
۷۰	۹۲/۴۵	ترکیب فارسی و لاتین	۴۸۶×۶۸۷	گوگل	۳۷
۹۲	۱۰۰	لاتین	۳۹۴×۵۸۳	گوگل	۳۸
۶۱/۲۹	۸۴/۴۴	ترکیب فارسی و لاتین	۴۹۹×۵۸۰	گوگل	۳۹
۶۸/۱۸	۹۳/۷۵	لاتین	۴۶۲×۵۱۶	Bing	۴۰

از آن جایی که تحقیقات روی متون فارسی بسیار محدود و نتایج آن‌ها ضعیف است و از طرفی کد این الگوریتم‌ها در مقالات مربوطه در دسترس نیست [۲۴ و ۲۵]، امکان مقایسه روش مطرح شده با آن‌ها وجود ندارد. اما درصد دقت ارائه شده در کار [۲۴] برای استخراج کلمات متون فارسی ۶۰٪ و در کار [۲۵] ۷۷/۳٪ بود که در کار ما به ۸۸/۶۲٪ رسیده است.

تحقیقات انجام شده بر روی متون لاتین هم نشان‌دهنده ضعف اکثر آن‌ها در تشخیص متون در رنگ‌ها، جهات، قلم‌ها و اندازه‌های مختلف است [۱۴، ۱۵، ۱۶، ۱۷ و ...]. کار انجام شده توسط چیانگ و نوبلاک [۵] نیز روی استخراج و تشخیص متن روی نقشه‌های تاریخی لاتین متمرکز شده است و علاوه بر این روشی خودکار نیست. لذا امکان استفاده از نقشه‌های مورد استفاده توسط آن‌ها به عنوان داده ورودی سیستم وجود ندارد؛ اما درصد دقت و فراخوانی استخراج متون در کار آن‌ها و روی منابع دارای متون لاتین مشترک با کار ما به ترتیب ۷۸/۵٪ و ۷۷/۶٪ بود که در کار ما به ۸۷/۷۵٪ و ۹۶/۱۸٪ رسیده است. جدول (۲-۴) درصد استخراج متون از نقشه‌های منابع مختلف مورد استفاده را به تفکیک منبع نشان می‌دهد.

جدول ۲-۴: درصد استخراج متون از نقشه‌های منابع مختلف

منبع نقشه	درصد دقت	درصد فراخوانی
کتاب اول	۸۸/۶۴	۹۳/۴۸
Bing	۹۳/۸۵	۷۸/۳۸
سایت شهرداری تهران	۸۸/۴۶	۸۸/۷۳
گوگل	۹۳/۳۴	۸۹/۷۷
Tehranmaps	۸۲/۱۲	۷۳/۱۹
متفرقه	۹۵/۹۳	۹۶/۰۷

فصل پنجم

نتیجہ گیری و پیشہ و شہادہ

۵-۱ مروری بر فصل‌های پایان‌نامه

در این پایان‌نامه، استخراج متن از نقشه جغرافیایی با استفاده از کوانتیزاسیون فضای رنگ مطالعه و بررسی گردید و کلمات لایه متنی برای ورود به OCR آماده شدند.

در فصل دوم به بررسی سابقه موضوع استخراج خطوط خیابان، استخراج تقاطع خیابان، استخراج ناحیه مورد نظر و استخراج متن پرداخته شد. استخراج متون لاتین و فارسی هریک به طور جداگانه مورد بررسی قرار گرفت و نقاط ضعف و قوت هر روش بیان شد. همچنین اهمیت موضوع مورد بررسی با توجه به کارهای قبلی انجام گرفته در این زمینه، آشکار شده است.

در فصل سوم، الگوریتم‌های به کار رفته در روش پیشنهادی به طور جداگانه مورد بررسی قرار گرفته‌اند. تاریخچه، الگوریتم و روش عملکرد آن‌ها در یک شبکه عصبی بیان شده است. همچنین در این فصل به تعریف کوانتیزاسیون رنگ پرداختیم. متوجه شدیم که الگوریتم جابه‌جایی میانگین تغییرات فضای رنگ را کاهش می‌دهد و الگوریتم نگاشت خود سامان‌ده فضای رنگ را کوانتیزه می‌کند.

