

ارائه یک الگوریتم خوشه سازی رنگ^۱ جدید بر اساس معیار MinMax و بهینه سازی آن به کمک الگوریتمهای ژنتیکی و استفاده از آن در آشکارسازی عیوب رنگی کاشی‌ها

مرتضی خادمی
دانشکده مهندسی دانشگاه فردوسی مشهد
mkhademi@hotmail.com

مهدی سعادت‌مند طرزجان
دانشکده برق دانشگاه صنعتی خواجه نصیر الدین طوسی
m_saadatmand_tarzjan@yahoo.com

چکیده در این مقاله، الگوریتمی جدید بر پایه‌ی مدلسازی خوشه‌ای برای آشکارسازی عیوب رنگی کاشی‌ها پیشنهاد شده است. عیوب رنگی، نواقصی هستند که رنگ آنها با تمام رنگهای بکار رفته در کاشی متفاوت است. ایده‌ی اصلی این است که ابتدا در مرحله‌ی استخراج ویژگی، رنگهای یک کاشی بدون عیب را خوشه‌بندی می‌کنیم. حاصل انجام این کار تعدادی مرکز خوشه (که تشکیل یک جدول رنگ می‌دهند) و آستانه‌های (بیشترین خطای هر خوشه) نظیر آنها است. سپس در مرحله‌ی بازرسی، جدول رنگ مذکور با قانون نزدیکترین همسایه بر تصویر تحت بازرسی اعمال می‌شود. رنگهایی که خطای آنها بیش از مقدار آستانه‌ی نظیرشان است متعلق به نواحی معیوب کاشی هستند. در این مقاله به منظور افزایش دقت فرآیند مدلسازی، الگوریتم خوشه‌بندی جدیدی ارائه شده است که هدف آن کمینه کردن خطای خوشه‌ها، طبق معیار MinMax است و پاسخهای الگوریتم پیشنهادی به کمک الگوریتمهای ژنتیکی بهینه گردیده است. نتایج تجربی بیانگر دقت الگوریتم پیشنهادی در آشکارسازی عیوب رنگی است.

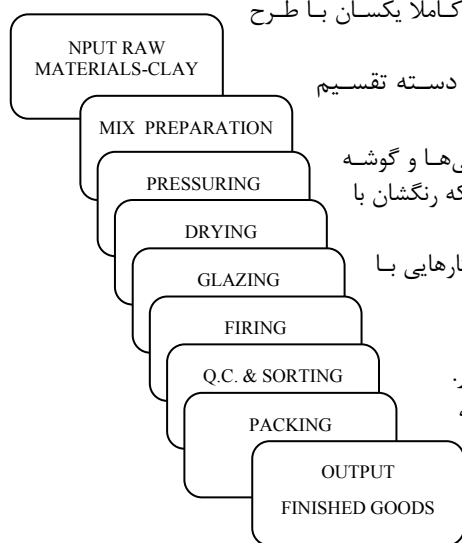
واژگان کلیدی آشکارسازی عیوب کاشی‌ها، دستگاههای مختصات رنگ، مدلسازی خوشه‌ای، معیار MinMax

۱- مقدمه

عبارت محصولات سرامیکی به اشیایی از جنس مواد معدنی غیرفلزی اطلاق می‌شود که بر اساس نیاز شکل داده شده و سپس طی فرآیند معینی مقاومت مکانیکی لازم را بدست آورده‌اند. کاشی‌ها، سرامیکها و چینی‌ها از این دست محصولات هستند (که از این به بعد، جهت سهولت همه‌ی آنها را با عنوان کاشی نام می‌بریم). مراحل تولید این محصولات را می‌توان در شکل ۱ مشاهده کرد. با گسترش فن‌آوری، همه‌ی بخشهای تولید این صنعت بجز مرحله‌ی کنترل کیفیت و درجه‌بندی کاشی‌ها که هنوز به صورت دستی انجام می‌شود و در کل فرآیند تولید، گلوگاهی را بوجود آورده است، بسیار متحول شده است. کنترل کیفیت، مرحله‌ای کلیدی در فرآیند تولید کاشی‌هاست و در کیفیت و قیمت محصول نهایی اهمیتی حیاتی دارد. هم‌اکنون هر کارگر ماهر بر اساس مجموعه‌ای از کاشی‌های نمونه‌ی مرجع، معیارهایی شخصی برای خویش تعیین و بر اساس آنها، کاشی‌ها را درجه‌بندی می‌کند. بدیهی است که اولین گام در درجه‌بندی کاشیها، آشکارسازی عیوب آنهاست. بر اساس نوع رنگ‌آمیزی و طرح، کاشی‌ها را می‌توان به سه دسته‌ی کلی تقسیم نمود:

- (a) **کاشی‌های ساده:** این کاشی‌ها دارای طرح یا الگویی نبوده و تمام سطح آنها به صورت یکدست و یکنواخت، تنها با یک رنگ، رنگ‌آمیزی شده است مانند سرامیکهای ساده و کاشی‌های ساده سنتی.
- (b) **کاشی‌های طرح‌دار:** می‌توان گفت که سطح این کاشی‌ها در حقیقت یک تصویر بافتی است و الزاماً دو کاشی یکسان از نظر درجه‌بندی از نظر طرح کاملاً یکسان نیستند، مانند تمام انواع سرامیکها (به استثنای سرامیکهای ساده).

^۱ color quantization



شکل ۱: مراحل تولید محصولات سرامیکی

(c) کاشی‌های الگودار: همه کاشی‌های الگودار یک خط تولید باید دارای طرحی کاملاً یکسان با طرح مرجع باشند و کوچکترین اختلاف به عنوان عیب در نظر گرفته می‌شود. از دیدگاه اثری که عیوب بر ظاهر کاشی می‌گذارند، می‌توان آنها را به چهار دسته تقسیم نمود:

(a) عیوب رنگی: که ناشی از عیوبی از قبیل خالها، لب‌پریدگی‌ها، لعاب‌نگرفتگی‌ها و گوشه شکستگی‌ها و حتی ترکها و خراشهاست. مهمترین مشخصه این عیوب این است که رنگشان با رنگهای بکاررفته در رنگ‌آمیزی کاشی متفاوت است.

