

انتقال میانگین سریع از طریق نمایش چگالی تراکم

چکیده

روش انتقال میانگین یکی از روش‌های خوشه‌بندی ثابت شده‌ای است که به صورت گسترده در کاربردهای تصویربرداری همچون قطعه‌بندی تصویر و ویدیو، نويزدایی، مسیریابی اشیاء، طبقه‌بندی بافت و غیره مورد استفاده قرار گرفته است. با این وجود روش انتقال میانگین دارای پیچیدگی زمانی نسبتاً بالایی است که در بسیاری از نقاط داده‌ای، ابرخطی می‌باشد. در این مقاله، روش انتقال میانگین سریع جدیدی ارائه می‌نماییم که بر نمونه‌برداری تصادفی برآورد چگالی کرنل (KDE) مبتنی می‌باشد. به صورت تئوری نشان می‌دهیم که KDE کاهش یافته حاصل، به KDE داده‌های کامل با دقت معینی نزدیک می‌باشد. به‌علاوه ثابت می‌کنیم که پیچیدگی زمانی روش انتقال میانگین سریع پیشنهادی نسبت به پیچیدگی زمانی روش انتقال میانگین اصلی به میزان بسیار قابل ملاحظه‌ای پایین‌تر می‌باشد؛ بهره‌نوعی برای مجموعه داده‌های بزرگ، چندین مرتبه می‌باشد. آزمایش‌های انجام شده نشان می‌دهند که نتایج قطعه‌بندی تصویر و ویدیو روش انتقال میانگین سریع پیشنهادی با نتایج قطعه‌بندی تصویر و ویدیو مبتنی بر روش انتقال میانگین استاندارد مشابه می‌باشد. همچنین کاربرد جدید روش انتقال میانگین سریع برای ساخت موثر سلسله‌مراتب گرافی برای تصاویر را نیز ارائه می‌نماییم؛ ساختار حاصل برای حل مسائل بینایی کامپیوتر که می‌توان آنها را همانند مسائل گراف شامل استریو، قطعه‌بندی نیمه‌خودکار و شار اپتیکی مطرح نمود بسیار سودمند می‌باشند.

1. مقدمه

برآورد چگالی کرنل و روش خوشه‌بندی انتقال میانگین از روش‌های بسیار مقبول در زمینه بینایی رایانه‌ای می‌باشند، به عنوان مثال منابع [5، 10] و مراجع ذکر شده در آنها را ملاحظه نمایید. از روش انتقال میانگین به صورت گسترده در کاربردهای تصویربرداری همچون قطعه‌بندی تصویر و ویدیو [5، 20]، نويززدایی [3]، مسیریابی اشیاء [7] و طبقه‌بندی بافت [11] و غیره استفاده شده است. به طور کلی روش انتقال میانگین از دو مرحله تشکیل شده است: (الف) ساخت چگالی احتمالی که توزیع نقاط مربوطه را در برخی از جاهای فضای ویژگی منعکس می‌نماید و (ب) نگاشت هر نقطه در مد (بیشینه) چگالی که به آن نزدیک می‌باشد.

یکی از مشکلات اصلی در بکارگیری روش انتقال میانگین مبتنی بر خوشه‌بندی در مجموعه داده‌های بزرگ، پیچیدگی محاسباتی آن است که در تعدادی از نقاط داده‌ای، ابرخطی می‌باشد. روش‌های در دسترس مختلفی برای افزایش سرعت روش انتقال مانگین وجود دارند. دمنتون و مگرت [8] از یک روش متوالی، مقیاس فضا مانند در انتقال میانگین با افزایش پهنای باند استفاده کرده‌اند. یانگ و همکارانش [23] برای افزایش سرعت کلی در توالی انتقال میانگین، از تبدیل گاوسی سریع بهره برده‌اند. گائو و همکارانش [13]، مجموع انتقال میانگین را به تعدادی از زیرمجموعه‌های موضعی تفکیک نموده‌اند. پاریس و دوراند [17] از تفکیک‌پذیری کرنل گاوسی چندبعدی برای اجرای d حلقه یک بعدی جداگانه استفاده نموده‌اند. وانگ و همکارانش [21] از ساختار داده زیرکی، درخت دوتایی، جهت افزایش سرعت انتقال میانگین استفاده کرده‌اند. همچنین مقاله ویوالدی و سواتو در مورد "انتقال سریع" [19] نیز در حدی به این مقوله مربوط می‌باشد.