در فصل چهارم، الگوریتم‌های ارائه‌شده برای استخراج متون از نقشه‌های شهری شرح داده شده است و نتایج شبیه‌سازی‌ها روی سه نوع نقشه مختلف نشان داده شده است. در نقشه اول نشان داده شده، متون فارسی در جهات مختلفی وجود دارند و تک‌تک نقشه‌های حاصل شده از اعمال الگوریتم روی آن نشان داده شده‌اند. در نقشه دوم فارسی متون در رنگ‌ها، اندازه‌ها، جهات و قلم‌های مختلفی وجود دارند و کارآیی روش پیشنهادی روی آن به طور کامل مشخص است. با اجرای الگوریتم روی نقشه سوم این ادعا را که روش پیشنهادی در مورد نقشه‌های لاتین نیز نتایج خوبی به دست می‌دهد، به اثبات رسانده‌ایم.

۵-۲ نتیجه‌گیری

این مقاله الگوریتمی برای استخراج متون فارسی و لاتین از نقشه‌های شهری ارائه می‌کند. روش ارائه شده مستقل از قلم، اندازه، رنگ و جهت متون است. ما از یک الگوریتم جابه‌جایی میانگین با پارامترهای تنظیم شده مناسب استفاده می‌کنیم. پارامترهای روش جابه‌جایی میانگین با تکرار فرآیند پیاده‌سازی الگوریتم و مشاهده نتایج آن به گونه‌ای تنظیم شده‌اند که روی تعداد زیادی از نقشه‌ها نتیجه مطلوبی بدست می‌دهند. سپس روش نگاشت خود سامان‌ده را اعمال می‌کنیم تا ابعاد فضای رنگ کاهش یابد. از آن جایی که یکی از سه لایه یک، پنج یا هفت حاوی بیشترین اطلاعات متنی است، در هر یک از این سه لایه تعداد عناصر متصلی که در آستانه مشخصی قرار دارند، توسط سیستم شمارش می‌شوند. از بین این سه لایه، لایه‌ای را که حاوی بیشترین تعداد عناصر متصل است، به عنوان لایه برتر در نظر می‌گیریم و آن را ابتدا به یک تصویر سطح خاکستری و سپس به یک تصویر دودویی تبدیل می‌کنیم. به این صورت که مقادیر صفر به عنوان پس‌زمینه و سایر مقادیر به عنوان پیش‌زمینه در نظر گرفته می‌شوند. جهت جداسازی کلمات از لایه متنی، با استفاده از یک المان ساختاری تصویر را گسترش داده می‌شود، با استفاده از تحلیل اجزای متصل و "AND" کردن تصویر هر جزء متصل با تصویر اصلی مرتبط با آن، عناصر متصل آن را می‌یابیم تا کلمات به تفکیک استخراج شوند. در گام بعدی جعبه محیطی اطراف هر کلمه را یافته، پروفایل تصویر در راستای سطرها در جعبه محیطی هر کلمه در زوایای مختلف بین 90° تا $+90^\circ$ محاسبه می‌شود. زاویه‌ای را که در آن تعداد مقادیر غیرصفر پروفایل تصویر کلمه کمترین مقدار خود را داشته باشد، به عنوان زاویه چرخش مطلوب در نظر گرفته می‌شود. سپس هر کلمه به اندازه زاویه چرخش مطلوب خودش می‌چرخد تا در راستای افقی قرار بگیرد. نتایج تجربی دقت 90.4% و فراخوانی 86.6% را برای رهیافت پیشنهادی در استخراج متون نشان می‌دهد. خطاهای موجود در هنگام چرخیدن کلمات به صورت افقی هم ناشی از به هم چسبیدن بعضی کلمات مجزا به علت فشردگی اطلاعات روی نقشه‌ها است.

۵-۳ کارهای آتی

در این زیربخش، به دنبال مسائلی که در این پایان‌نامه مورد مطالعه قرار گرفت، چند پیشنهاد جهت ادامه مطالعه و تحقیق ارائه می‌شود:

- زمان لازم برای انجام پردازش توسط این روش بسته به ابعاد نقشه و فشردگی اطلاعات روی آن توسط یک نرم‌افزار Matlab 2014 A روی یک کامپیوتر ۲ هسته‌ای با سیستم عامل Windows 7 بین ۸ تا ۳۵ دقیقه می‌باشد. لذا اعمال روشی که بتواند به سیستم کمک کند تا زمان کمتری را برای انجام پردازش صرف کند، یکی از اهداف محققان در این زمینه است.
- مطالعه و بررسی روشی که بتواند به بهبود روش کنونی کمک کند و درصد دقت و فراخوانی آن را بالاتر ببرد.
- ورود کلمات به نرم‌افزارهای OCR فارسی و لاتین و محاسبه درصد تشخیص آنها