(b) عیوب تیز: از قبیل ترکها، سوراخها، خالها و لکه‌ها. این عیوب در تصویر ساختارهایی با کنتراست بالا بوجود می‌آورند.

(c) عیوب پهن: اشکالات طرح در کاشی‌های طرح‌دار.

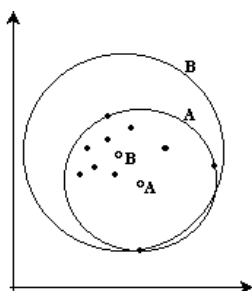
(d) عیوب الگو: عبارت است از هر نوع تخلفی از الگوی مرجع در کاشی‌های الگودار. در این مقاله الگوریتمی جدید برای آشکارسازی عیوب رنگی کاشیها ارائه شده است که مختص نوع خاصی از کاشی‌ها نیست. برای مطالعه در زمینه‌ی آشکارسازی سایر عیوب کاشی‌ها به [1]، [2] و [3] مراجعه کنید. بخشهای بعدی این مقاله بدین شرح‌اند. در بخش بعدی دستگاه مختصات رنگ مورد استفاده‌ی ما در این مقاله معرفی می‌شود. در بخش سوم الگوریتم خوشه‌بندی پیشنهادی شرح داده شده است. بخش چهارم به بیان نتایج شبیه‌سازی‌ها می‌پردازد و بخشهای پنجم و ششم به ترتیب به نتیجه‌گیری و مراجع اختصاص داده شده است.

۲- انتخاب دستگاه مختصات رنگ

همانطور که می‌دانید نور بر روی چشم دو اثر مهم دارد: روشنایی (brightness) که نشان‌دهنده تراکم طول موجهای مختلف است و رنگ که بیانگر ترکیب طیفی آن می‌باشد. توماس یانگ و ماکسول نشان دادند که هر رنگ به صورت ترکیبی از سه رنگ پایه است. همچنین نشان داده شده است که رنگها تشکیل یک فضای برداری می‌دهند. بر این اساس تا کنون، برای بیان رنگها، سیستمهای مختصات مختلفی ارائه شده است که هر یک دارای ویژگیهای خاصی است. به طور کلی مقوله رنگ از سه منظر بررسی می‌شود [4]: ۱- تطابق رنگ (color matching)، ۲- تفاوت بین رنگها (color differences) و ۳- درک رنگ (color appearance). آنچه که در این مقاله اهمیت دارد مقوله "تفاوت بین رنگها" است. این مبحث به دنبال معیاری برای اندازه‌گیری اختلاف دو رنگ است؛ به طوری که حتی الامکان به آنچه چشم درک می‌کند، نزدیک باشد. یکی از روشهایی متداول تحقق این خواسته، طراحی سیستمهای مختصات رنگ خاص است. یکی از معروفترین این سیستمها که در سال ۱۹۷۶ توصیه گردیده، $CIE-L^*u^*v^*$ است (برای مشاهده معادلات تبدیل رنگ به [4] و [5] مراجعه کنید). با اینکه $CIE-L^*u^*v^*$ به منظور اندازه‌گیری اختلاف رنگها پیشنهاد شده است، اما هنوز هم در این فضا اختلاف دو رنگ دقیقاً برابر با فاصله اقلیدسی آنها، نیست (گرچه نسبت به سایر فضاهای رنگی تخمین مناسبتری است) [4]. ما با قبول این تقریب، از این فضای رنگی استفاده می‌کنیم. از اینجا به بعد در این مقاله، بردارهای رنگها همه در این دستگاه مختصات بیان می‌شوند، مگر آنکه خلاف آن صریحاً ذکر شود.

۳- مدلسازی خوشه‌ای

خوشه‌بندی روشی استاندارد برای تجزیه مسائل به اجزای کوچکتر است. این روش داده‌های همسان را در یک کلاسها قرار می‌دهد. می‌توان از مشخصات این کلاسها برای فهمیدن جزئیات مسأله یا یافتن گروههای مشابه در میان داده‌های موجود، سود جست. از جمله زمینه‌هایی که خوشه‌بندی در آنها کاربرد فراوانی دارد می‌توان به فشرده‌سازی اطلاعات و استخراج ویژگی اشاره کرد. اکثر قریب به اتفاق این الگوریتمها با هدف فشرده‌سازی اطلاعات طراحی گردیده‌اند و به همین جهت بیشتر بر روی مشخصه‌های آماری خوشه‌ها (همچون میانگین، واریانس و ...) تکیه دارند [6]، [7] و [8].



شکل ۲: یک خوشه نمونه.

برای تشخیص عیوب رنگی لازم است، برای رنگهای بکار رفته در کاشی مدلی بدست آورد. معمولاً در یک تصویر کاشی بیش از ۱۰۰۰۰۰ پیکسل وجود دارد؛ درحالی‌که تنوع رنگ در آن خیلی زیاد نیست. بنابراین پیکسلها در فضای رنگی خوشه‌هایی تشکیل می‌دهند که می‌توانند مبنای کار مدل‌سازی قرار گیرند. اندیشه اصلی این است که با استخراج مراکز این خوشه‌ها از تصویر یک کاشی مرجع بدون عیب و محاسبه بیشترین

خطای خوشه‌ها^۲، شاخصی از پراکندگی بردارهای رنگی بدست آورد. بدیهی است که در مرحله بازرسی، تمام پیکسلهایی که خطایشان از بیشترین خطای خوشه‌ای که به آن تعلق دارند، بیشتر باشد، متعلق به عیوب رنگی کاشی است.

