

استفاده از الگوریتم جستجوی گرانشی بهبود یافته در خوشه‌یابی خودکار تصاویر

علیرضا سردار^۱، سید حمید ظهیری^۲

^۱کارشناس ارشد الکترونیک از دانشگاه بیرجند، alre_sar@yahoo.com

^۲دانشیار گروه الکترونیک دانشگاه بیرجند، hzahiri@birjand.ac.ir

چکیده:

از نظر عملی مسئله خوشه‌یابی به عنوان یکی از مهم‌ترین مباحث در بازشناسی الگو است و به علت کاربرد فراوان آن در زمینه‌های داده‌کاوی، پردازش تصویر، ماشین بینایی و... توجه بسیاری از محققان را به خود جلب کرده است. هرچند تاکنون روش‌های ابتکاری مختلفی به منظور خوشه‌یابی به کار گرفته شده است، ولی در اکثر این روش‌ها، نیاز است که تعداد خوشه‌ها از قبل مشخص گردد (خوشه‌یابی نظارت شده) این مسئله یکی از مهم‌ترین معایب این روش‌ها محسوب می‌شود. زیرا در بعضی مسایل خوشه‌یابی، اطلاعاتی از تعداد خوشه‌ها در دسترس نیست. البته الگوریتم‌های مختلفی برای رفع این مشکل ارائه شده است که از آن جمله می‌توان به *MEPSO* و *GCUK*، *VLIGA*، *DCPSO* که مبتنی بر روش‌های هوش جمعی می‌باشد، اشاره کرد. در این مقاله سعی شده است که یک روش خوشه‌یابی نظارت نشده به منظور خوشه‌یابی تصاویر با استفاده از الگوریتم بهبود یافته جستجوی گرانشی ارائه شود. در ابتدا الگوریتم جستجوی گرانشی که یک الگوریتم جدید در زمینه هوش جمعی است معرفی شده و پس از بهبود عملکرد آن، (با الهام از پدیده سیاه‌چاله‌ها در طبیعت) الگوریتم خوشه‌یابی مورد نظر ارائه می‌گردد. در پایان پس از بررسی عملکرد الگوریتم پیشنهادی بر روی تعدادی مجموعه داده آزمایشی، در دو کاربرد کوانتیزه‌سازی و ناحیه‌بندی تصویر مورد بررسی قرار می‌گیرد.

کلید واژه‌ها: خوشه‌یابی نظارت نشده تصاویر، هوش جمعی، الگوریتم جستجوگر گرانشی بهبود یافته، کوانتیزه‌سازی تصویر، ناحیه‌بندی تصویر

۱. مقدمه

به دو دسته خوشه‌یابی نظارت شده و خوشه‌یابی نظارت نشده تقسیم می‌شوند، در خوشه‌یابی نظارت شده، تعداد خوشه‌ها از قبل توسط کاربر تعیین می‌گردد در حالی که در خوشه‌یابی نظارت نشده، تعداد بهینه خوشه‌ها توسط خود الگوریتم به دست می‌آید. تاکنون الگوریتم‌های خوشه‌یابی مختلفی ارائه شده است [3]، [4] که از مهم‌ترین آنها می‌توان به الگوریتم K-means [5] اشاره کرد. با توجه به اینکه در الگوریتم K-means، نیاز است که یک تعداد مراکز اولیه خوشه مشخص گردد و این که این الگوریتم با معیار میانگین فاصله درون کلاسی عمل می‌کند، در بسیاری از موارد (به خصوص زمانی که خوشه‌ها از

خوشه‌یابی داده‌ها یک امر مهم در مبحث داده‌کاوی است. هدف از خوشه‌یابی این است که از بین یک سری داده‌های ورودی، آنهایی که از نظر یک تعداد ویژگی خاص، به هم شبیه‌تر هستند در یک خوشه قرار بگیرند. به عبارت دیگر داده‌های قرار گرفته در یک خوشه بیشترین شباهت را به یکدیگر دارند و داده‌های موجود در خوشه‌های متفاوت از یکدیگر متمایز هستند. الگوریتم‌های خوشه‌یابی از نظر تعیین تعداد خوشه‌ها

استفاده از روش معرفی شده در بخش ۳ توضیح داده خواهد شد. کوانتیزه‌سازی و ناحیه‌بندی تصاویر به عنوان سایر کاربردهای روش پیشنهادی در پردازش تصویر به ترتیب در بخش‌های ۵ و ۶ ارائه خواهند گردید. بخش ۷ در بردارنده نتایج کمی به دست آمده و همچنین مقایسه عملکرد روش پیشنهاد شده با سایر روش‌های مشابه می‌باشد. نهایتاً بخش ۸ با بحث پیرامون نتایج به دست آمده مقاله را به پایان می‌رساند.

۲. الگوریتم جستجوی گرانشی (GSA)

الگوریتم جستجوی گرانشی (GSA) [17] در سال ۲۰۰۹ معرفی شد. در این الگوریتم، ابتدا تعداد N جرم در فضای پاسخ مسئله که می‌تواند D بعد داشته باشد به صورت تصادفی ایجاد گردیده و سپس با توجه به موقعیت و میزان برازندگی جرم‌ها، که توسط یک تابع برازندگی به صورت معادله (۱) مشخص می‌شود، مقداری جرم به هر یک از آنها تعلق می‌گیرد.

$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)}$$

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^N m_j(t)}$$

$$worst(t) = \max_{j \in \{1 \dots N\}} fit_j(t)$$

$$best(t) = \min_{j \in \{1 \dots N\}} fit_j(t) \quad (1)$$

با الهام از قانون گرانش، جرم‌ها به یکدیگر نیرو وارد کرده و در اثر این نیرو، هر جرم مقداری شتاب در جهت مشخصی به خود می‌گیرد که بردار شتاب حاصله، توسط معادله (۲) محاسبه می‌گردد.

$$a_i^d(t) = G(t) \sum_{\substack{j=1 \\ j \neq i}}^N \left[rand_j \frac{M_j(t)}{R_{ij}(t) + \epsilon} (x_j^d(t) - x_i^d(t)) \right]$$

$$G(t) = G_0 \times e^{-\alpha \frac{t}{T}} \quad (2)$$

در رابطه فوق، R فاصله بین ذره i و j در بعد d ام و در تکرار t ام الگوریتم است و M نیز، میزان جرم ذره i ام را نشان می‌دهد که بر ذره i ام نیرو وارد کرده و باعث اعمال شتاب به آن می‌شود. تابع سیگما به منظور جمع شتاب‌های وارده از تمام ذرات بر

نظر اندازه و تعداد همگن نیستند) ممکن است در یک بهینه محلی گرفتار شود. از آنجاکه الگوریتم‌های هوش جمعی مانند PSO [10] و GA [9] و ACO [11]، فضای پاسخ را به طور موثرتری جستجو می‌کنند، به منظور رفع این مشکل روش‌های جدید خوشه‌یابی با استفاده از الگوریتم‌های هوش جمعی ارائه شده است [12][13][1]. [27].

یکی دیگر از معایب روش‌های خوشه‌یابی ابتکاری مانند الگوریتم K-means، نیاز به آگاهی از تعداد خوشه‌ها قبل از به کارگیری آن می‌باشد و به عبارت دیگر فقط به منظور خوشه‌یابی نظارت شده می‌تواند مورد استفاده قرار گیرد. هر چند جهت رفع این مشکل روش‌های ابتکاری مختلفی ارائه شده است [6][7]، اما در بسیاری از موارد پاسخ‌های به دست آمده، پاسخ‌های کاملاً بهینه‌ای نیست. با توجه به عملکرد مناسب الگوریتم‌های هوش جمعی، به منظور استفاده از این الگوریتم‌ها جهت ارائه یک الگوریتم خوشه‌یابی نظارت نشده، تحقیقات گسترده‌ای صورت گرفته که از آن جمله می‌توان به الگوریتم‌های GA [8] و GCUK [14] (با استفاده از GA)، DCPSO [2] و MEPSO [15] (با استفاده از PSO) و IAFHA [16] (با استفاده از ACO) و سایر روش‌ها [28][29] اشاره کرد.

یکی از جدیدترین الگوریتم‌ها در زمینه هوش جمعی الگوریتم جستجوی گرانشی [17] (GSA) است. در این مقاله ابتدا این الگوریتم معرفی شده و سپس با اشاره به مشکلات آن، با الهام از پدیده سیاهچاله‌ها در طبیعت، این الگوریتم بهبود داده شده و با نام الگوریتم جستجوی گرانشی سیاره‌ها و سیاهچاله‌ها (PABH-GSA: Planets And Black Holes GSA) ارائه می‌گردد. در ادامه، با معرفی یک تابع برازندگی مناسب، برای الگوریتم پیشنهادی، یک الگوریتم خوشه‌یابی نظارت نشده معرفی می‌گردد که پس از بررسی عملکرد آن بر روی تعدادی مجموعه داده مصنوعی، جهت خوشه‌یابی خودکار تصاویر سطح خاکستری و ناحیه بندی تصاویر رنگی به کمک خوشه‌یابی اطلاعات رنگ، مورد استفاده قرار می‌گیرد.