مراج

- [1] Tabassum M. and Shorif Uddin M. (2011) "Extraction of ROI in Geographical Map Image," **Journal of Emerging Trends in Computing and Information Sciences**, Vol 2, No. 5, pp.237-242.
- [2] Myers G. K., Mulgaonkar P. G., Chen Ch.H., Decurtins J.L. and Chen E., (1996), "Verification-Based Approach for automated Text and Feature Extraction from Raster-Scanned Maps", Vol. 1072, pp. 190-203, **Graphic Recognition Methods and Applications**, Kasturi R. and Tombre K., Springer Berlin Heidelberg.
- [3] Li L., Nagy G., Samal A., Seth S.C. and Xu Y. (2000) "Integrated text and line-art extraction from a topographic map", **International Journal on Document Analysis and Recognition(IJDAR)**, Vol 2, Issue 4, pp 177-185.
- [4] Chiang Y. Y. and Knoblock C.A., (2010) "An approach for recognizing text labels in raster maps,"20th International Conference on Pattern Recognition (ICPR), pp. 3199-3202.
- [5] Chiang Y-Y. and Knoblock C.A., (2014) " Recognizing text in raster maps", **GeoInfomatica**, Vol. 19, Issue 1, pp. 1-27.
- [6] Roy P. P., Pal U., Llado's J. and Delalandre M., (2009) "Multi-Oriented and Multi-sized Touching Character Segmentation using Dynamic Programming", 10th International Conference on Document Analysis and Recognition (ICDAR), pp.11-15.
- [7] Cao R. and Tan Ch. L., (2002) "Text/Graphics Separation in Maps", Vol 2390, pp. 167-177, **Graphics Recognition Algorithm and Applications**, Blostein D. and Kwon Y.B., Springer Berlin Heidelberg.
- [8] <http://www.avval.ir/tehran-map/>, (Retrieval at 2017/02/01).
- [9] Deseilligny M. P., Men H. and Stamon G., (1995), Character string recognition on maps, a rotation-invariant recognition method, "**ELSEVIER, Pattern Recognition Letter** ", Vol. 16, Issue 12, pp. 1297-1310, .
- [10] <http://www.streetdirectory.com/malaysia/johor/>, (Retrieval at 2017/02/01).
- [11] <https://www.google.com/maps/>, (Retrieval at 2017/02/01).
- [12] Chiang Y-Y. and Knoblock C. A., (2011) "A general approach for extracting road vector data from raster maps," **International Journal on Document Analysis and Recognition(IJDAR)**, Vol. 12. pp. 55-81.
- [13] Chiang Y-Y., Knoblock C.A. and Chen C-C., (2005) "Automatic Extraction of Road Intersections from Raster Maps", GIS Proceedings of the 13th annual ACM international workshop on Geographic information systems, pp. 267-276, Germany.
- [14] Goto H. and Aso H., (1999) "Extracting curved lines using local linearity of the

text line", **International Journal of Document Analysis and Recognition (IJ DAR)**, vol.2, pp. 111-118.

- [15] Fletcher L.A. and Kasturi R., (1988) "A robust algorithm for text string separation from mixed text/graphics images", **IEEE Transactions Pattern Analysis and Machine Intelligence**, Vol 10, Issue 6, PP. 910-918.
- [16] Chen L-H. and Wang J-Y., (1997) "A system for extracting and recognizing numeral strings on maps", Proceedings of the 4th International Conference on Document Analysis and Recognition (ICDAR), Vol. 1, pp. 337-341.
- [17] Hones F. and Litcher J., (1994) "Layout Extraction of Mixed-Mode documents", **Machine Vision and Applications**, Vol. 7, Issue. 4, pp. 237-246.
- [18] Pal U., Sinha S. and Chadhuri B. B., (2003) "Multi-oriented English Text Line Identification," 13th Scandinavian Conference (SCIA), pp. 1146-1153.
- [19] Tan C.L. and Ng P.O., (1998) "Text extraction using pyramid", **Pattern recognition, Elsevier**, Vol. 31, No. 1, pp. 63-72.
- [20] Velazquez A. and Levachkine S. (2004), "Text/Graphic Separation and Recognition in Raster- scanned Color Cartographic Maps", Vol. 3088, pp. 63-74, **Graphic Recognition, Recent Advances and Perspectives**, Lladós J. and Kwan Y.B., Springer Berlin Heidelberg.
- [21] Roy P.P, Pal U, Lladós J and Kimura F., (2008) "Multi-Oriented English Text Line Extraction using Background and Foreground Information", The eighth IAPR international workshop on document analysis systems, IEEE, pp. 315-322.
- [22] Hunag Ch- L. and Lin Y-W., (1999) "Automatic Taiwanese Municipal Color Map Reading Systems", 12th Conference on Vision Interface, pp. 249-255 Trois-Rivieres, Canada.
- [23] Pouderoux J., Gonzato J. C., Pereira A. and Guitton P., (2007) "Toponym recognition in scanned color topographic maps", 9th International Conference on Document Analysis and Recognition (ICDAR), Vol. 1, pp. 531-535.