در این مقاله، هدف ما از استفاده از الگوریتمهای خوشه‌بندی، استخراج ویژگیهایی چون مرکز خوشه‌ها و ماکزیمم خطای آنهاست. بیشترین خطای هر خوشه یا به طور خلاصه خطای هر خوشه رابطه مستقیمی با نحوه انتخاب مرکز خوشه دارد. به عنوان مثال به شکل ۲ توجه کنید. در این شکل برای خوشه موجود دو مرکز متفاوت انتخاب شده است (مراکز با نقاط توخالی و داده‌ها با نقاط توپر مشخص شده‌اند). همانطور که می‌بینید بیشترین خطای خوشه، به ازای این دو مرکز یکسان نیست. این مطلب با دو دایره‌ی A و B نشان داده شده است. مرکز خوشه را باید چنان انتخاب کرد که بیشترین خطای خوشه مینیمم شود. در اینجا تمام داده‌های یک خوشه مهم هستند و باید در مدلسازی برای همه آنها ارزش یکسانی قائل شد، بنابراین روشهای آماری چندان کاربرد ندارند. این مطلب را می‌توان در شکل ۲ نیز دید (مرکز B، بردار میانگین است). با توجه به اینکه هرچه خطای خوشه‌ها کمتر باشد، دقت ما در تشخیص عیوب رنگی بیشتر می‌شود، پس بهترین مرکز، برداری است که بیشترین خطای خوشه را مینیمم کند. برای دستیابی به چنین مرکزی کافی است با استفاده از یک الگوریتم بهینه‌ساز عددی، مرکز اولیه‌ی خوشه را بهینه نمود. معادله (۳) تابع هدف این بهینه‌ساز (که باید مینیمم شود) را نشان می‌دهد.

$$e_i = \sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2} \quad i = 1, 2, \dots, N \quad (1)$$

$$\vec{E} = [e_i] \quad i = 1, 2, \dots, N \quad (2)$$

$$e_{\max} = \|\vec{E}\|_{\infty} = \lim_{p \rightarrow \infty} \sqrt[p]{\sum_{i=1}^N (e_i)^p} \quad (3)$$

که منظور از (x, y, z) مرکز خوشه‌ای است که بردار (x_i, y_i, z_i) متعلق به آن است و باید بهینه شود و N تعداد اعضای خوشه‌ی مذکور است. بدیهی است که معادله‌ی (۳) بیانگر بیشترین خطای خوشه است. تابع $f(x, y, z)$ را طبق معادله‌ی (۴) تعریف کرده و از آن نسبت به x مشتق جزئی می‌گیریم.

$$f(x, y, z) = \sqrt[p]{\sum_{i=1}^N (e_i)^p} \quad (4)$$

$$\frac{\partial f}{\partial x} = \frac{1}{p} \times \frac{1}{f^{p-1}} \times \frac{\partial}{\partial x} \left[\sum_{i=1}^N (e_i)^p \right] = \frac{1}{p} \times \frac{1}{f^{p-1}} \times \left[\sum_{i=1}^N \frac{\partial}{\partial x} (e_i)^p \right] \quad (5)$$

$$\frac{\partial (e_i)^p}{\partial x} = p \times (e_i)^{p-2} \times (x - x_i) \quad (6)$$

از (۵) و (۶)، معادله‌ی (۷) و از آن، معادله‌ی (۸) را می‌توان نتیجه گرفت.

$$\frac{\partial f}{\partial x} = \frac{\sum_{i=1}^N (e_i)^{p-2} \times (x - x_i)}{f^{p-1}} \quad (7)$$

$$\lim_{p \rightarrow \infty} \frac{\partial f}{\partial x} = \frac{x - x_j}{e_{\max}} \quad e_{\max} = e_j \quad (8)$$

اکنون بر اساس معادله (۸) می‌توان یک الگوریتم بازگشتی برای بهینه‌سازی مرکز خوشه‌ها طراحی کرد؛ خصوصاً که محاسبات بالا برای پارامترهای y و z نیز یکسان است. این الگوریتم در جدول ۱ آورده شده است.

همانطور که می‌بینید قلب این الگوریتم معادله‌ی (۱۰) است که نتیجه‌ی مستقیم معادله‌ی (۸) می‌باشد. در حقیقت، ما برای کاهش خطای خوشه‌ها از روش بیشترین شیب سود جست‌ایم [11]؛ به طوری که مراکز در فضا در خلاف جهت مشتق تابع هدف حرکت می‌کنند. سرعت همگرایی و دقت پاسخهای بهینه‌ساز رابطه مستقیمی با مقادیر پارامترهای η^j دارد. اگر این پارامترها بیش از حد کوچک باشند، الگوریتم دیرتر همگرا می‌شود و حتی ممکن است در مینیممهای محلی ضعیف نیز بدام بیافتد و چنانچه بیش از حد بزرگ باشد سبب نوسان و ناپایداری الگوریتم می‌گردد. بنابراین تنظیم مقدار این پارامترها از اهمیت ویژه‌ای برخوردار است. تنظیم η^j ها بسیار شبیه به تنظیم پارامترهای سرعت آموزش، در شبکه‌های عصبی مصنوعی است [9]. ما برای انجام اینکار از روشی که در [10] برای تنظیم خودکار سرعت آموزش شبکه‌های پیش‌رو ارائه شده است، استفاده می‌کنیم. در این شیوه به تغییرات متوالی بردارهای تصحیح توجه می‌شود. مقدار همبستگی دو برداری که مرکز یک خوشه، طی دو مرحله بهینه‌سازی، به اندازه‌ی آنها جابجا شده است را در نظر بگیرید. بسته به مقدار همبستگی سه حالت مختلف می‌توان در نظر گرفت:

^۲ منظور از خطای یک بردار در اینجا، فاصله اقلیدسی بین آن بردار با مرکز خوشه‌ای است که بردار، متعلق به آن است (هر بردار می‌تواند یک رنگ باشد).

۱. اگر مقدار همبستگی نزدیک یک باشد، دو بردار تقریباً همجهت بوده‌اند؛ بنابراین تابع هدف در محل فعلی مرکز خوشه صاف و بدون برجستگی و فرورفتگیهای تند است. پس می‌توان با افزایش η^j ها بر سرعت همگرایی افزود.

۲. اگر مقدار همبستگی نزدیک منهای یک باشد، دو بردار تقریباً در خلاف جهت یکدیگرند؛ بنابراین تابع هدف در محل فعلی مرکز خوشه دارای برجستگیها و فرورفتگیهای تند است. پس برای پرهیز از نوسان الگوریتم در دو لبه‌ی این ناهمواریها η^j ها را باید کاهش داد.