نحوه سازمان‌دهی این مقاله به این صورت است که در بخش ۲ به معرفی اجمالی الگوریتم جستجوی گرانشی پرداخته خواهد شد. سپس در بخش ۳ روش بهبود یافته آن در این تحقیق، یعنی الگوریتم جستجوی گرانشی سیاهچاله‌ها بیان می‌شود. در بخش ۴، الگوریتم پیشنهادی برای خوشه‌یابی نظارت نشده با

کنند و یا این که شتاب اعمالی به آنها از طرف جرم‌های کوچکی که در فاصله کم‌تری از آنها قرار دارند به نسبت بیشتر باشد و به سمت آنها(نقاط دارای برازندگی کم‌تر) حرکت کنند. از طرفی به جرم‌های بزرگ‌تری که در فاصله نزدیک‌تری نسبت به هم قرار دارند، شتاب بسیار قویتری اعمال می‌شود و به تبع آن سرعت و جابجایی بسیار زیادی پیدا می‌کنند که با توجه به گسسته بودن الگوریتم، در تکرار بعدی، نه تنها از محدوده بهینه خارج شده، بلکه ممکن است در فاصله بسیار دورتری نسبت به آن قرار بگیرند. این عوامل سبب می‌شود که این الگوریتم همگرایی نداشته باشد. بنابراین بهترین گزینه این است که شتاب گرانشی به فاصله بین جرم‌ها وابسته نباشد. علاوه بر این، عدم وابستگی شتاب جرم‌ها به فاصله، باعث می‌شود که در مراحل اولیه اجرای الگوریتم، تمام ذرات به دلیل بزرگ بودن $G(t)$ ، دارای شتاب و سرعت احتمالی زیادی باشند که جستجوی کلی فضای پاسخ را در مراحل اولیه اجرای الگوریتم به دنبال دارد، به مرور، با افزایش تعداد تکرار الگوریتم، سرعت و شتاب احتمالی آنها، به دلیل کوچک شدن بیش از حد $G(t)$ کاهش می‌یابد و این باعث می‌شود که محدوده‌های بهینه‌ای که در مراحل اولیه به دست آمده است با توجه به کاهش سرعت جرم‌ها با دقت بیشتری جستجو شود و به عبارت دیگر باعث جستجوی محلی در فضای پاسخ می‌شود.

هر چند مقدار کلی شتاب هر جرم به فاصله آن تا دیگر اجرام بستگی ندارد، اما مقدار شتاب در هر بعد، از حاصلضرب مقدار کلی شتاب در یک ضریب به دست می‌آید که مقدار این ضریب برابر است با نسبت فاصله در آن بعد به فاصله در تمام ابعاد، بنابراین در هر بعد با کمتر شدن فاصله، شتاب اعمالی در آن بعد نیز کمتر می‌شود، این روند ممکن است همگرایی الگوریتم به یک بهینه محلی را به دنبال داشته باشد. برای رفع این مشکل، در این تحقیق، مقدار شتاب در هر بعد نیز مستقل از فاصله در نظر گرفته شد و معادله شتاب به صورت معادله (۴) تغییر یافت. همچنین به منظور افزایش خصوصیت تصادفی جستجو، مقدار rand برای هر بعد نیز متفاوت در نظر گرفته شد.

$$a_i^d(t) = G(t) \sum_{j=1}^N \left[\text{rand}_j^d(t) M_j(t) \text{sign}_j^d(t) \right]$$

$$\text{sign}_j^d(t) = \begin{cases} \frac{x_j^d(t) - x_i^d(t)}{|x_j^d(t) - x_i^d(t)|} & \text{if } i \neq j \\ 0 & \text{if } i = j \end{cases} \quad (4)$$

ذره α م در هر بعد به کار رفته است و بردار جابجایی ضرب شده نیز جهت شتاب را مشخص می‌کند. همچنین $G(t)$ به عنوان ثابت گرانش در نظر گرفته شده، که در هر بار تکرار الگوریتم با توجه به معادله داده شده به روز می‌گردد. متغیر T نشان دهنده تعداد تکرار الگوریتم و G_0 و α توسط کاربر تعیین می‌گردد. سرعت و موقعیت هر ذره، با توجه به شتاب اعمالی به آن توسط معادلات زیر به روز می‌شود.

$$v_i^d(t+1) = \text{rand}_j \cdot v_i^d(t) + a_i^d(t)$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \quad (3)$$

با تغییر موقعیت هر ذره، مجدداً با توجه به معادله (۱) میزان برازندگی آنتی‌عین و روال قبلی تکرار می‌شود و این روند برای T بار تکرار الگوریتم اجرا می‌گردد، در نهایت بهترین ذره در تکرار T ام به عنوان پاسخ الگوریتم استخراج می‌شود.

۳. الگوریتم جستجوی گرانشی سیاره‌ها و سیاهچاله‌ها (PABH-GSA)

هرچند نتایج به دست آمده از GSA، به ازای یک سری توابع برازندگی [17]، نسبت به GA و PSO مناسب‌تر است، اما این الگوریتم دارای نقاط ضعفی است که در این قسمت علاوه بر شرح آن، سعی شده است با اعمال ایده‌های جدید، تا حدودی زیادی، این مشکلات برطرف گردد.

همان طور که می‌دانیم در طبیعت شتاب گرانشی وارده بر یک جسم با عکس مجذور فاصله آن تا جسم دیگر متناسب است، ولی در تحقیقات به عمل آمده در مورد GSA مشخص شد که استفاده از آن، واگرایی الگوریتم را به دنبال خواهد داشت. پس از بررسی‌های مختلف، به منظور دخیل بودن فاصله در تعیین شتاب بین دو ذره، مشخص شد که بهترین گزینه، استفاده از معکوس فاصله بین جرم‌ها است. با توجه به معادله (۲)، مشاهده می‌شود که نسبت قدرمطلق بردار فاصله در صورت (برای تمام ابعاد) به اندازه فاصله در مخرج کسر، برابر یک است. بنابراین با این توضیحات مشخص می‌شود که در GSA، مقدار شتاب وارده شده به هر جرم، به فاصله بین اجرام بستگی ندارد، دلیل این امر را می‌توان به این صورت تحلیل کرد که وجود هر توان مثبت از فاصله، در مخرج کسر، موجب می‌شود تا به جرم‌های کوچک (دارای برازندگی کم‌تر) که در فاصله‌های دورتری نسبت جرم‌های بزرگ (دارای برازندگی بیشتر) قرار گرفته‌اند، شتاب کم‌تری اعمال شود و در نتیجه دیرتر به سمت هدف حرکت

خوبی مورد بررسی قرار نگیرد و الگوریتم به یک بهینه محلی همگرا شود.

به وضوح مشخص است که مقدار β وابستگی زیادی به دامنه ابعاد مسئله دارد. یعنی برای مسائلی با فضای پاسخ بزرگ‌تر، به سیاه‌چاله‌های قوی‌تر و در نتیجه β های بزرگ‌تری نیاز است و بالعکس برای مسائلی با فضای پاسخ کوچک‌تر، مقدار β باید کوچک‌تر انتخاب شود.

ممکن است این اشکال مطرح شود که با ثابت نگه داشتن تعدادی از جرم‌ها و تبدیل آنها به سیاه‌چاله، الگوریتم به سمت یک بهینه محلی همگرا خواهد شد. اما از آنجا که تعداد ۱۰ جرم برتر، ثابت نگه داشته شده است که هرکدام می‌تواند در مکان متفاوتی از فضای مسئله قرار داشته باشد و از طرفی جرم آنها نیز به طور تصادفی با تابع rand تعیین می‌شود (معادله (۵)) و همچنین این که محل سیاه‌چاله‌ها در هر بار تکرار الگوریتم، به روز می‌شود (بهترین ۱۰ جرم در هر بار تکرار الگوریتم تبدیل به سیاه‌چاله می‌شود)، احتمال همگرا شدن به یک بهینه محلی بسیار کاهش می‌یابد.

با آمدن سیاه‌چاله‌ها، دیگر ضریب G_0 در معادله (۲) کارایی زیادی نخواهد داشت زیرا تعیین‌کننده اصلی در مقدار و جهت شتاب اجرام، سیاه‌چاله‌ها خواهند بود لذا برای خلاصه کردن معادله می‌توان $G_0=1$ در نظر گرفت، همچنین با بررسی مقدار α مشخص می‌شود که با تغییر مقدار T (تعداد تکرار الگوریتم) نیاز است که مقدار α نیز تغییر یابد و می‌توان این تغییرات را به صورت خطی در نظر گرفت.

$$\tau = \alpha / T \quad (6)$$

با انجام این تغییرات تابع $G(t)$ که به ۳ متغیر G_0 ، T و α وابسته بود. تنها به یک متغیر به نام τ وابسته می‌شود.

$$G(t) = \exp(-\tau \times t) \quad (7)$$

با اعمال $G(t)$ جدید، در بررسی‌های انجام شده مشخص شد که برای یک سیستم n بعدی با دامنه ابعاد $[-b, b]$ ، انتخاب $\beta =$ گزینه مناسبی است، همچنین برای عملکرد مناسب الگوریتم بهتر است مقدار τ در بازه $[0.5, 1, 0]$ انتخاب شود.

اعمال تغییرات مذکور و بررسی نتایج، بهبود قابل توجه در عملکرد الگوریتم را نشان می‌دهد. سیاه‌چاله‌ها باعث می‌شوند که تمام جرم‌ها در نهایت به هم پیوسته و یک جرم واحد را تشکیل دهند. از طرفی ثابت نگه داشتن سیاه‌چاله‌ها در هر بار تکرار الگوریتم، باعث می‌شود تا برازندگی بهترین جرم در هر بار

با تغییرات انجام شده، همان طور که از قبل پیش‌بینی می‌شد نتایج به دست آمده نسبت به حالت قبل بهبود خوبی داشت.