[۲۴] کبیر ا.، غفاری برانقار ع. و کنگرلو ک.، (۱۳۹۰) "جداسازی متن فارسی از تصویر اسکن شده نقشه‌های شهری"، هفتمین کنفرانس ماشین بینایی و پردازش تصویر ایران، تهران، دانشگاه علم و صنعت.

[۲۵] کبیر ا.، غفاری برانقار ع. و کنگرلو ک.، (۱۳۹۱) "یک روش بازنمایی بر پایه تبدیل فاصله برای آشکارسازی متن در تصاویر نقشه‌های شهری"، بیستمین کنفرانس مهندسی برق ایران.

[26] Pujol A. and Chen L. (2007), "Color quantization for image processing using self

information", 6th International conference on information, communication and signal, IEEE.

- [27] Meskaldji Kh., Boucherkha S. and Chikhi S. (2009), "color quantization and its impact on color histogram based image retrieval", First international conference on network digital technologies (NDT), IEEE.
- [28] Orchard M.T. and Bouman C.A., (1991), "color quantization of images", **IEEE Transaction on Signal Processing**, Vol.39, Issue 12, pp. 2677-2690.
- [29] <http://www.cs.montana.edu/~halla/cs525/intro.html>, (Retrieval at 2017/02/01).
- [۳۰] کثیری بیدهندی س. و صفا بخش ر.، (۱۳۸۵) "کوانتیزاسیون رنگ مبتنی بر شبکه نگاشت خود سامان ده تطبیقی با زمان پروانه‌ای"، چهارمین کنفرانس ماشین بینایی و پردازش تصویر ایران، دانشگاه فردوسی مشهد.
- [31] Celebi, M.E., (2009) "An effective color quantization method based on the competitive learning paradigm", Proceeding of the 2009 international conference on image processing, computer vision and pattern recognition, 2 volumes, Las Vegas.
- [32] Atsalakis A., Papamarkos N., Kroupis N., Soudris D. and Thnailakis A., (2004), "Colour quantization technique based on image decomposition and its embeded system implementation", **IEEE proceedings vision image and signal processing**, Vol. 151, No. 6, pp. 511-524.
- [33] https://github.com/delimitry/octree_color_quantizer, (Retrieval at 2017/02/01).
- [34] Wen Zh.Q. and Cai Z.X., (2006), "Mean-shift algorithm and its application in tracking of objects", Proceedings of the fifth international conference on machine learning and cybernetics, IEEE, pp. 4024-4028, Dalian.
- [35] Fukunaga K. and Hostetler L., (1975) "The estimation of the gradient of a density function, with applications in pattern recognition", **IEEE Transactions on Information Theory**, Vol. 21, Issue 1, pp. 32-40.
- [36] Cheng Y., (1995) "Mean Shift, mode seeking and clustering" **IEEE Transactions on Pattern Analysis and Machine Intelligence**, Vol. 17, Issue 8, pp. 790-799.
- [37] Tao W., Jin H. and Zhang Y., (2007), "Color image segmentation based on mean shift and normalized cut", **IEEE Transactions on systems, man, and cybernetics, Part B, Cybernetics**, Vol. 37, No. 5, pp 1382-1389.
- [38] Aliyari Ghassabeh Y., (2015) "A sufficient condition for the convergence of the mean shift algorithm with Gaussian Kernel", **Journal of Multivariate Analysis**, Vol. 135, pp. 1-10.