در این مقاله روش جدیدی را معرفی نموده‌ایم که در آن مستقیماً با تشریح یا پیچیدگی فضای برآورد چگالی کرنل مواجه می‌باشیم که در تعدادی از نقاط داده‌ای خطی می‌باشند. تمرکز اصلی این مقاله بر روی روش انتقال میانگین سریع جدیدی است که بر محاسبه برآورد چگالی کوچک شده کرنل با استفاده از روش‌های نمونه‌برداری مبتنی می‌باشد. به صورت نظری نشان می‌دهیم که KDE کاهش یافته، به KDE داده‌های کامل با دقت موردنظر نزدیک می‌باشد. پیچیدگی زمانی روش انتقال میانگین سریع پیشنهادی مبتنی بر KDE کاهش یافته، به میزان قابل توجهی از پیچیدگی

زمانی روش انتقال میانگین اصلی کمتر می‌باشد؛ مقدار بهره نوعی در مورد مجموعه داده‌های بزرگ، چند برابر است. در تعدادی از آزمایش‌های انجام شده در خصوص قطعه‌بندی، وجود این بهره بزرگ را به صورت تجربی اثبات می‌نماییم. روش انجام فرایند ساخت KDE ارائه شده متراکم‌تر، ساده می‌باشد؛ به علاوه، روش جدید بر روش‌های موجود برای افزایش انتقال میانگین متعامد بوده و در اکثر حالات می‌توان آن را همراستا با این روش‌های قدیمی‌تر برای عملکرد بسیار بهتر به اجرا درآورد. شایان ذکر است با اینکه پژوهش‌های متعددی در خصوص (نمایش KDE متراکم) در گذشته انجام پذیرفته‌اند اما این پژوهش‌ها شدیداً بر روش‌های مبتنی بر شبکه‌های عصبی و نگاشت‌های خودآرا استوار هستند [18، 22، 12]. شبکه‌های عصبی به پیچیدگی اجرایی بالایی (همچنین مسائل دیگر) منجر می‌گردند که در این پژوهش تلاش نموده‌ایم از آن اجتناب نماییم.

بقیه قسمت‌های این مقاله به صورت زیر سازمان یافته‌اند. در بخش 2، در ابتدا به صورت خلاصه چارچوب KDE و الگوریتم انتقال میانگین برای یافتن مد موردنظر را مورد بررسی قرار داده‌ایم. سپس روش نمونه‌برداری برای ایجاد نمایشی متراکم‌تر از KDE ارائه نموده‌ایم. سپس براساس این KDE متراکم، روش انتقال میانگین سریع خود را تعریف و پیچیدگی محاسباتی آن را در مقایسه با انتقال میانگین استاندارد مورد تحلیل قرار داده‌ایم. همچنین قانونی برای انتخاب پهنای باند بهینه برای روش انتقال میانگین سریع ارائه نموده‌ایم. در بخش 3، عملکرد روش پیشنهادی در کارهای قطعه‌بندی تصویر و ویدیو را نشان داده و نتایج آن را با روش انتقال میانگین استاندارد مقایسه کرده‌ایم. در بخش 4، کاربرد نوین روش انتقال میانگین سریع برای ایجاد سلسه‌مراتب گراف چندمقیاسی برای تصاویر را ارائه نموده‌ایم. بخش 5 نیز به نتیجه‌گیری اختصاص داده شده است.

2. انتقال میانگین سریع

در این بخش، الگوریتم انتقال میانگین سریع را ارائه می‌نماییم. پس از مروری مختصر بر الگوریتم انتقال میانگین استاندارد، نمایش‌های متراکم موجود برای برآورد چگالی کرنل را به صورت کلی مورد بحث قرار داده و ثابت می‌نماییم که الگوی مبتنی بر نمونه‌برداری این نمایش را برای ما به ارمغان خواهد آورد. پس از بدست آوردن این نمایش فشرده،

روش سریعی برای انتقال میانگین پیشنهاد نموده و نشان می‌دهیم که پیچیدگی آن از پیچیدگی مربوط به روش انتقال میانگین استاندارد بسیار کمتر می‌باشد. سپس گونه‌ای معتدل از خوشه‌بندی این الگوریتم را مورد بررسی قرار داده و بحث خود را با انتخاب بهینه پهنای باند به اتمام می‌رسانیم.