با بررسی مجدد الگوریتم مشخص می‌شود که نقاط بهینه (همان نقاط دارای جرم بیشتر) نیز در اثر نیروی وارده از دیگر اجرام تغییر مکان پیدا می‌کنند که این به صورت مانعی در برابر همگرایی این الگوریتم عمل می‌کند. برای رفع این مشکل، یک ایده بررسی شد و آن این که اگر تعدادی از نقاط دارای بیشترین جرم ثابت نگه داشته شود و بقیه جرم‌ها به حرکت خود ادامه دهند نتیجه الگوریتم چه خواهد شد؟

این کار با ثابت نگه داشتن ۱۰ نقطه که دارای بیشترین جرم هستند انجام شد. با اعمال این ایده، مشاهده شد که سرعت و دقت الگوریتم به طرز چشم‌گیری افزایش پیدا می‌کند، با این موفقیت و با الهام از پدیده سیاه‌چاله‌ها در طبیعت، به نظر رسید که اگر این بهترین نقاط را (۱۰ نقطه با جرم بیشتر) به عنوان سیاه‌چاله‌های ثابت و بقیه نقاط را به صورت سیاره‌های سرگردان در فضا در نظر بگیریم. پاسخ از این هم بهتر خواهد شد. به این صورت که، درست است سیاره‌ها به یکدیگر نیرو وارد می‌کنند ولی این نیرو در مقابل نیروی وارد شده توسط سیاه‌چاله ناچیز خواهد بود و باعث می‌شود که همه سیاره‌ها به صورت تصادفی به سمت سیاه‌چاله‌ها حرکت کنند. این کار را می‌توان به راحتی با چندین برابر کردن جرم بهترین ذرات به صورت معادله (۵) انجام داد.

$$M_j(t) = (\beta \times rand_j(t) + 1) \times M_j(t) \quad (5)$$

$$j \in \{\text{best 10 masses}\}$$

که در آن β یک عدد ثابت است و توسط کاربر تعیین می‌شود، عدد تصادفی rand برای جلوگیری از به دام افتادن الگوریتم در یک بهینه محلی، اعمال شده است، که توضیحات مربوط به آن در ادامه آمده است، عدد یک نیز به این دلیل به معادله اضافه شده که جرم جدید بوجود آمده، همیشه بزرگ‌تر مساوی جرم قبلی باشد.

در نگاه اول این طور به نظر می‌رسد که هرچه β بزرگ‌تر باشد پاسخ مناسب‌تری به دست خواهد آمد، ولی همان طور که قبلاً گفته شد با توجه به گسسته بودن الگوریتم، در صورت انتخاب β بزرگ، ذرات دارای شتاب، سرعت و جابجایی زیادی خواهند شد و این باعث می‌شود که جرم‌ها در شکاف زمانی ایجاد شده از نواحی بهینه عبور کرده و از آن دور شوند، از طرف دیگر سرعت بیش از حد جرم‌ها باعث می‌شود که فضای پاسخ به

مناسبی دارد. معادلات (۸) تا (۱۱) نحوه به دست آوردن DB را نشان می‌دهد.

$$S_{i,q} = \left(\frac{1}{|c_i|} \sum_{x \in c_i} \|x - m_i\|_2^q \right)^{1/q} \quad (8)$$

$$d_{ij,t} = \left(\sum_{s=1}^p |m_{si} - m_{sj}|^t \right)^{1/t} = \|m_i - m_j\|_t \quad (9)$$

$$R_{i,qt} = \max_{i \neq j} \left(\frac{S_{i,q} + S_{j,q}}{d_{ij,t}} \right) \quad (10)$$

$$DB = \frac{1}{n} \sum_{i=1}^n R_{i,qt} \quad (11)$$

در معادلات فوق، $S_{i,q}$ میزان پراکندگی داده‌های درون خوشه i را نسبت به مرکز خوشه m_i مشخص می‌کند، $|c_i|$ برابر تعداد داده‌های اختصاص یافته به خوشه i است و m_i از میانگین‌گیری این داده‌ها به دست می‌آید. $d_{ij,t}$ فاصله بین دو مرکز خوشه i و j را مشخص می‌کند، p برابر ابعاد داده n برابر تعداد خوشه‌ها است، همچنین، q یک عدد صحیح است که $q, t \geq 1$ و این دو می‌تواند مستقل از یکدیگر تعریف شود. در اینجا $q=t=2$ انتخاب شده است.

از آنجایی که در خوشه‌یابی نظارت نشده، همزمان دو هدف تعداد مناسب خوشه‌ها و تعیین مراکز بهینه را دنبال می‌کنیم، برای عملکرد بهتر الگوریتم و کاهش فضای جستجو، مراکز خوشه‌های پیشنهادی هر ذره، با میانگین‌گیری از داده‌های اختصاص یافته به هر خوشه به‌روز می‌شود که یک موقعیت بسیار مناسب‌تر نسبت به موقعیت قبلی می‌باشد. این تغییر منجر به نتایج بهتر و سریعتر می‌شود و آزمایشات انجام شده نیز این موضوع را به اثبات رساند.

با توجه به این که مقادیر آستانه T به طور تصادفی در بازه $[0,1]$ انتخاب می‌شود، احتمال دارد برای یک جرم، هیچ آستانه‌ای بزرگ‌تر مساوی 0.5 نباشد (هیچ مرکز خوشه‌ای فعال نباشد). برای رفع این مشکل با توجه به این که حداقل تعداد خوشه‌ها برابر دو خوشه است، از بین مقادیر آستانه، دو مقدار به صورت تصادفی انتخاب، و مجدداً به طور تصادفی در بازه $[0,1]$ مقادیر می‌شود. از طرفی در حین اجرای الگوریتم، ممکن است مقادیر آستانه به مقادیری بیشتر از 1 و یا کم‌تر از 0 تغییر پیدا کند، که در این صورت، مقادیر آستانه بیشتر از 1 به 1 و کم‌تر از 0 به 0 تغییر داده می‌شود. به منظور ارزیابی

تکرار الگوریتم، بهتر از تکرارهای قبلی آن و یا حداقل برابر آن باشد، که شبیه یک حافظه برای نگهداری بهترین نقاط در طی اجرای الگوریتم عمل می‌کند، در صورتیکه در الگوریتم GSA چنین خصوصیتی وجود ندارد و ممکن است برازندگی بهترین جرم در یک تکرار، بسیار بدتر از تکرارهای خیلی قبل‌تر از آن باشد. نتایج ارزیابی PABH-GSA در بخش ۱-۷- وارد شده است.

۴. الگوریتم خوشه‌یابی نظارت نشده با استفاده از PABH-GSA

به منظور خوشه‌یابی با استفاده از PABH-GSA به این صورت عمل شد که موقعیت هر ذره در فضای پاسخ مسئله، نشان‌دهنده تعدادی مرکز خوشه پیشنهادی و تعدادی آستانه جهت فعال یا غیر فعال کردن هر یک از این مراکز خوشه می‌باشد، تعداد این مراکز خوشه، برابر حداقل تعداد خوشه تعیین شده توسط کاربر است و برای هر ذره، فقط مرکز خوشه‌ای که آستانه مربوط به آن، فعال شده است، به عنوان یک مرکز خوشه پیشنهادی مورد قبول بوده و از بقیه مراکز خوشه صرف نظر می‌شود. برای درک بهتر، اگر یک مجموعه داده دارای d بعد باشد و برای آن حداکثر K مرکز خوشه در نظر گرفته شود، هر ذره در فضایی با ابعاد $K+K \times d$ سیر می‌کند که K بعد اول، به عنوان آستانه Tr و بقیه ابعاد به صورت بردارهای d بعدی و به عنوان مراکز خوشه پیشنهادی توسط این ذره در نظر گرفته می‌شود. مقادیر آستانه Tr (بعد اول) در بازه $[0,1]$ قرار دارد، اگر $Tr \geq 0.5$ باشد، مرکز خوشه متناظر با آن فعال، و در غیر این صورت ($Tr < 0.5$)، مرکز خوشه متناظر با آن غیر فعال می‌گردد. در زیر، بردار مکان هر ذره، برای یک مسئله خوشه‌یابی با $d=2$ و $K=3$ را نشان می‌دهد، که در آن $Tr1$ آستانه مربوط به مرکز خوشه $M1$ است.

Tr1	Tr2	Tr3	M11	M12	M21	M22	M31	M32
-----	-----	-----	-----	-----	-----	-----	-----	-----

هر ذره در هر تکرار الگوریتم، توسط یک تابع برازندگی ارزیابی شده و در نهایت، ذره دارای بهترین برازندگی، هم تعداد خوشه‌ها و هم مراکز آنها را مشخص می‌کند.

تاکنون توابع برازندگی (شاخص‌های) مختلفی برای خوشه‌یابی فازی [20], [21] و غیر فازی [19], [22] ارائه شده است. در اینجا از شاخص [19] Davies-Bouldin (DB) به عنوان تابع برازندگی استفاده شده است که در خوشه‌یابی نظارت نشده عملکرد

داده‌هایی با تعداد خوشه کم مناسب بود ولی برای داده‌هایی با پراکندگی زیاد، افزایش بیش از حد تعداد خوشه‌ها را به دنبال داشت. با بررسی فراوان مشخص شد، بهترین ایده، استفاده از یک ضریب لگاریتمی است و با اعمال آن، شاخص DB جدید با نام NDB به صورت معادله (۱۲) ارائه شد.

$$NDB = \frac{1}{(1 + \ln(n)) \times n} \sum_{i=1}^n R_{i,qt} \quad (12)$$

با تغییر فوق، الگوریتم خوشه‌یابی نظارت نشده پیشنهادی، مجدداً با مجموعه داده‌های مصنوعی مورد ارزیابی قرار گرفت که نتایج مربوط به آن در بخش ۲-۷ قرار گرفته است.