۳. اگر بین دو بردار همبستگی‌ای وجود نداشته باشد، بهتر است η^j ها را تغییری ندهیم. با توجه به مطالب گفته شده معادله‌ی (۱۰) را به شکل ذیل تغییر می‌دهیم.

$$\Delta \bar{C}^j(k) = \bar{V}_{\max}^j - \bar{C}^j(k) \quad (11)$$

$$r^j(k) = \begin{cases} 0 & k = 1 \\ \frac{\Delta \bar{C}^j(k) \cdot \Delta \bar{C}^j(k-1)}{|\Delta \bar{C}^j(k)| \times |\Delta \bar{C}^j(k-1)|} & \text{otherwise} \end{cases} \quad (12)$$

$$\eta^j(k) = \eta^j(k-1) \times (1 + c \times r^j(k)) \quad , c = \begin{cases} .25 & r^j(k) \geq 0 \\ .5 & \text{otherwise} \end{cases} \quad (13)$$

$$\bar{C}^j(k+1) = \bar{C}^j(k) + \eta^j(k) \times \Delta \bar{C}^j(k) \quad (14)$$

که $r^j(k)$ مقدار همبستگی بین بردار جابجایی فعلی و بردار جابجایی مرحله‌ی قبل است. با توجه به اینکه $\|\bar{C}^j(k) - \bar{V}_{\max}^j\|$ عبارتی اسکالر (عددی) است آن را در دل η^j ها جای داده‌ایم. در آغاز کار باید مراکز خوشه‌ها را مقداردهی اولیه کرد. برای اینکار از بین بردارهای ورودی، به صورت تصادفی، m بردار را انتخاب می‌کنیم. مقدار آستانه tr تعیین کننده‌ی دقت خوشه‌بندی است. هرچه این مقدار کوچکتر باشد زمان بیشتری صرف بهینه‌سازی می‌شود و در نتیجه دقت بیشتری نیز بدست می‌آید. اکنون الگوریتم خوشه‌بندی پیشنهادیمان کامل گردیده است، آن را (Min-Max Color clustering) MiMaC می‌نامیم.

جدول ۱: الگوریتم بهینه‌سازی مراکز خوشه‌ها

۱- پارامترهای tr مقدار آستانه‌ی شرط خاتمه، η^j سرعت بهینه‌سازی و \bar{C}^j ($m, j=1, 2, \dots$) مراکز خوشه‌ها را مقداردهی اولیه کنید.

۲- در مرحله‌ی k ام، بردارها را بر اساس قانون نزدیکترین همسایه خوشه‌بندی کنید. به عبارت دیگر هر بردار متعلق به خوشه‌ای است که نسبت به بقیه، به مرکز آن نزدیکتر است.

۳- از میان بردارهای هر خوشه، دورترین بردار از مرکز خوشه را بیابید. در خوشه‌ی j ام، این بردار را با \bar{V}_{\max}^j و فاصله‌ی آن تا مرکز را با $\bar{E}_{\max}^j(k)$ نشان می‌دهیم. چنانچه نامعادله‌ی ذیل برقرار باشد الگوریتم بهینه‌سازی خاتمه می‌یابد.

$$\max_{j=1}^m \left(\bar{E}_{\max}^j(k) - \bar{E}_{\max}^j(k-1) \right) < tr \quad (9)$$

۴- با استفاده از معادله‌ی ذیل مراکز خوشه‌ها را بروز نمایید.

$$\bar{C}^j(k+1) = \bar{C}^j(k) + \eta^j \left(\frac{\bar{V}_{\max}^j - \bar{C}^j(k)}{\|\bar{C}^j(k) - \bar{V}_{\max}^j\|} \right) \quad (10)$$

۵- به k یک واحد افزوده، به ۲ بروید.

۴- بهینه‌سازی MiMaC با استفاده از الگوریتمهای ژنتیکی

در آشکارسازی عیوب رنگی، دقت مدل اهمیت زیادی دارد. هرچه دقت مدل زیادتر شود، بر دقت الگوریتم آشکارسازی افزوده می‌شود. دقت MiMaC به مقداردهی اولیه خوشه‌ها در مرحله اول الگوریتم جدول ۱ وابسته است؛ به طوری که می‌توان با مقداردهی اولیه مناسب، پاسخ را بهینه نمود. به عبارت دیگر هرچه مقدار اولیه خوشه‌ها به مقادیر بهینه نزدیکتر باشد، انتظار می‌رود که الگوریتم به پاسخ‌های مناسبتری همگرا گردد. برای بهینه‌سازی الگوریتم از الگوریتمهای ژنتیکی استفاده کرده‌ایم. الگوریتمهای ژنتیکی خانواده‌ای از مدل‌های محاسباتی هستند که از تکامل موجودات (مسأله انتخاب طبیعی در طبیعت) الهام گرفته شده [12] و اولین بار توسط J. Holland پیشنهاد گردیده‌اند [13]. انتخاب طبیعی یک فرایند زیستی است که بر اساس آن در یک جامعه رقابتی، احتمال برنده شدن افراد قویتر بیشتر است. در یک الگوریتم ژنتیکی فرض می‌شود که حل بالقوه یک مسأله (که آن را کروموزوم می‌نامند)، عضوی از یک جامعه باشد که می‌توان آن را با تعدادی پارامتر نمایش داد. این پارامترها به

عنوان ژنهای کروموزوم مربوطه شناخته می‌شوند. یک تابع هزینه (evaluation function)، نیز برای مشخص کردن شایستگی کروموزومها تعریف می‌شود. طبیعتاً این تعریف به اهداف مسأله بهینه‌سازی وابسته خواهد بود. در یک کاربرد عملی الگوریتم ژنتیکی، باید جمعیت اولیه کروموزومها بصورت تصادفی تولید شود. جمعیت اولیه مذکور، نسل اول را تشکیل می‌دهد. از این پس در هر مرحله، الگوریتم با انتخاب تعدادی از کروموزومهای جمعیت فعلی (والدین) و اعمال عملگرهای برش (crossover) و جهش (mutation)، کروموزومهای نسل بعد (فرزندان) را بوجود می‌آورد. انتظاری که از این فرایند تکاملی (تغییرات ژنها) داریم این است که کروموزوم بهتر، تعداد فرزندان بیشتری تولید کند و در نتیجه امکان بقایش در نسلهای بعد بیشتر باشد. این فرایند ادامه می‌یابد تا شرط خاتمه الگوریتم برآورده شود. امروزه از الگوریتم ژنتیکی در کاربردهای مختلفی مانند بهینه‌سازی توابع، شناسایی سیستمها، پردازش تصویر و... استفاده می‌شود. با این که دامنه کاربرد این الگوریتمها بسیار وسیع است اما اغلب به آنها به چشم «بهینه‌سازهای توابع» نگاه می‌شود [12]. هنگامی که هدف، پیدا کردن جوابی نزدیک به پاسخ بهینه، برای یک تابع چندبعدی با چندین قید باشد، بهینه‌سازهای ژنتیکی بسیار مؤثر و کارا هستند [14]. برای مطالعه بیشتر درباره الگوریتمهای ژنتیکی به [12]، [15] و [16] مراجعه کنید.