2.1. مروری بر روش انتقال میانگین

در این بخش، روش انتقال میانگین عادی را مورد بررسی قرار می‌دهیم. به صورت خلاصه، این کار را با تعریف برآورد چگالی کرنل (KDE) آغاز می‌نماییم. داده‌های در دسترس ما، مجموعه‌ای از نقاط $\{x_i\}_{i=1}^n$ هستند که گاهی اوقات به آنها بردارهای ویژگی اطلاق شده و در فضای اقلیدسی موجودیت دارند: $x_i \in \mathbb{R}^d$. در کاربردهای بینایی کامپیوتری، معمولاً یک چنین برداری به ازای هر پیکسل (یا وکسل در حالت سه بعدی) وجود دارد؛ این احتمال وجود دارد که این بردار رنگ، رنگی که موقعیت به آن اضافه شده است، بافت و غیره باشد. بنابراین برآورد چگالی کرنل (KDE) این داده‌ها به صورت زیر می‌باشند

$$f(x) = \frac{1}{n} \sum_{i=1}^n K_{\mathbf{H}}(x - x_i)$$

این رابطه، برآوری از چگالی احتمال در بردارنده نقاط در جایی است که در آن تابع $K_{\mathbf{H}}(x - x_i)$ اساساً یک برآمدگی متمرکز در x_i می‌باشد. به صورت خاص $K_{\mathbf{H}}(z) = |\mathbf{H}|^{-1/2} K(\mathbf{H}^{-1/2} z)$ را در نظر می‌گیریم که در آن کرنل K خود چگالی احتمال با میانگین صفر، کوواریانس شناسه بوده و $\lim_{\|x\| \rightarrow \infty} \|x\|^d K(x) = 0$ را ارضا می‌نماید. گزینه‌های متداول برای K عبارتند از: کرنل‌های گاوسی، یکنواخت و اپانچنیکوف (چندبعدی). در بسیاری از حالت‌های موردنظر، \mathbf{H} قطری خواهد بود. به طور کلی این حالت ضروری نمی‌باشد.

الگوریتم انتقال میانگین اساساً یک الگوریتم تپه‌نوردی است؛ یعنی الگوریتم انتقال میانگین که از هر نقطه x شروع می‌شود، یک الگوی توالی ضروری است که را به بالای تپه KDE برده و در نهایت در بیشینه (یا مد) موضعی KDE متوقف می‌شود که حوزه مطلوب x در آن واقع شده است. برای تعیین توالی انتقال میانگین، شکل کرنل را به صورت

صریح ساده نموده و شکل تقارن شعاعی $K(x) = ck(\|z\|^2)$ را بدست می‌آوریم که k در آن، پروفایلی یک بعدی همانند پروفایل‌های گاوسین یک بعدی، یکنواخت یا اپانچنیکوف بوده و C یک بهنجارش می‌باشد. با قرار دادن علامت $g = k'$ ، توالی انتقال میانگین به صورت زیر در می‌آید

$$x \leftarrow \frac{\sum_{i=1}^n x_i g \left(\left\| \frac{x-x_i}{h} \right\|^2 \right)}{\sum_{i=1}^n g \left(\left\| \frac{x-x_i}{h} \right\|^2 \right)} \equiv M(x) \quad (1)$$

برای آوردن x به مدی که حوزه مطلوب در آن واقع می‌باشد، فرایند تکرار تعدادی نامتناهی از زمان‌ها تضمین شده است. مزیت این روش بر روش صعود گرادینانی عادی، عدم نیاز به مجموعه پارامتر گام زمانی است (شایان ذکر است که چنین پارامتری در عبارت بالا وجود ندارد)؛ از لحاظ عملی، انتقال میانگین در تعداد گام‌های بسیار کوچک که معمولاً در حدود 5 می‌باشد، همگرا می‌گردد.

برای استفاده از الگوریتم انتقال میانگین برای قطعه‌بندی یا خوشه‌بندی در مجموعه $\{x_i\}_{i=1}^n$ می‌توان فرایند تکرار را از هر نقطه داده آغاز نمود. با قرار دادن $M^1(x) = M(x)$ ، $M^2(x) = M(M(x))$ و غیره، هر نقطه x_i را در $M^\infty(x_i)$ می‌نگاریم (خاطر نشان می‌نماییم که معمولاً $M^5(x_i)$ این کار را انجام خواهد داد). از آنجا که تعداد مدها از تعداد نقاط بسیار کمتر می‌باشند بنابراین شکلی از قطعه‌بندی یا خوشه‌بندی وجود دارد که در عمل در تعدادی از کاربردها بسیار خوب عمل می‌نمایند [20, 6, 14].