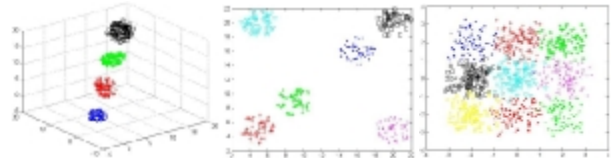
۵. کوانتیزه‌سازی تصویر با استفاده از الگوریتم خوشه‌یابی پیشنهادی

در بعضی موارد به منظور کاهش فضا، جهت ذخیره‌سازی و یا ارسال و دریافت یک تصویر، نیاز است تا تصویر مورد نظر فشرده گردد. کوانتیزه‌سازی تصویر، یک روش فشرده‌سازی با اتلاف (برگشت ناپذیر) است که به موجب آن تعداد رنگ‌های نمایش داده شده در یک تصویر رنگی دیجیتالی و یا تعداد سطوح خاکستری در یک تصویر سطح خاکستری کاهش می‌یابد. روش‌های معمول برای این کار، استفاده از آستانه‌گذاری و یا الگوریتم‌هایی مانند K-means است. در این روش‌ها، به علت عدم انتخاب تعداد رنگ‌های بهینه و یا خوشه‌یابی نامناسب، تصویر کوانتیزه شده ممکن است از کیفیت مناسبی برخوردار نباشد.

با توجه به عملکرد مناسب الگوریتم پیشنهادی برای مجموعه داده‌های مورد آزمایش، تلاش شد تا آن به منظور خوشه‌یابی تعدادی تصویر مرجع (با سطح خاکستری) استفاده شود و نتایج به دست آمده با نتایج GCUK و VLIGA موجود در مرجع [14] مقایسه گردد. با توجه به این که در مرجع ذکر شده شاخص DB به عنوان تابع برازندگی مورد استفاده قرار گرفته است، در اینجا نیز جهت مقایسه نتایج از این شاخص استفاده شد.

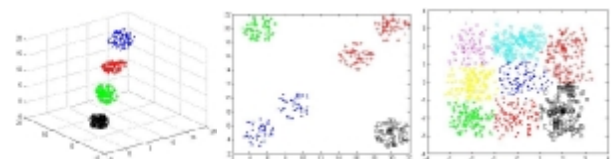
به منظور کاهش هزینه محاسباتی، به جای خوشه‌یابی مقادیر پیکسل‌ها، هیستوگرام تصویر خوشه‌یابی شد. بدین صورت که، در یک تصویر سطح خاکستری که مقدار هر پیکسل با ۸ بیت داده مشخص شده، تمام داده‌ها را می‌توان به ۲۵۶ سطح تقسیم کرد که هر سطح دارای تعداد مشخصی پیکسل است. حال به

الگوریتم پیشنهادی با استفاده از شاخص DB از ۳ مجموعه داده مصنوعی [23] به صورت شکل (۱) استفاده شد.



شکل (۱) سه مجموعه داده که تعداد خوشه‌ها از راست به چپ به ترتیب ۹، ۶ و ۴ و تعداد داده‌ها به ترتیب برابر ۹۰۰، ۳۰۰ و ۴۰۰ داده می‌باشد

نتایج خوشه‌یابی نظارت نشده برای مجموعه داده‌های فوق با تعداد جرم $N=50$ ، تعداد تکرار $T=100$ و $\tau = 0.02$ با حداکثر تعداد خوشه $K=15$ ، در شکل (۲) نشان داده شده است.



شکل (۲) نتایج خوشه‌یابی با الگوریتم پیشنهادی برای سه مجموعه داده

همان طور که در شکل (۲) دیده می‌شود، الگوریتم برای مجموعه داده ۹ خوشه‌ای، تعداد ۸ خوشه و برای مجموعه داده ۶ خوشه‌ای، تعداد ۴ خوشه را نتیجه داده است. در ابتدا ممکن است این نتیجه ناشی از عملکرد نامناسب الگوریتم به نظر برسد، ولی با بررسی شاخص DB مشخص می‌شود که برای مجموعه ۶ خوشه‌ای، شاخص DB برای تعداد ۴ خوشه، نسبت به تعداد ۶ خوشه نتیجه بهتری به دست می‌دهد. بنابراین نتیجه نامناسب به دست آمده، به دلیل استفاده از شاخص DB است. با دقت در معادله (۱۱) مربوط به شاخص DB دیده می‌شود که تابع سیگما با افزایش تعداد خوشه‌ها و n موجود در مخرج با کاهش تعداد خوشه‌ها مخالفت می‌کند ولی از آنجاکه با افزایش تعداد خوشه‌ها، نرخ رشد تابع سیگما به مراتب بیشتر از نرخ رشد n موجود در مخرج است، استفاده از شاخص DB، تعداد خوشه‌های کمتری را نتیجه می‌دهد. برای رفع مشکل فوق، سعی شد با تغییر در معادله، از جمله استفاده از توان‌های بالاتر برای n ، کاهش نرخ رشد آن را جبران کرد، این تغییر برای

دلیل این که یکی دارای شدت روشنایی بیشتری است، درون دو خوشه مجزا قرار بگیرند. با توجه به آنچه گفته شد، برای ناحیه‌بندی تصویر براساس رنگ، معمولاً از فضای رنگی RGB استفاده نمی‌شود و این فضای رنگ صرفاً به منظور کوانتیزه‌سازی (فشرده‌سازی) مورد استفاده قرار می‌گیرد. فضاهای رنگ دیگری مانند HSV و Lab وجود دارد که در آنها اطلاعات شدت روشنایی از اطلاعات رنگی متمایز شده است که برای ناحیه‌بندی بر اساس رنگ، معمولاً از این گونه فضاهای رنگی استفاده می‌شود. با توجه به این که اطلاعات رنگ، در فضای رنگی Lab نسبت به سایر فضاهای رنگی، از همواری بیشتری برخوردار است، در این تحقیق از فضای رنگی Lab استفاده شد.

از آنجاکه برای ناحیه‌بندی با استفاده از خوشه‌یابی، هر پیکسل به عنوان یک داده ورودی در نظر گرفته می‌شود، برای تصاویر با ابعاد بالا، هزینه محاسباتی زیادی صرف می‌شود، برای رفع این مشکل ابتدا هر تصویر ورودی به ابعاد 160×160 تغییر داده شد و پس از خوشه‌یابی تصویر جدید، نتایج به دست آمده مجدداً به اندازه اصلی بازگردانده شد. با این کار، برای تمام تصاویر ورودی، فقط ۲۵۶۰۰ داده خوشه‌یابی می‌شود که سبب صرفه‌جویی زیادی در هزینه محاسباتی (به خصوص برای تصاویر با ابعاد بالا) خواهد شد. از طرفی با کاهش ابعاد تصویر، مقداری خطا به فرایند ناحیه‌بندی تحمیل می‌شود ولی نتایج به دست آمده برای تعدادی تصویر مورد آزمایش نشان داد که این میزان خطا در مقابل کاهش هزینه محاسباتی به دست آمده بسیار ناچیز است.

روند ناحیه‌بندی در روش پیشنهادی به این صورت است که در ابتدا، تصویر ورودی مانند شکل (۳) به یک تصویر با ابعاد 160×160 مطابق شکل (۴) تغییر داده می‌شود، سپس این تصویر مانند شکل (۵) به فضای رنگ Lab انتقال داده می‌شود.



شکل (۳) یک نمونه تصویر ورودی مورد آزمایش جهت ناحیه‌بندی

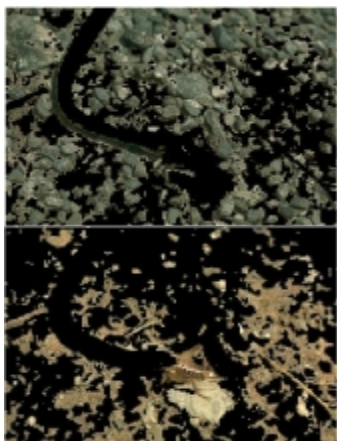
جای خوشه‌یابی هر پیکسل، می‌توان سطوح را خوشه‌یابی کرد. از آنجا که تعداد پیکسل‌های تصویر به مراتب از تعداد سطوح خاکستری (مخصوصاً برای تصاویر با ابعاد زیاد) بیشتر است، خوشه‌یابی هیستوگرام تصویر هزینه محاسباتی را به شدت کاهش می‌دهد. نتایج مربوط به این قسمت در بخش ۳-۷ وارد شده است.