همانطور که می‌دانید هر الگوریتم ژنتیکی از بخشهای رمزنگاری (coding)، انتخاب والدین (selection)، برش و جهش تشکیل شده است. فرض کنید بخواهیم M خوشه داشته باشیم. بنابراین باید هر کروموزوم دارای M ژن باشد. هر ژن که برداری سه عنصری است، بیانگر مرکز یک خوشه است. هدف الگوریتم ژنتیکی پیشنهادی، مینیم کردن بیشترین خطای خوشهها است. برای بدست آوردن خوشه‌بندی نهایی نظیر هر کروموزوم، MiMaC را با خوشه‌هایی که آن کروموزوم معرفی می‌کند، مقارنه اولیه کرده، اجازه می‌دهیم الگوریتم همگرا شود. مقدار تابع هدف برای کروموزوم مذکور برابر است با بیشترین خطای خوشهها. شایستگی هر کروموزوم را برای انتخاب شدن به عنوان پدر، با استفاده از تابع انتخاب بر حسب رتبه تعیین می‌کنیم. اگر N تعداد کل کروموزومهای جمعیت و r مرتبه یک کروموزوم در جمعیت از نظر میزان ارزش (به ترتیب نزولی) باشد، شانسی که تابع شایستگی به آن برای شرکت در تولید نسل بعدی به عنوان پدر می‌دهد عبارت است از:

$$w_r = \frac{2(M-r)}{M(M-1)} \quad (15)$$

عملگر برشی که برای تولید فرزندان استفاده شده است، شامل دو بخش است. در مرحله اول نیمی از ژنهای هریک از والدین به فرزند منتقل می‌شود. سپس در مرحله دوم عدد صحیح p - به صورت تصادفی - بین صفر و M انتخاب می‌شود. سپس در کروموزوم فرزند، p ژن به صورت تصادفی انتخاب شده و به جای آنها ژنهایی که با استفاده از عملگر برش یک نقطه‌ای [12] بوجود آمده‌اند³، جایگزین می‌شود. مزیت اساسی این عملگر این است که به بعضی از خوشه‌های والدین اجازه می‌دهد تا در نسل بعد نیز به بقا خود ادامه دهند. به این ترتیب الگوریتم ژنتیکی قادر خواهد بود بخشی از تجربیات مفید خود را در طول تکامل حفظ کند. عملگر جهش مورد استفاده همان عملگر رایج است. برای هر ژن عددی تصادفی تولید می‌شود و چنانچه این عدد کمتر از احتمال جهش بود، ژن مذکور با ژنی که به صورت تصادفی تولید شده، جایگزین می‌شود.

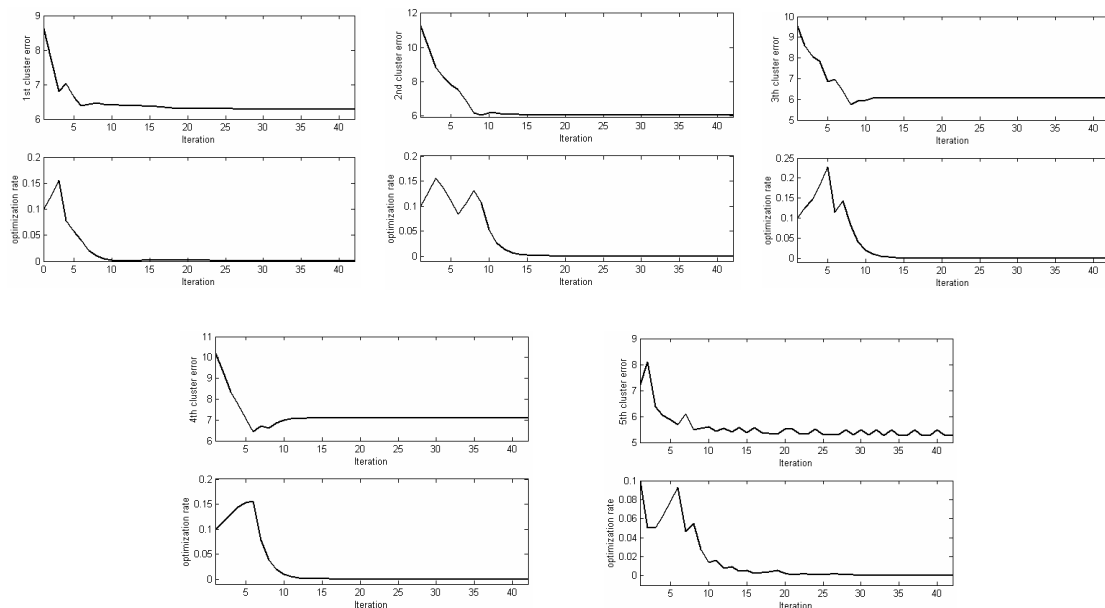
تکامل بیش از آنکه به حفظ گونه‌هایی بپردازد که تطابق بیشتری با محیط اطرافشان داشته‌اند، به حذف گونه‌هایی پرداخته که کمتر با محیط اطرافشان تطبیق داشته‌اند [18]. به عبارت دیگر در تکامل حذف عناصر ضعیفتر به مراتب اثر بیشتری از حفظ عناصر قویتر دارد. در الگوریتم ژنتیکی پیشنهادی مشابه با Genitor [19]، در هر تولید نسل (generation)، از میان جمعیت موجود تنها دو کروموزوم به عنوان ولی (parent) انتخاب می‌شوند و بنابراین تنها یک کروموزوم فرزند تولید می‌گردد. جمعیت جدید با جایگذاری این دو فرزند به جای بدترین کروموزومهای جمعیت قبلی (آنهايي که بدترین درجه شایستگی را دارند) بوجود می‌آید. اکنون الگوریتم ژنتیکی کامل گردیده است. نتایج شبیه‌سازیهای انجام شده در بخش بعدی ذکر گردیده است.