- [39] Aliyari Ghassabeh Y., (2013), On the convergence of the mean shift algorithm in the one dimensional space, "**Pattern Recognition Letters, ELSEVIER**", Vol. 34, Issue 12, pp. 1423- 1427.
- [40] <http://upper64k.com/blog/articles/2/implementing-the-mean-shift-algorithm-in-octave>, (Retrieval at 2017/02/01).
- [41] Comaniciu D. and Meer P., (2009) "Mean shift: a robust approach toward feature space analysis", **IEEE Transactions on pattern analysis**, Vol.24, pp: 603-619.
- [42] http://pami.xmu.edu.cn/~wlzhao/mean_shift.htm, (Retrieval at 2017/02/01).
- [43] Kohonen, T. and Honkela, T (2007). "Kohonen Network". Scholarpedia, (Retrieval at 2017/02/01).
- [44] Malsburg V. D. , (1973), "Self- organization of orientation sensitive cells in the striate cortex", *C. Kybernetik* , Vol. 14, Issue 2, pp. 85-100.
- [45] Turing A. M., (1952) "The chemical basis of morphogenesis", *Philosophical Transaction of The Royal Society*, Vol. 237, pp. 5-72.
- [46] Fritzke B, (1994) "Growing cell structures- a self organizing network for unsupervised and supervised learning" , *Neural networks*, Vol. 7, No. 9, pp. 1441-1460.
- [47] ایمانی ز.، (۱۳۹۲)، پایان نامه کارشناسی ارشد، "تشخیص برون خطی کلمه دست نوشته فارسی با استفاده از مدل مخفی مارکوف" ، دانشگاه صنعتی شاهرود، شاهرود.
- [48] Fong Ch. K., (2007) "A study in deploying self organized map (SOM) in an open source JZEE cluster and caching system", *International Conference on Complex medical engineering, IEEE*, pp. 778-781.
- [49] <http://ppt-online.org/32073-page-92/107>, (Retrieval at 2017/02/01)
- [50] Park H.J., Kinn K.B. and Cha E.Y, (2012) "An Effective Color Quantization Method Using Octree-Based Self Organizing Maps," **Computational intelligence and Neuroscience**, Vol. 34, pp 2259-2281.
- [51] <http://region17.tehran.ir/Portals/0/Image/NewWebsite/map/golchinmap.jpg>, (Retrieval at 2017/02/01)
- [52] <https://www.bing.com/maps/>, (Retrieval at 2017/02/01).
- [53] <http://map.tehran.ir/>, (Retrieval at 2017/02/01).

واژه‌نامه انگلیسی به فارسی

A

Analysis	تحلیل
Anchor	لنگر
Angle	زاویه
Approach	رهیافت
Area/ ratio filter	فیلتر مساحت/ نسبت
Artificial neural network	شبکه عصبی مصنوعی
Aspect ratio	نسبت دید
Assumption	فرض
Automatic	خودکار
Autopilot Automobiles	خودروهای بدون سرنشین

B

Background	پس‌زمینه
Bandwidth	پهنای باند
Baseline	خط کرسی
Blind clustering	خوشه‌بندی کور
Blob	حباب
Block	بلوک
Boundingbox	جعبه محیطی
Bottom- up	از پایین به بالا

C

Candidate	نماینده
Center- cut	برش مرکز
Centerline property	خاصیت خط مرکزی
Clustering	طبقه‌بندی
Color quantization	کوانتیزاسیون رنگ
Column	سطر
Combining	ترکیب کردن
Competition	رقابت
Completely connected	کاملاً متصل
Compressing	فشرده‌سازی
Connected components	اجزای متصل
Content	محتوا
Convolution	کانولوشن
Convergence	همگرایی
Crossing point	نقطه تقاطع
Curvature	خمیدگی
Curved	خمیده
D	
Data Mining	کاوش داده‌ها
Dilation	گسترش
Direction	جهت
Discrete	گسسته
Divide and conquer extraction	استخراج تقسیم و تصرف