۶. ناحیه‌بندی تصویر با استفاده از الگوریتم خوشه‌یابی پیشنهادی

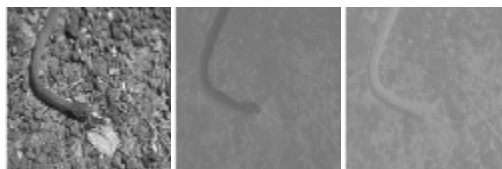
ناحیه‌بندی تصویر یک روند اساسی در برخی کاربردهای پردازش تصویر و ماشین بینایی است و می‌توان آن را به عنوان اولین اقدام پردازش سطح پایین در بازشناسی الگو و پردازش تصویر در نظر گرفت. هدف از ناحیه‌بندی تصویر، تقسیم یک تصویر به تعدادی ناحیه است که این نواحی، اشیاء یا قسمت‌های آنها را نمایش می‌دهد. روش‌های ناحیه‌بندی موجود می‌توان به دو دسته کلی ناحیه‌بندی براساس اطلاعات ناحیه و ناحیه‌بندی براساس اطلاعات نقطه، تقسیم کرد. برای ناحیه‌بندی براساس اطلاعات ناحیه، روش‌های مختلفی وجود دارد که از آن جمله می‌توان به ناحیه‌بندی براساس اطلاعات بافت [24]، ناحیه‌بندی براساس اطلاعات لبه [25]، ناحیه‌بندی با استفاده از تبدیل حوضچه [26] و... اشاره کرد. در اینجا منظور از ناحیه‌بندی تصویر، ناحیه‌بندی براساس اطلاعات نقطه است. در این روش، شدت روشنایی یا اطلاعات رنگ هر پیکسل به عنوان داده ورودی جهت خوشه‌یابی در نظر گرفته می‌شود.

در یک تصویر رنگی RGB، به هر پیکسل، یک بردار عددی ۳ بعدی اختصاص داده می‌شود که هر بعد مقدار روشنایی هر یک از رنگ‌های قرمز، سبز و آبی را مشخص می‌کند. به ازای مقادیر مختلف اختصاص یافته به هر یک از این سه رنگ اصلی، رنگ‌های مختلف به ایجاد می‌شود. حال اگر اطلاعات رنگی یک پیکسل، به عنوان یک مرکز خوشه سه بعدی در نظر گرفته شود، با استفاده از خوشه‌یابی می‌توان بردارهای رنگی نزدیک به این مرکز خوشه را درون یک خوشه قرار داد. به عبارت دیگر، با استفاده از خوشه‌یابی می‌توان یک تصویر را براساس اطلاعات رنگ آن ناحیه‌بندی کرد.

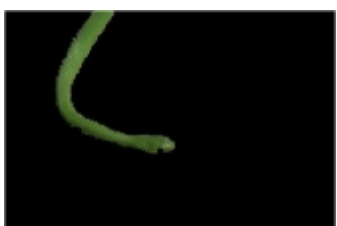
یکی از مشکلات فضای رنگ RGB این است که به ازای تغییرات در شدت روشنایی هر یک از رنگ‌های اصلی، شدت روشنایی و کنتراست رنگ به دست آمده تغییر می‌کند بنابراین ممکن است مثلاً دو پیکسل که دارای یک رنگ هستند، به



شکل (۴) تغییر ابعاد تصویر، به ابعاد جدید ۱۶۰×۱۶۰



شکل (۵) انتقال تصویر به فضای رنگ Lab (به ترتیب از چپ به راست تصویر اول مربوط به L، دوم a و سوم b)



شکل (۷) بازگرداندن نتایج ناحیه‌بندی شکل (۶) به ابعاد تصویر اولیه

در فضای رنگی Lab، لایه L، مقادیر شدت روشنایی هر رنگ، لایه a اطلاعات رنگ‌های ترکیبی سبز و مگنتا و لایه b اطلاعات رنگ‌های ترکیبی آبی و زرد را مشخص می‌کند. برای ناحیه‌بندی براساس رنگ، فقط لایه‌های a و b مورد استفاده قرار می‌گیرد که به صورت یک مجموعه داده ۲ بعدی (به عنوان ورودی الگوریتم پیشنهادی) در نظر گرفته می‌شود.

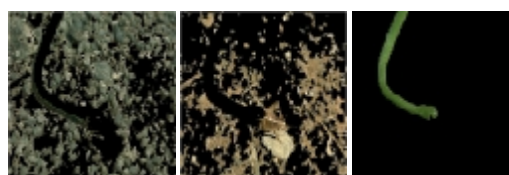
الگوریتم پیشنهادی بر روی تعدادی از تصاویر مرجع برگرفته شده از سایت دانشگاه برکلی بررسی شد که جزئیات مربوط به آن در بخش ۴-۷-۷ آمده است.

نتایج خوشه‌یابی داده‌های ورودی، در شکل (۶) نمایش داده شده است. برای الگوریتم پیشنهادی، تعداد جرم‌ها $N=50$ ، تعداد تکرار $T=50$ ، $\tau = 0.02$ و حداکثر تعداد خوشه‌ها $K=5$ انتخاب شده است و این الگوریتم، تعداد بهینه خوشه‌ها را برابر ۳ خوشه نتیجه داده است.

۷. نتایج و ارزیابی

۷-۱- ارزیابی الگوریتم پیشنهادی PABH-GSA

الگوریتم پیشنهادی PABH-GSA برای ۲۳ تابع محک استاندارد [17],[18] بررسی شد. جدول (۱) نتایج به دست آمده برای PABH-GSA در مقایسه با نتایج PSO و GSA را نشان می‌دهد. برای PABH-GSA، $\tau = 0.02$ در نظر گرفته شده است.

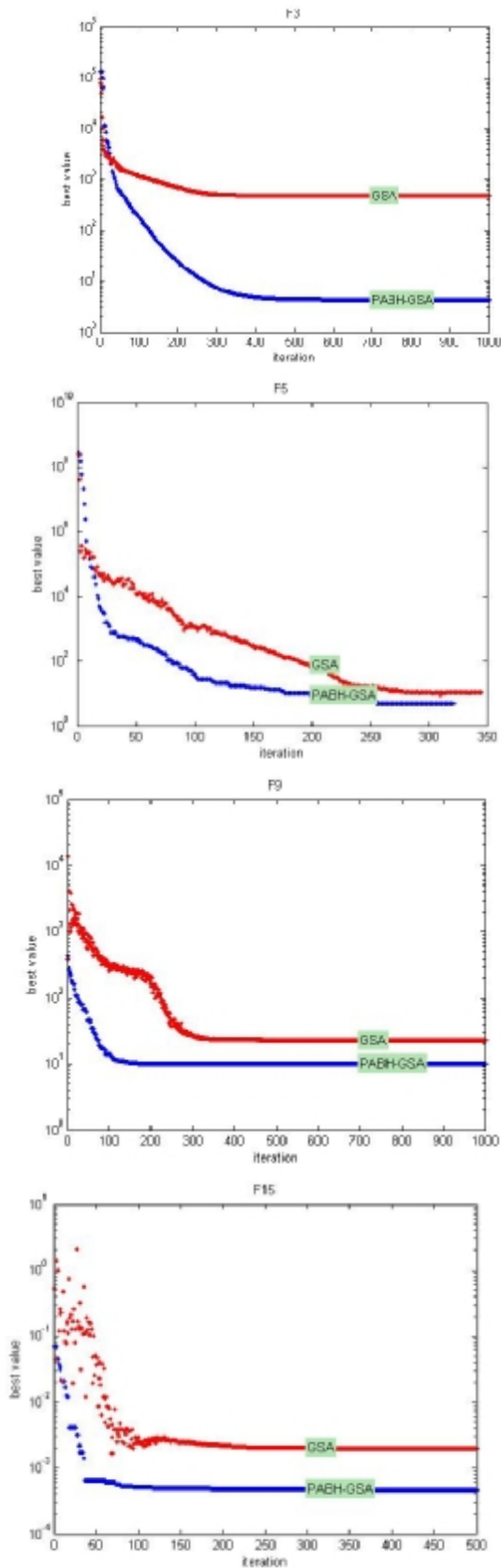


شکل (۶) ناحیه‌بندی به دست آمده با توجه به نتایج خوشه‌یابی داده‌های ورودی برای حداکثر تعداد ۵ خوشه

جدول (۱) مقایسه نتایج الگوریتم پیشنهادی برای ۲۳ تابع محک [17],[18]

f(t)	GSA	PSO	PABH-GSA	Global minimum
F1	2.1×10^{-10}	5×10^{-2}	$(6.4 \pm 1.8) \times 10^{-16}$	0
F2	6.9×10^{-5}	2	$(1.5 \pm 0.1) \times 10^{-8}$	0
F3	$\frac{0}{16 \times 10^{+3}}$	$2.9 \times 10^{+4}$	12.64 ± 6.8	0

پس از اینکه ناحیه‌بندی توسط الگوریتم پیشنهادی انجام شد، مجدداً مطابق شکل (۷)، تصاویر ناحیه‌بندی شده به اندازه اصلی بازگردانده می‌شود.