2.2 KDE متراکم‌تر: مسئله

بدون محاسبه صریح پیچیدگی خوشه‌بندی انتقال میانگین (بحث در خصوص این تمرین را به بخش 2.5 موكول می‌نماییم)، خاطر نشان می‌نماییم که زمان اجرا در n ، تعداد نقاط داده‌ای ابرخطی خواهد بود. اگر با تصاویر بزرگی که از دوربین‌های روزانه حاصل می‌شوند سر و کار داشته باشیم آنگاه n به سادگی در محدوده 10^7 می‌باشند. اما موضوع ناگوار آن است که گاهی اوقات علاقمندیم تا از روش انتقال میانگین برای ویدیوها و یا در تصویرسازی سه بعدی همچون تصاویر CT یا MRI استفاده نماییم؛ در این حالت، n از مرتبه 10^8 یا بالاتر نیز به سادگی قابل انتظار خواهد

بود. بنابراین بهتر آن است که روش انتقال میانگین سریع استفاده نماییم. یکی از تنگناهای اصلی فراروی سرعت، پیچیدگی توصیف KDE است که $O(n)$ می‌باشد؛ بنابراین با بررسی مسئله یافتن توصیفی متراکم‌تر برای KDE، کار خود را آغاز می‌نماییم.

می‌خواهیم KDE نزدیک به $f(x)$ را بیابیم که از نقاط بسیار کمتری برای ایجاد آن استفاده می‌نماید. یعنی درصد حل مسئله زیر می‌باشیم:

$$\min_{\{\hat{x}_j\}_{j=1}^m, \hat{H}} D(\hat{f}(\cdot), f(\cdot)) \text{ subject to } \hat{f}(x) = \frac{1}{m} \sum_{j=1}^m K_{\hat{H}}(x - \hat{x}_j)$$

که D در آن معیار فاصله بین چگالی‌های احتمال و m تعداد نقاط ثابت با $n \ll m$ می‌باشند. این روش بهینه‌سازی در جستجوی یافتن نقاط داده‌ای \hat{x} متضمن KDE \hat{f} است که به خوبی KDE اصلی را تقریب زده اما از تعداد نقاط بسیار کمتری استفاده می‌نماید.

گزینه‌های طبیعی مختلفی برای مقیاس فاصله D ، از فواصل L_p گونه گرفته تا مقیاس‌های نظری با اطلاعات بیشتری همچون واگرایی کولبک-لیبلر وجود دارند. در تمامی این حالت‌ها، به طور کلی حل مسئله بهینه‌سازی حاصل کار دشواری است؛ این موضوع ناشی از این حقیقت است که \hat{x} در کرنل K ظاهر می‌گردد که به بهینه‌سازی غیرمحدب منجر شده و راه‌حل‌های کلی موثری را نمی‌توان برای آن یافت. با این وجود f و \hat{f} چیزی بیشتر از تابع می‌باشند؛ هر دو آنها چگالی بوده و این حقیقت، ابزاری برای یافتن راه‌حلی موثر جهت مسئله نمایش تراکم را برای ما فراهم می‌آورد.

3.2 KDE متراکم‌تر از طریق نمونه‌برداری

راه‌حل ارائه شده ما بسیار ساده می‌باشد: با نمونه‌برداری از توزیع داده شده با $f(\cdot)$ ، نمونه‌های موردنظر \hat{x} خود را انتخاب می‌نماییم. قبل از پرداختن به منطق این روش، خاطرنشان می‌نماییم که این نمونه‌برداری به صورت کلی امکان‌پذیر بوده و می‌توان آن را به عنوان یک روش سه مرحله‌ای به اجرا درآورد:

قضیه 1. برای هر $j = 1, \dots, m$ ، فرض می‌کنیم که \hat{x}_j به صورت زیر ساخته می‌شود:

1. عدد صحیح $r_j \in \{1, \dots, n\}$ را به صورت تصادفی انتخاب می‌نماییم.