Dominant غالب

Dynamic پویا

E

Euclidean اقلیدوسی

Expandability گسترش

Extrapolate برونپایی

F

Filtering فیلتر کردن

Fingerprint اثر انگشت

Flexible انعطاف پذیر

Foreground پیش‌زمینه

Format قالب

Fuzzy فازی

G

Geographical Information Systems سیستم‌های اطلاعات جغرافیایی

Global کلی

Gradient گرادیان

Grouping گروه‌بندی کردن

H

Height ارتفاع

Hexagonal شش ضلعی

Hidden layer لایه مخفی

Hue فام

I

Information اطلاعات

Integrated یکپارچه

Intensity روشنایی

Image processing پردازش تصویر

Independent مستقل

J

K

Kernel کرنل

K-Means میانگین K

L

Labeling برچسب‌زنی

Length طول

Learning rate نرخ یادگیری

Line- pairing criteria معیار عرض خط

M

Machine learning یادگیری ماشین

Mean- shift algorithm الگوریتم جابه‌جایی میانگین

Median- cut برش میانه

MinMax کمترین حداقل

Mix- processing algorithm الگوریتم پردازش مخلوط

Modeling مدلسازی

Morphological operators عملگرهای مورفولوژیکی

Multi oriented چند جهته

N

Nearest neighbor clustering method روش خوشه‌بندی نزدیک‌ترین همسایه

Neighboring property خاصیت همسایگی

Neuron نرون

Node گره

Non- parametric غیر پارامتریک

Non- zero غیر صفر

O

Optical character recognition تشخیص نویسه نوری

Optimize بهینه

P

Parallel موازی

Pattern الگو

Post- clustering پس طبقه‌بندی

Precision دقت

Pre- clustering پیش طبقه‌بندی

Pre- processing پیش پردازش

Projection profile پروفایل افکنش

Pyramid اهرام

Q

R

Recall	فراخوانی
Recursively	بازگشتی
Region of interest	ناحیه مورد نظر
Resolution	وضوح
Retrieval	بازیابی
Road intersections	تقاطع خیابان

S

Salient points	نقاط برجسته
Saturation	اشباع
Scale	مقیاس
Self organizing map	نگاشت خود سامانده
Single oriented	تک جهته
Sharpening	تیز کردن
Smoothing	هموار کردن
Splitting	لغزیدن
Square	مربعی
String	رشته
Structure element	المان ساختاری
Sublayer	زیر لایه
Supervised	بانظارت
Symbol	نماد

T

Template based matching	تطبیق بر پایه الگو
-------------------------------	--------------------

Text labels برچسب‌های متنی

Threshold آستانه

Top- down approach رهیافت بالا به پایین

Topographic مکان‌نگاری

Touching همپوشان

Transform تبدیل

Triple سه‌گانه

U

Unique منحصر به فرد

Unsupervised بدون نظارت

V

Vector data داده‌های برداری

W

Water reservoir concept مفهوم مخزن آب

Width عرض

World wide web شبکه وسیع جهانی

X

Y

Z

Abstract

Text layers in maps provide valuable information by relating names to their positions on the maps. Most previous researches in this field are focused on Latin texts and the results of presented methods on Persian texts are poor. In this thesis we present a new method for automatic extraction of texts from raster maps using self-organizing maps for quantization.

In the proposed approach a Mean-Shift algorithm is used to reduce changes in the color space. Consequently, we quantize the maps using SOM and reduce the dimensions of color space with it. Based on the results obtained through applying the program to different maps, either the first, the fifth or the seventh layer resulted from SOM is the best textual layer. In the next step, the numbers of connected components which are in a predefined threshold, in each of the three aforementioned layers are counted. The Text layer is the layer with the most connected components. In this layer each of the components has an intensity value in gray level which depends to their main color in the original map, but the black background has no textual information. Therefore, we consider black pixels as background pixels and the other pixels as foreground. By using this approach, the text layer is extracted.

To prepare words for OCR, we expand the binarized image using a structure element and "dilation" morphological operator. At the next step, we find connected components in the dilated image and "AND" each dilated component to its related component in the original binarized image. Using boundingbox property, we find each boundingbox profile in each angle between -90 and +90. The angle in which the number of non-zero values of the word's profile has its minimum value is our desired rotation angle. Therefore, we rotate each word by its desired rotation value to be placed in horizontal position. This method is independent of font, size, direction and color of the texts and can find both Persian and Latin texts in maps. Experimental results show %90.4 precision and %86.6 recall in text extraction from raster maps in our proposed approach.

Keywords- Color space quantization, Mean- Shift algorithm, Preparation for OCR, Text extraction, Self- organizing maps.



Shahrood University of Technology
Faculty of Electrical & Robotics Engineering

M.Sc. Thesis in Communication Systems Engineering

Text Extraction from Raster Maps Using Color Space Quantization

By: **Sanaz Hadipour Abkenar**

Supervisor:

Dr. Alireza Ahmadyfard

February 2017