شکل (۸) نمودار بهترین مقدار برازندگی GSA و PABH-GSA برای ۴ تابع محک در تکرارهای مختلف الگوریتم

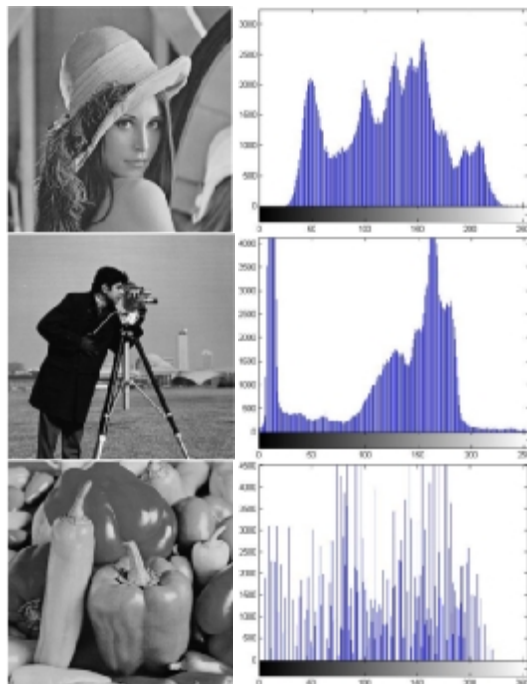
F4	8.5×10^{-6}	23.6	(1. $11 \pm 0.2) \times 10^{-8}$	0
F5	25.16	$3.7 \times 10^{+4}$	7.3 ± 2.7	0
F6	$2.6 \times 10^7_{10}$	0.02	0	0
F7	0.533	1.04	0.082 ± 0.02	0
F8	$-1.1 \times 10^{+3}$	$-9.8 \times 10^{+3}$	(-3. $26 \pm 1.2) \times 10^{+3}$	-12569.5
F9	15.32	72.8	9.65 ± 3.3	0
F10	1.1×10^{-5}	0.02	(6. $76 \pm 1.1) \times 10^{-9}$	0
F11	0.29	0.055	$(3.2 \pm 0.8) \times 10^{-3}$	0
F12	0.051	0.11	0.75 ± 0.25	0
F13	$2.4 \times 10^7_{14}$	2.2×10^{-23}	(4. $55 \pm 0.8) \times 10^{-17}$	0
F14	2.98	0.998	7 ± 5.97	1
F15	5.1×10^{-3}	7.1×10^{-4}	(4. $97 \pm 0.9) \times 10^{-4}$	3.075×10^{-4}
F16	-1.0316	-1.0316	-1.0316	-1.0316
F17	0.3979	0.3979	0.3979	0.398
F18	3	3	3	3
F19	-3.8438	-3.8628	-3.8628	-3.86
F20	-3.0867	-3.2369	-3.3224	-3.32
F21	-5.3278	-6.6290	-10.1531	-10.1532
F22	-10.402	-10.402	-10.4029	-10.4029
F23	-10.536	-10.536	-10.5364	-10.5364

مقایسه نتایج به دست آمده با نتایج GSA و PSO[17]، یک بهبود چشمگیر در دقت الگوریتم GSA را نشان می‌دهد. شکل (۸) نمودار بهترین مقدار برازندگی در هر بار تکرار الگوریتم GSA و PABH-GSA برای ۴ تابع محک F3، F5، F9 و F15 را مشخص می‌کند. همان طور که نمودارهای به دست آمده نشان می‌دهد، الگوریتم پیشنهادی علاوه بر افزایش دقت نسبت GSA، سرعت همگرایی بیشتری دارد. برای جزئیات بیشتر در مورد توابع محک به مراجع [17]، [18] رجوع کنید.

هم تعداد بهینه خوشه‌ها را به درستی انتخاب کرده و هم خوشه‌یابی بهینه‌ای انجام داده است.

۹-۱- ارزیابی الگوریتم، در کاربرد کوانتیزه‌سازی تصاویر

به منظور ارزیابی الگوریتم پیشنهادی، در کاربرد کوانتیزه‌سازی تصویر، از ۳ تصویر مرجع LENA، CAMERAMAN و PEPPERS استفاده شد که نتایج مربوط به آن در جدول (۳) مشخص شده است. برای الگوریتم $N=50$ ، $K=20$ ، $\tau=0.02$ و $T=100$ انتخاب و الگوریتم ۱۰ بار به طور مستقل اجرا شد. برای GSA نیز، $\alpha=2$ ، $G0=10$ و بقیه پارامترها شبیه PABH-GSA می‌باشد. در اینجا به منظور مقایسه با نتایج VLIGA، از شاخص DB با $q=t=1$ استفاده شده است.

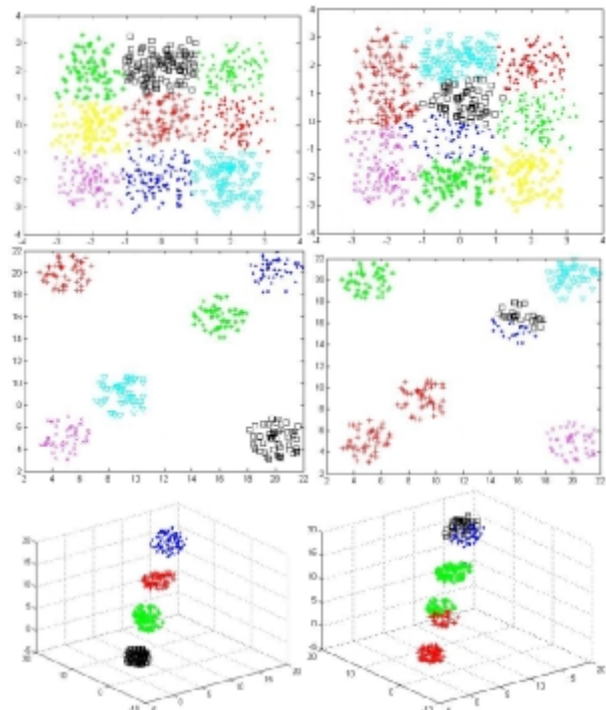


شکل (۱۰) سه تصویر مرجع مورد آزمایش به همراه هیستوگرام آن



۸-۱- ارزیابی الگوریتم خوشه‌یابی نظارت نشده، با استفاده از PABH-GSA

نتایج خوشه‌یابی برای سه مجموعه داده شکل (۱) با استفاده از NDB و همچنین مقایسه آن با نتایج الگوریتم K-means در شکل (۹) نشان داده شده است. مقدار NDB به دست آمده در هر مورد، در جدول (۲) مشخص شده است.



شکل (۹) نتایج خوشه‌یابی ۳ مجموعه داده، ستون سمت راست نتایج خوشه‌یابی با الگوریتم K-means و ستون سمت چپ نتایج خوشه‌یابی نظارت نشده با PABH-GSA

جدول (۲) مقدار شاخص NDB برای نتایج نشان داده شده در شکل (۴)

الگوریتم	مجموعه ۴ تایی	مجموعه ۶ تایی	مجموعه ۹ تایی
K-means	۰/۴۳۸۱	۰/۲۴۸۵	۰/۲۴۶۸
PABH-GSA	۰/۱۹۱۰	۰/۱۳۷۹	۰/۲۳۱۰

همان طور که در شکل (۹) دیده می‌شود، با توجه به مشخص بودن تعداد بهینه خوشه‌ها، در الگوریتم K-means، خوشه‌یابی مناسبی انجام نشده و در یک بهینه محلی گرفتار شده است، ولی الگوریتم پیشنهادی، با وجود حداکثر تعداد خوشه $K=15$ ،

بسیار کم می‌باشد، لذا برای رفع این مشکل از شاخص NDB استفاده شد که نتایج آن برای GSA و PABH-GSA در جدول (۴) و شکل (۱۲) مشخص شده است.

جدول (۴) نتایج برازندگی NDB و تعداد خوشه‌ها K برای سه تصویر مورد آزمایش به ازای الگوریتم‌های GSA و PABH-GSA

IMAGE		GSA	PABH-GSA
LENA	NDB	0. 2062± 0. 0306	0. 1458± 0. 0065
	K	6-14	12-16
CAMERA MAN	NDB	0. 1814± 0. 0183	0. 1380± 0. 0040
	K	8-14	12-15
PEPPERS	NDB	0. 1893± 0. 0259	0. 1355± 0. 0069
	K	6-13	12-15

با توجه به جدول (۴) و تصاویر به دست آمده مشخص می‌شود که استفاده از شاخص NDB، باعث شده تا الگوریتم تعداد مناسب خوشه‌ها را به دست آورد، به‌طوری‌که تصاویر خروجی PABH-GSA (شکل (۶)) که تنها دارای ۱۱ الی ۱۲ سطح خاکستری هستند، تفاوت چندانی با تصاویری اصلی (دارای ۲۵۶ سطح) ندارند. از طرفی مقایسه این نتایج با نتایج GSA، نشان می‌دهد که PABH-GSA نسبت به GSA، عملکرد بسیار مناسب‌تری دارد. مطابق جدول (۴) نتایج الگوریتم GSA، دارای برازندگی کمتر و پراکندگی بیشتری (حتی در مورد تعداد خوشه‌ها) می‌باشد.



شکل (۱۱) نتایج خوشه‌یابی نظارت نشده برای سه تصویر ورودی - ستون سمت راست مربوط به PABH-GSA و ستون سمت چپ مربوط به GSA است. هرکدام از شکل‌ها با کمترین تعداد خوشه به دست آمده مطابق جدول (۳) نشان داده شده است.