2. نمونه تصادفی δ_j را از $K(\cdot)$ انتخاب می‌کنیم.

3. $\hat{x}_j = x_{r_j} + H^{1/2} \delta_j$ را قرار می‌دهیم.

بنابراین \hat{x}_j ، نمونه مناسبی از f می‌باشد.

اثبات: مرجع [9] را ملاحظه نمایید.

مادامی که می‌توان نمونه را به سادگی از $K(\cdot)$ ساده‌سازی نمود (همانند آنچه که در حالت‌های موجود برای کرنل‌های گاوسین، یکنواخت و اپانچنیکوف بود)، روش مورد نظر بسیار ساده خواهد بود.

اکنون، سؤال اصلی این است: اگر \hat{f} KDE مبتنی بر نمونه‌های تصادفی \hat{x} را بسازیم آیا به KDE واقعی نزدیک خواهند بود یا خیر؟ البته \hat{f} KDE به خودی خود، متغیری تصادفی است (یعنی یک تابع تصادفی) و بنابراین هر نتیجه‌ای که به دنبال اثبات آن باشیم، ماهیتی احتمالی خواهند داشت. در حقیقت نتیجه زیر را داریم که مطابق انتظار، نزدیک بودن f و \hat{f} در یک حس یا مفهوم L_2 را تضمین می‌نماید:

قضیه 2. همانند آنچه که در بالا بود f ، KDE با n نقطه و \hat{f} ، KDE ایجاد شده با روش نمونه‌برداری m بار از f را در نظر گرفته و ماتریس پهنای باند قطری $\hat{H} = \hat{h}^2 \mathbf{I}$ را مدنظر قرار می‌دهیم. فاصله مربعی چشم‌انتظاری L_2 بین دو

چگالی را در رابطه $J = E \left[\int (f(x) - \hat{f}(x))^2 dx \right]$ در نظر می‌گیریم. بنابراین

$$J \leq 4A\hat{h} + A^2\hat{h}^2V + \frac{B}{m\hat{h}^d} + \frac{ABV}{m\hat{h}^{d-1}} \quad (2)$$

که در آن A ، B و V ثابت‌هایی هستند که به \hat{h} یا m بستگی ندارند.

اثبات: مرجع [9] را ملاحظه نمایید.

معنی این قضیه بسیار سراسر است: دو KDE، f و \hat{f} (مطابق انتظار) نزدیک خواهند بود اگر m به اندازه کافی بزرگ بوده و پهنای باند \hat{h} به صورت مناسب به عنوان تابعی از m انتخاب شده باشد. این دقیقاً همان چیزی است که در

نمایش تراکم به دنبال آن می‌باشیم: پیچیدگی توصیف KDE از $O(n)$ به $O(m)$ کاهش یافته است اما در حس L_2 مورد انتظار، مقداری نزدیک به همان چگالی ایجاد نموده‌ایم.

به مسئله انتخاب پهنای باند در بخش 2.7 برمی‌گردیم؛ در بخش بعدی، به مسئله حساس نحوه استفاده از نمایش KDE کاهیده خود در خوشه‌بندی انتقال میانگین باز خواهیم گشت.

2.4. انتقال میانگین سریع

چگونه می‌توان KDE با تراکم بیشتر خود را الگوریتم خوشه‌بندی انتقال میانگین گنجانند؟ از روش سه مرحله‌ای زیر استفاده می‌کنیم:

1. نمونه برداری: m نمونه از چگالی f را در نظر می‌گیریم تا $\{\hat{x}_j\}_{j=1}^m$ را بدست آوریم. چگالی جدید $\hat{f}(x) = \sum_{j=1}^m K_{\hat{h}}(x, \hat{x}_j)$ را تشکیل می‌دهیم.

2. انتقال میانگین: روش انتقال میانگین را بر روی هر m نمونه به اجرا درمی‌آوریم: $\hat{x}_j \rightarrow \hat{M}^\infty(\hat{x}_j)$. در اینجا، \hat{M} حاکی از آن است که از \hat{f} (به جای f) برای انتقال میانگین استفاده نموده‌ایم.

3. نگاهت رو به عقب: برای هر x_i ، نزدیک‌ترین نمونه جدید \hat{x}_j^* را می‌یابیم. بنابراین $x_i \rightarrow \hat{M}^\infty(\hat{x}_j^*)$.