۵- شبیه‌سازی

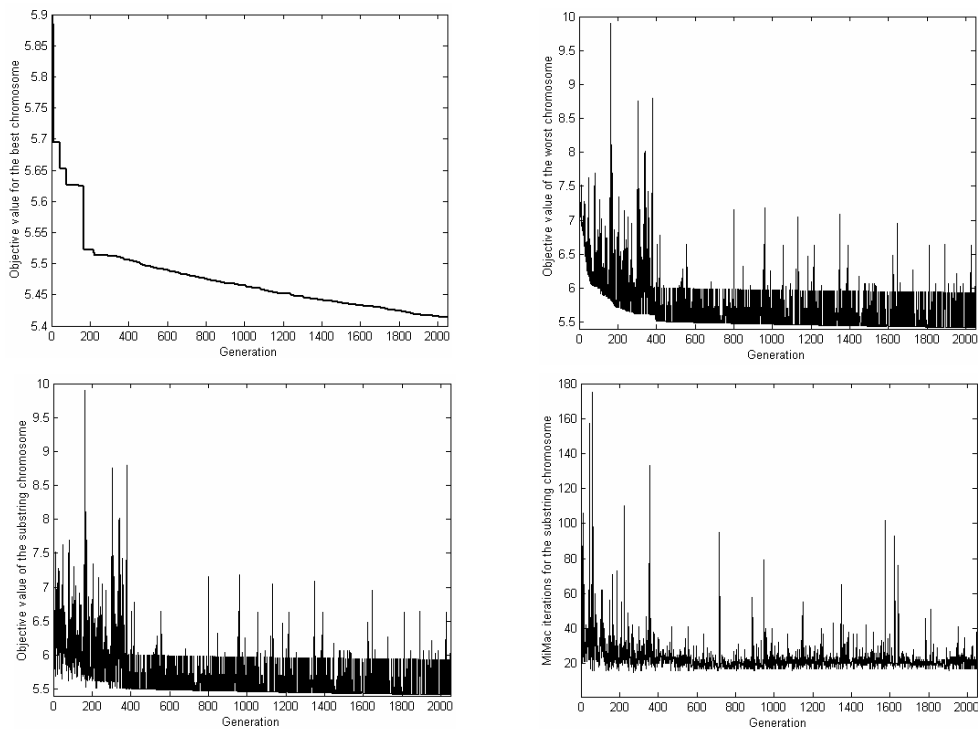
ما از الگوریتم ژنتیکی پیشنهادی برای آشکارسازی عیوب کاشی‌ها استفاده کرده‌ایم؛ به این ترتیب که ابتدا در مرحله‌ی استخراج ویژگی با استفاده از این الگوریتم رنگهای موجود در یک کاشی مرجع را خوشه‌بندی کرده، بیشترین خطای هر خوشه را بدست می‌آوریم. سپس در مرحله‌ی بازرسی، برای آشکارسازی عیوب رنگی کاشی تحت بازرسی، ابتدا با استفاده از خوشه‌های بدست آمده در مرحله‌ی قبل، طبق قانون نزدیکترین همسایه، رنگهای کاشی جدید را خوشه‌بندی کرده، خطای هر بردار رنگ را محاسبه می‌کنیم. بدیهی است که چنانچه خطای برداری بیش از ماکزیمم خطای خوشه‌ی نظیرش باشد، پیکسل نظیر آن بردار متعلق به یک عیب رنگی در سطح کاشی است. شبیه‌سازیها بر روی بیش از ۳۰ کاشی در ۵ طرح متفاوت انجام شد، که نتایج در تمام موارد رضایتبخش بود.

³ برای هر ژن عدد تصادفی r تولید می‌شود و ترکیب خطی ژنهای والدین با ضرایب r و $1-r$ محاسبه می‌گردد.

کاشی نشان داده شده در شکل ۵-سمت چپ یک کاشی طرح‌دار بدون عیب است که برای خوشه‌بندی رنگهایش، از MiMaC استفاده شده است. MiMaC را بصورت تصادفی مقدار دهی اولیه کرده‌ایم. در شکل‌های ۳ و ۴ به ترتیب منحنی‌های تغییرات خطای هر خوشه و تغییرات پارامتر سرعت بهینه‌سازی هر خوشه نشان داده شده است. منحنی‌ها نشان می‌دهند که الگوریتم توانسته است طی تقریباً چهار مرحله بهینه‌سازی به پاسخ نهایی‌اش همگرا شود. همچنین از مقایسه‌ی منحنی‌های شکل ۳ و ۴ به روشنی می‌توان دید که تقریباً هر کجا که منحنی‌های خطا به طور یکنواخت کم شده‌اند منحنی‌های سرعت اوج گرفته‌اند و هر کجا منحنی‌های خطا نوسان داشته‌اند از سرعت بهینه‌سازی نیز کاسته شده است. برای خوشه‌بندی شبه بهینه کاشی مذکور، الگوریتم ژنتیکی پیشنهادی مورد استفاده قرار گرفت. در شبیه‌سازی، اندازه جمعیت ۳۰ و احتمال جهش ۰.۰۱ انتخاب گردید. منحنی‌های تغییرات مقدار تابع هزینه بهترین کروموزوم، بدترین کروموزوم، فرزند بوجود آمده در هر تولید نسل



شکل ۳: منحنی‌های بیشترین خطا و تغییرات نرخ بهینه‌سازی خوشه‌های تصویر شکل ۵-سمت چپ.



شکل ۴: (بالا-چپ) منحنی مقدار تابع هزینه بهترین کروموزوم هر نسل در طول فرآیند تکامل؛ (بالا-راست) منحنی مقدار تابع هزینه بدترین کروموزوم هر نسل در طول فرآیند تکامل؛ (پایین-چپ) منحنی مقدار تابع هزینه کروموزوم فرزند تولید شده در هر تولید نسل؛ (پایین-راست) منحنی تعداد تکرارهای MiMaC برای کروموزومهای فرزند تولید شده در هر تولید نسل.