جدول (۳) نتایج برازندگی DB و تعداد خوشه‌ها K برای سه تصویر مورد آزمایش به ازای الگوریتم‌های مختلف

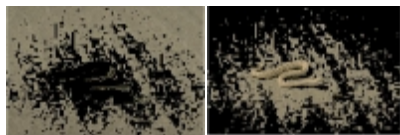
IMAGE		VLIG A	GCU K	GSA	PABH-GSA
LENA	D	0.	0.	0. 4802± 0. 0356	0. 4613± 0. 0091
	B	5203	5309		
	K	5	5	5-11	4-5
CAMERA MAN	D	0.	0.	0. 4650± 0. 0505	0. 2690± 0. 0149
	B	4262	4623		
	K	5	4	3-7	2
PEPPERS	D	0.	0.	0. 4773± 0. 0187	0. 4559± 0. 0074
	B	5192	5343		
	K	7	4	4-13	11-15

همان‌طور که جدول (۳) نشان می‌دهد، PABH-GSA، به ازای تمام تصاویر، نتایج بهتری دارد. نتیجه الگوریتم GSA نیز برای تصویر LENA و PEPPERS نسبت به VLIGA و GCUK بهتر است ولی از مقایسه نتایج GSA با PABH-GSA مشخص می‌شود که نتایج GSA، چه از نظر برازندگی DB و چه از نظر تعداد خوشه K، پراکندگی زیادی دارد و این نشان می‌دهد که GSA، در اجراهای مختلف، در بهینه‌های محلی متفاوتی گرفتار می‌شود. همچنین با توجه به توضیحات قبل در مورد شاخص DB، به جز تصویر PEPPERS که هیستوگرام آن پراکندگی زیادی دارد، در بقیه موارد، تعداد خوشه‌های به دست آمده

شکل(۱۴) نتایج ناحیه بندی تصویر شماره ۱ با استفاده از PABH-GSA برای حداکثر تعداد ۴ خوشه



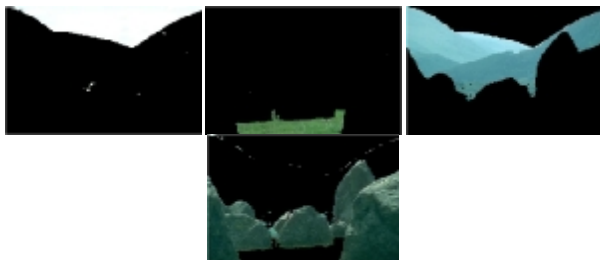
شکل(۱۵) نتایج ناحیه بندی تصویر شماره ۱ با استفاده از GSA برای حداکثر تعداد ۴ خوشه



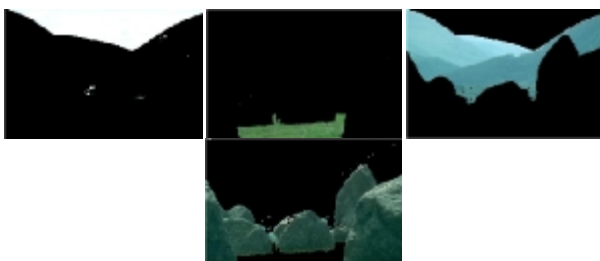
شکل(۱۶) نتایج ناحیه بندی تصویر شماره ۱ با استفاده از K-means



شکل(۱۷) تصویر شماره ۲



شکل(۱۸) نتایج ناحیه بندی تصویر شماره ۲ با استفاده از PABH-GSA برای حداکثر تعداد ۱۰ خوشه



شکل(۱۹) نتایج ناحیه بندی تصویر شماره ۲ با استفاده از GSA برای حداکثر تعداد ۱۰ خوشه

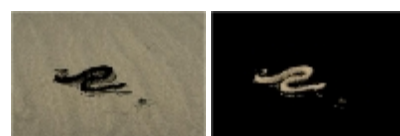


شکل (۱۲) نتایج خوشه‌یابی نظارت نشده برای سه تصویر ورودی با استفاده از شاخص NDB - ستون سمت راست مربوط به PABH-GSA و ستون سمت چپ مربوط به GSA است. هر کدام از شکل‌ها با کمترین تعداد خوشه به دست آمده مطابق جدول(۴) نشان داده شده است.

۱-۱۰-۱- ارزیابی الگوریتم، در کاربرد ناحیه بندی تصاویر جهت ارزیابی الگوریتم پیشنهادی در کاربرد ناحیه بندی تصاویر و مقایسه آن با، GSA و K-means، تعدادی تصویر مرجع از سایت دانشگاه برکلی انتخاب شد. برای PABH-GSA، $N=50$ ، $T=50$ و $\tau=0.02$ تعیین شد و برای GSA نیز، $\alpha=2$ و $G_0=10$ و بقیه پارامترها شبیه PABH-GSA تنظیم گردید. با توجه به نتایج به دست آمده در شکل‌های (۱۳) الی (۴۰) می‌توان گفت که PABH-GSA نسبت به GSA و K-means، نتایج به مراتب بهتری ارائه داده است. همان طور که دیده می‌شود، در بعضی از موارد، تعداد خوشه‌های به دست آمده توسط GSA، بهینه نیست و نتایج الگوریتم K-means، نیز در اکثر موارد مناسب نمی‌باشد. تنها پاسخ مناسب K-means، برای تصویر شماره ۷ است که در آن، هم تعداد خوشه‌ها کم بوده (برابر ۲ خوشه) و هم فاصله آنها از یکدیگر زیاد است. لازم به ذکر است که در تمام موارد، برای الگوریتم K-means، تعداد خوشه‌ها برابر تعداد خوشه‌های به دست آمده توسط روش پیشنهادی در نظر گرفته شده است.

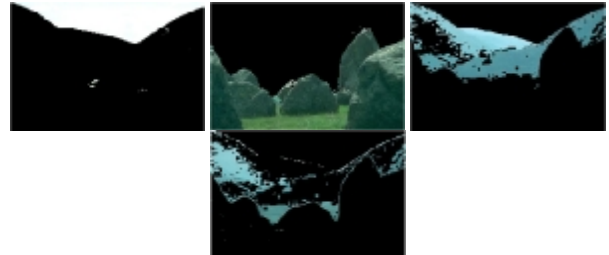


شکل(۱۳) تصویر شماره ۱





شکل (۲۶) نتایج ناحیه‌بندی تصویر شماره ۴ با استفاده از *PABH-GSA* برای حداکثر تعداد ۵ خوشه



شکل (۲۰) نتایج ناحیه‌بندی تصویر شماره ۲ با استفاده از *K-mean*



شکل (۲۷) نتایج ناحیه‌بندی تصویر شماره ۴ با استفاده از *GSA* برای حداکثر تعداد ۵ خوشه



شکل (۲۱) تصویر شماره ۳



شکل (۲۸) نتایج ناحیه‌بندی تصویر شماره ۴ با استفاده از *K-means*



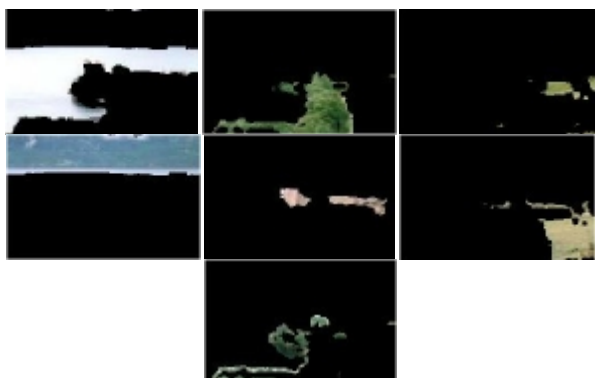
شکل (۲۲) نتایج ناحیه‌بندی تصویر شماره ۳ با استفاده از *PABH-GSA* برای حداکثر تعداد ۵ خوشه



شکل (۲۹) تصویر شماره ۵



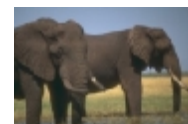
شکل (۲۳) نتایج ناحیه‌بندی تصویر شماره ۳ با استفاده از *GSA* برای حداکثر تعداد ۵ خوشه



شکل (۳۰) نتایج ناحیه‌بندی تصویر شماره ۵ با استفاده از *PABH-GSA* برای حداکثر تعداد ۱۰ خوشه



شکل (۲۴) نتایج ناحیه‌بندی تصویر شماره ۳ با استفاده از *K-means*



شکل (۲۵) تصویر شماره ۴

شکل (۳۵) نتایج ناحیه‌بندی تصویر شماره ۶ با استفاده از GSA
برای حداکثر تعداد ۱۰ خوشه



شکل (۳۶) نتایج ناحیه‌بندی تصویر شماره ۶ با استفاده از K-
means



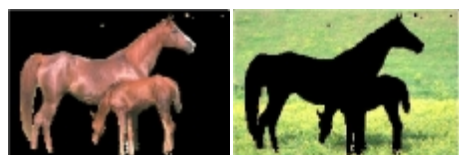
شکل (۳۷) تصویر شماره ۷



شکل (۳۸) نتایج ناحیه‌بندی تصویر شماره ۷ با استفاده از
PABH-GSA برای حداکثر تعداد ۵ خوشه



شکل (۳۹) نتایج ناحیه‌بندی تصویر شماره ۷ با استفاده از
GSA برای حداکثر تعداد ۵ خوشه

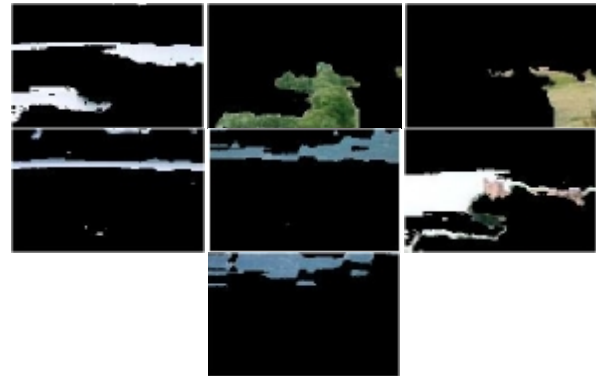


شکل (۴۰) نتایج ناحیه‌بندی تصویر شماره ۷ با استفاده از K-
means

جدول (۵)، نتایج عددی به دست آمده را نشان می‌دهد. با توجه به این جدول و تصاویر به دست آمده، مشخص می‌شود که روش پیشنهادی هم تعداد مناسب خوشه‌ها را مشخص کرده و هم خوشه‌یابی بهتری انجام داده است.



شکل (۳۱) نتایج ناحیه‌بندی تصویر شماره ۵ با استفاده از GSA
برای حداکثر تعداد ۱۰ خوشه



شکل (۳۲) نتایج ناحیه‌بندی تصویر شماره ۵ با استفاده از K-
means



شکل (۳۳) تصویر شماره ۶



شکل (۳۴) نتایج ناحیه‌بندی تصویر شماره ۶ با استفاده از
PABH-GSA برای حداکثر تعداد ۱۰ خوشه



الگوریتم خوشه‌یابی نظارت نشده پیشنهادی، برای کوانتیزه‌سازی و ناحیه‌بندی تعدادی از تصاویر به کار رفت. کوانتیزه‌سازی تصویر، فقط بر روی تصاویر سطح خاکستری و با استفاده از هیستوگرام آنها انجام شد و نتایج به دست آمده، با نتایج دو الگوریتم GCUK و VLIGA مورد مقایسه قرار گرفت و مشخص شد عملکرد الگوریتم پیشنهادی برای تصاویر مورد آزمایش، بسیار بهتر از دو الگوریتم دیگر می‌باشد.

به منظور ناحیه‌بندی تصویر با استفاده از الگوریتم پیشنهادی، از فضای رنگی Lab (به علت پاسخ مناسب آن) استفاده شد. نتایج نشان داد، الگوریتم پیشنهادی نسبت به GSA و K-means عملکرد بسیار مناسب‌تری دارد. با توجه به آنچه گفته شد، تغییر انجام شده در GSA به معنای بهبود کامل این الگوریتم نیست و احتمال دارد با ایده‌های دیگر، به نتایج بهتری دست یافت.

در الگوریتم خوشه‌یابی پیشنهادی نیز، می‌توان به جای استفاده از شاخص NDB از شاخص‌های دیگری استفاده کرد. همچنین در ناحیه‌بندی تصویر، استفاده از فضای رنگ Lab، صرفاً به خاطر عملکرد مناسب آن نسبت به فضاهای رنگی دیگر مانند HSV بوده است، در اینجا نیز جا دارد که فضاهای رنگی دیگر نیز مورد بررسی قرار گرفته و یا برای یک کاربرد خاص، فضای رنگ منحصر به فردی تعریف شود.

مراجع

- [1] M. Omran and A. Engelbrecht, "Particle swarm optimization method for image clustering", Int. J. Pattern Recognition Artificial Intell, (2005) .
- [2] M. Omran and A. Salman, "Dynamic clustering using particle swarm optimization with application in unsupervised image classification", Computing and Technology, (2005) .
- [3] K. Jain and R. C. Dubes, "Algorithms for Clustering Data", Prentice-Hall, Englewood Cliffs, NJ, (1988) .
- [4] R. O Duda and P. E hart, "Pattern classification and Scene Analysis", John Willy & Sons, NY, USA, (1973) .
- [5] J. T. Tou and R. C. Gonzalez, "Pattern Recognition Principles", Addison-Wesley, Reading, (1974)
- [6] G. Ball and D. Hall, "A Clustering Technique for Summarizing Multivariate Data", Behavioral Science, (1967) .
- [7] D. Pelleg and A. Moore, "X-means: Extending K-means with efficient estimation of the number of

جدول (۱۶-۳) نتایج برازندگی NDB و تعداد خوشه‌ها K به دست آمده برای تصاویر مورد آزمایش به ازای الگوریتم‌های مختلف

تصویر		PABH-GSA	K-means	GSA
تصویر شماره ۱	NDB	۰/۳۲۴۳	۰/۵۲۲۸	۰/۳۲۷۶
	K	۲	۲	۲
تصویر شماره ۲	NDB	۰/۱۴۸۳	۰/۳۲۳۴	۰/۱۷۹۸
	K	۴	۴	۴
تصویر شماره ۳	NDB	۰/۲۷۷۶	۰/۵۰۹۴	۰/۲۹۳۳
	K	۳	۳	۳
تصویر شماره ۴	NDB	۰/۲۵۱۷	۰/۲۵۷۴	۰/۲۹۴۲
	K	۴	۴	۳
تصویر شماره ۵	NDB	۰/۲۳۹۰	۰/۲۶۰۹	۰/۳۲۳۰
	K	۷	۷	۳
تصویر شماره ۶	NDB	۰/۲۰۲۲	۰/۲۴۲۰	۰/۲۹۸۶
	K	۳	۳	۴
تصویر شماره ۷	NDB	۰/۲۷۸۳	۰/۲۵۹۶	۰/۲۸۷۹
	K	۲	۲	۲

۸. نتیجه‌گیری و بحث

در این تحقیق پس از بررسی نقاط ضعف GSA، با الهام از پدیده سیاه‌چاله‌ها در طبیعت، نوع بهبود یافته آن با نام الگوریتم جستجوی گرانشی سیاره‌ها و سیاه‌چاله‌ها (PABH-GSA) معرفی شد که با توجه به نتایج موجود در جدول (۱) در اکثر موارد نسبت به PSO و GSA عملکرد بسیار مناسب‌تری دارد. سپس تلاش شد تا با استفاده از آن یک الگوریتم خوشه‌یابی نظارت نشده ارائه گردد. جهت ارزیابی خوشه‌یابی انجام شده، پس از بررسی توابع برازندگی مختلف (در کاربرد خوشه‌یابی)، از شاخص DB استفاده شد، این شاخص نسبت به اکثر توابع برازندگی، هزینه محاسباتی بسیار کم‌تری دارد، مهم‌ترین عیب شاخص DB این است که در صورت استفاده از آن، تعداد خوشه‌های به دست آمده بسیار کم می‌باشد که به منظور رفع این مشکل، شاخص NDB با نتایج مناسب آن ارائه شد.

- [25] M. Tabb and N. Ahuja, "Multiscale image segmentation by integrated edge and region detection", IEEE Transactions on Image Processing, (1997).
- [26] S. Beucher and C. Lantuejoul, "Use of watersheds in contour detection", International workshop on image processing, (1979).
- [27] Y. Zhou, Z. Ouyang, J. Liu, G. Sang, "A Novel K-means Image Clustering Algorithm Based on Glowworm Swarm Optimization", PRZEGLĄD ELEKTROTECHNICZNY (Electrical Review), 2012.
- [28] S. Krinidis, M. Krinidis and V. Chatzis, "An Unsupervised Image Clustering Method Based on EEMD Image Histogram", Volume 3, Number 2, April 2012.
- [29] K. Arai, "Image Clustering Method Based on Density Maps Derived from Self-Organizing Mapping: SOM", International Journal of Advanced Computer Science and Applications (IJACSA), VOL. 3, Number 7, 2012.
- clusters", Proceedings of the 17th International Conference on Machine Learning, (2000).
- [8] S. Bandyopadhyay and U. Maulik, "Genetic clustering for automatic evolution of clusters and application to image classification", IEEE pattern recognition, (2002).
- [9] D. E. Goldberg, "Genetic Algorithms in Search, Optimization and machine Learning", Addison-Wesley, New York, (1989).
- [10] J. Kennedy and R. Eberhart, "Particle Swarm Optimization", Proceedings of IEEE International Conference on Neural Networks, Perth, Australia, (1995).
- [11] M. Dorigo and V. Maniezzo, "The ant system: an autocatalytic optimizing process", Technical Report, Politecnico di Milano, Italy, (1991).
- [12] C. F. Tsai and C. W. Tsai, "ACODF: a novel data clustering approach for data mining in large databases", The Journal of Systems and Software, (2004).
- [13] U. Maulick and S. Bandyopadhyay, "genetic Algorithm based Data Clustering Techniques", pattern recognition, (2000).
- [14] V. Katari and S. C. Satapathy, "Hybridized Improved Genetic Algorithm with Variable Length Chromosome for Image Clustering", IJCSNS International Journal of Computer Science and Network Security, (2007).
- [15] D. Swagatam and A. Ajit, "Automatic kernel clustering with a Multi-Elitist Particle Swarm Optimization Algorithm", Pattern Recognition Letters, (2008).
- [16] Z. Yu and O. C. Au, "An Adaptive Unsupervised Approach toward Pixel Clustering and Color Image Segmentation", Pattern Recognition, (2009).
- [17] E. Rashedi and H. Nezamabadi-pour, "GSA: A Gravitational Search Algorithm", Elsevier (2009).
- [18] X. Yao and Y. Liu, "Evolutionary programming made faster", IEEE Transaction on Evolutionary Computation, (1999).
- [19] D. L. Davies and D. W. Bouldin, "A cluster separation measure", IEEE Trans. Pattern Anal. Machine Intell, (1979).
- [20] J. C. Bezdek, "Cluster Validity with Fuzzy Sets", Cybernetics and systems, (1974).
- [21] X. L. Xie and G. A. Beni, "Validity Measure for Fuzzy Clustering", IEEE Trans. Pattern Anal. Mach. Intell, (1991).
- [22] C. H. Chou and M. C. Su, "A new cluster validity measure and its application to image compression", Pattern Anal, (2004).
- [23] <http://www.isical.ac.in/~sanghami/data.html>
- [24] A. C. Bovik and M. Clark, "Multichannel Texture Analysis Using Localized Spatial Filters", IEEE Transactions on Pattern Analysis and Machine Intelligence, (1990).