و تعداد تکرارهای MiMaC برای فرزند مذکور قبل از همگرا شدن به ترتیب در شکل ۴ نشان داده شده‌اند. همانطور که می‌بینید الگوریتم ژنتیکی توانسته است خطای خوشه‌ها را کاهش دهد. همچنین منحنی مقدار هزینه بدترین کروموزوم جمعیت نشان می‌دهد که در طول فرآیند تکامل اعضای جمعیت همواره بهترین کروموزوم را دنبال نموده‌اند بطوری که با کاهش مقدار هزینه بدترین کروموزوم جمعیت، مقدار هزینه سایر کروموزومها نیز به تدریج کاهش می‌یابد. منحنی مقدار هزینه کروموزوم فرزند تولید شده در هر تولید نسل نیز این موضوع را تأیید می‌کند. با مشاهده منحنی تعداد تکرارهای الگوریتم MiMaC برای کروموزومهای فرزند بروشنی می‌توان دریافت که با پیشرفت فرآیند تکامل و بهبود کروموزومهای تولید شده، تعداد تکرارهای MiMaC کاهش یافته و در نتیجه حجم محاسباتی الگوریتم نسبت به شرایط آغاز کار کمتر شده است. همانطور که گفته شد، از الگوریتم پیشنهادی برای آشکارسازی عیوب رنگی ۳۰ کاشی در ۵ طرح متفاوت استفاده شد که در تمام موارد نتایج رضایتبخش و قابل توجه بود. در شکل‌های ۵، ۶ و ۷ به عنوان نمونه حاصل اعمال الگوریتم بر روی سه کاشی معیوب با طرحهای متفاوت نشان داده شده است. در هر شکل سه تصویر نشان داده شده است: تصویر سمت راست تصویر مرجع و بدون عیب است که با استفاده از آنها، خوشه‌های تصویر استخراج شده است. تصویر وسط کاشی معیوبی را که می‌خواهیم عیوب رنگی آن را آشکار کنیم نشان می‌دهد و در نهایت در تصویر سمت راست، عیوب آشکار شده با استفاده از اعمال خوشه‌ها بر تصویر وسط نشان داده شده است. همانطور که می‌بینید نتایج در حد قابل قبولی است.

۶- نتیجه‌گیری

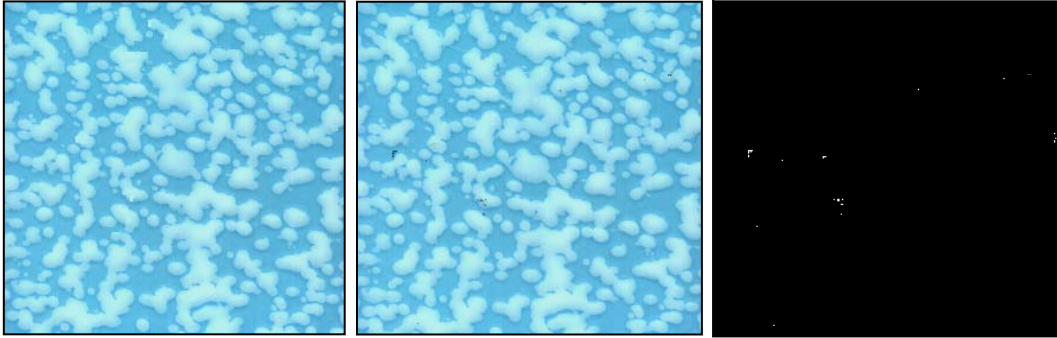
در این مقاله الگوریتمی جدید برای خوشه‌بندی رنگهای موجود در یک تصویر ارائه شده و از آن برای آشکارسازی عیوب رنگی کاشی‌ها استفاده شده است. MiMaC بر پایه‌ی کمینه کردن بیشترین خطای خوشه‌ها بنا نهاده شده است که لازمی مدل‌سازی خوشه‌ای است و این مهمترین مزیت آن نسبت به سایر هم‌تاهایش است. تأکید الگوریتم بر کمینه کردن بیشترین خطا می‌تواند سبب افزایش حساسیت آن بر نویز شود. از این رو در هنگام تصویر برداری از کاشی مرجع باید تا جای ممکن عوامل نویز و اعوجاج را حذف نمود. همچنین برای بهینه‌سازی پاسخهای MiMaC و مقدار دهی اولیه مناسب آن از یک الگوریتم ژنتیکی استفاده گردید. شبیه‌سازی‌های انجام شده نشان داد که الگوریتم ژنتیکی پیشنهادی در بهینه‌سازی پاسخهای MiMaC موفق عمل نموده است. علی‌رغم اینکه سرعت همگرایی MiMaC مناسب و کافی است اما الگوریتم ژنتیکی پیشنهادی به دلیل حجم زیاد محاسباتش از سرعت همگرایی کمی برخوردار است. البته با توجه به اینکه در مرحله استخراج ویژگی معمولاً از نظر زمانی محدودیت زیادی وجود ندارد بنابراین این کاستی نیز قابل چشم پوشی است. با وجود این، الگوریتم پیشنهادی پس از استخراج خوشه‌ها در مرحله‌ی بازرسی بسیار سریع و کارآمد است.

۷- مراجع

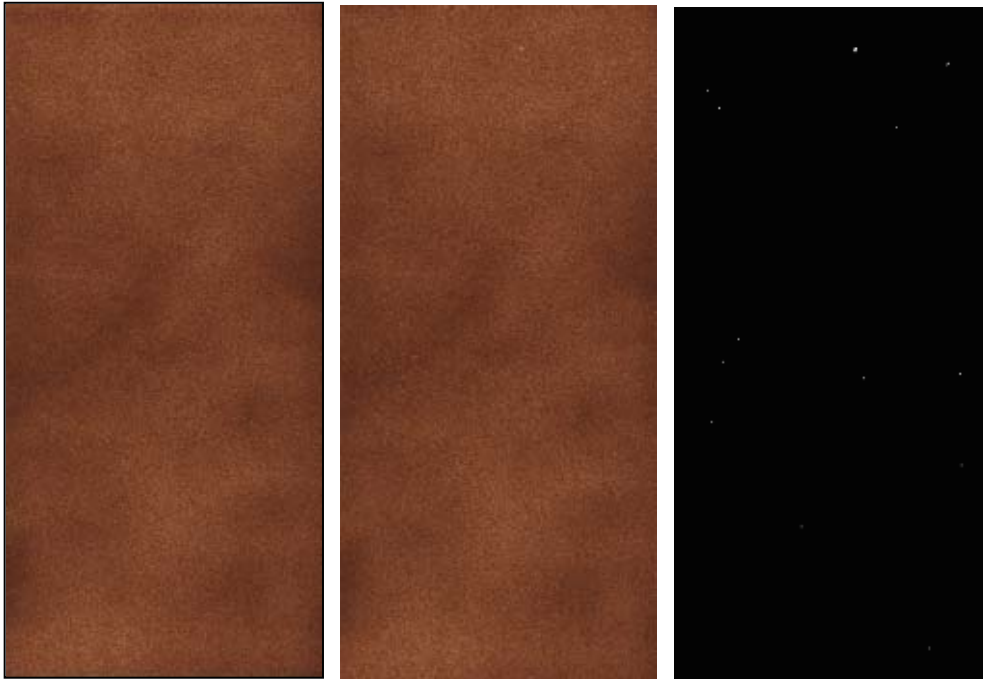
- [1] C. Boukouvalas, F. D. Natale, G. D. Toni, J. Kittler, R. Marik, M. Mirmehdi, M. Petrou, P. L. Roy, R. Salgari, G. Vernazza, "ASSIST: automatic system for surface inspection and sorting of tiles", *Journal of Material Processing Technology* 82, pp. 179-188, 1998.
- [2] C. Boukouvalas, J. Kittler, R. Marik, M. Mirmehdi and M. Petrou, "Ceramic Tile Inspection for Colour and Structural Defects", *Proceedings of AMPT95*, pp. 390-399, 1995. <http://www.compsci.bristol.ac.uk/Tools/Reports/Ps/mirmehdi-ampt95.ps.gz>
- [3] M. Mirmehdi, R. Marik, M. Petrou, J. Kittler, "Structural Fault Detection in Random Macro Textures", *Proceedings of Signal Processing VIII, Theories and Applications*, Vol. III, pp. 1499-1502, 1996. <http://www.compsci.bristol.ac.uk/Tools/Reports/Ps/mirmehdi-mmkp96.ps.gz>
- [4] A. N. Netravali, B. G. Haskell, "Digital Pictures, Representation and Compression", 2nd printing, Plenum Press, AT&T Bell Laboratories, 1989.
- [5] D. Bourgin, "Color Space FAQ", 1994, <ftp://turing.imag.fr/pub/compression/colorspace-faq>.
- [6] H. Masum, "Clustering Algorithms (an overview of clustering techniques, with potential applications)", <http://www.carleton.ca/~hmasum/clustering.html>
- [7] D. Fasulo, "An Analysis of Recent Work on Clustering Algorithms", Technical Report # 01-03-02, 1999.
- [8] K. Krishna, K. R. Ramakrishnan, M. A. L. Thathachar, "Vector Quantization using Genetic K-Means Algorithm for Image Compression", ICICS, 1997.
- [9] S. Haykin, *Neural Networks*, New York, Macmillan College Pub., 1994.
- [10] J. Y. F. Yam, T. W. S. Chow, "Extended Least Squares Based Algorithm for Training Feedforward Networks", *IEEE Trans on Neural Networks*, Vol.8, No.3, pp. 806-810, 1997.
- [11] J.S. Arora, O.A. Elwakeil, A.I. Chahnde, C.C. Hsieh, "Global optimization methods for engineering applications: a review", *Structural Optimization*, VOL. 9, p.p. 137-159, 1995.
- [12] D. Whitley, "A Genetic Algorithm Tutorial", *Statistics and Computer*, 4: 65-85, 1994.
- [13] J. Holland, "Adaptation in Natural and Artificial Systems", University of Michigan Press, 1975.
- [14] R. Samii, "Electromagnetic Optimization by Genetic Algorithms", Wiley – RsP, 1999.
- [15] D. E. Goldberg, "Genetic and Evolutionary Algorithms in the Real World", IlliGAL Report No. 99013,

March 1999.

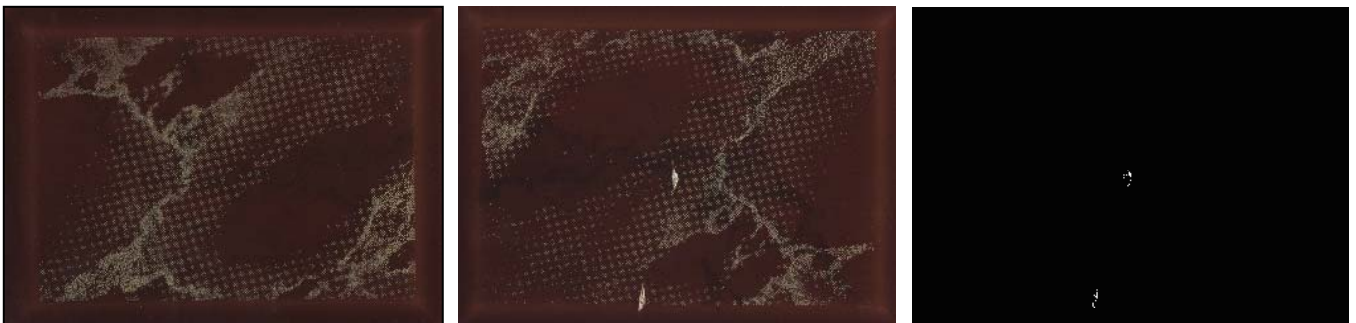
- [16] M. Mitchell, S. Forrest, "Genetic Algorithms and Artificial Life", Artificial Life, Vol. 1, No. 3, pp. 267-289, 1994. Reprinted in C. G. Langton (Ed.) Artificial Life: an Overview, MIT Press, Cambridge, MA (1995).
- [17] C. Darwin, "The Origin of Species by Means of Natural Selection", Murray, 1859.
- [18] D. Whitley, "The GENITOR Algorithm and Selective Pressure", Proc. 3th International Conference on Genetic Algorithms, ed. Morgan-Kaufmann: 116-121, 1989.



شکل ۵: (چپ) کاشی بدون عیب، (وسط) کاشی با عیوب رنگی، (راست) عیوب آشکار شده.



شکل ۶: (چپ) کاشی بدون عیب، (وسط) کاشی با عیوب رنگی، (راست) عیوب آشکار شده.



شکل ۷: (چپ) کاشی بدون عیب، (وسط) کاشی با عیوب رنگی، (راست) عیوب آشکار شده.