در مرحله 1، KDE کاهیده را ساخته و در مرحله 2، انتقال میانگین را بر روی این KDE کوچکتر به اجرا درمی‌آوریم. با تعیین این مدها در KDE کاهیده، مسئله موجود همانا نگاهت رو به عقب به سمت داده‌های اصلی است. در مرحله 3، با نگاهت از هر نقطه در داده‌های اصلی (x_i) به نزدیک‌ترین نقطه در داده‌های کاهیده (\hat{x}_j^*) و از آنجا به مدی که در آن نقطه جریان می‌یابد $(\hat{M}^\infty(\hat{x}_j^*))$ ، مواجه می‌باشیم.

افزایش سرعت اصلی در مرحله دوم رخ می‌دهد؛ به جای استفاده از تمامی m نمونه جهت محاسبه انتقال میانگین می‌توان از مجموعه کاهش یافته‌ای از m نمونه استفاده نمود. در بخش بعدی، افزایش سرعت نظری دقت که این روش در پی دارد را به صورت کیفی مورد بررسی قرار خواهیم داد؛ در اینجا خاطر نشان می‌شود که اجرای فرایند به

صورت ساده می‌تواند به افزایش سرعت n/m منتهی می‌شود که در عمل می‌تواند از مرتبه 100 بزرگتر و حتی خیلی بیشتر از آن نیز برود.

2.5. تحلیل پیچیدگی

جنبه کلیدی موجود در محاسبه پیچیدگی، سرعت جستجوی نزدیک‌ترین همسایه است. شایان ذکر است که در تکرار روش انتقال میانگین، معادله (1) را ملاحظه نمایید، باید نزدیک‌ترین همسایه‌های n نمونه x_i به نقطه موردنظر، x را محاسبه نمود. فرض کنید که ساختار داده‌ای را داریم که امکان جستجوی نزدیک‌ترین همسایه در زمان $q(n)$ را داده و به زمان پردازش $p(n)$ جهت محاسبه نیازمند می‌باشد؛ به طور خلاصه به نمونه‌هایی از این ساختارها خواهیم پرداخت اما در اینجا فعلاً آنها را باقی خواهیم گذاشت. در این حالت، هر تکرار فرایند انتقال میانگین به زمان $O(q(n))$ جهت محاسبه نیاز دارد؛ و فرض می‌کنیم (که در عمل هم صادق می‌باشد) که تکرارهای $O(1)$ برای همگرایی لازم می‌باشند، بنابراین هزینه اجرای الگوریتم بر روی تمامی n نقطه داده (یعنی الگوریتم کاهش کلی داده‌ها) به صورت زیر خواهد بود

$$T_{orig}(n) = O(p(n) + nq(n))$$

الگوریتم پیشنهادی ما چقدر سریع‌تر است؟ به بخش 2.4 برگشته و پیچیدگی مربوط به هر مرحله را تفکیک می‌کنیم. مرحله 1، مرحله نمونه‌برداری، $O(m)$ است. قبلاً پیچیدگی مرحله 2، مرحله انتقال میانگین، را محاسبه نموده‌ایم؛ که دارای عبارت ساده بالا است اما این کار را به جای n نمونه برای m نمونه انجام می‌دهیم یعنی $O(p(m) + mq(m))$. در مرحله 3، باید نگاشت رو به عقب را به اجرا درآوریم؛ یعنی برای هر n نمونه اصلی باید نزدیک‌ترین نقاط موجود m نمونه جدید را بیابیم. برای استفاده از ساختار داده‌های خود (که زمان پیش‌پردازش را قبلاً برای آن در مرحله 2 به حساب آورده‌ایم)، به $O(nq(m))$ نیاز داریم. بنابراین مجموع موردنظر به صورت زیر می‌باشد

$$\begin{aligned} T_{reduce}(n, m) &= O(m + p(m) + mq(m) + nq(m)) \\ &= O(p(m) + nq(m)) \end{aligned}$$

با مقایسه این رابطه با عبارت موجود برای $T_{orig}(n)$ ، و از آنجایی که $n = \Omega(m)$ می‌باشد، پیچیدگی آن را به صراحت کاهش داده‌ایم. از میزان وابستگی به جزئیات دقیق ساختار داده‌ها استفاده نموده و با این مسئله به مقابله می‌پردازیم.

در ساده‌ترین حالت، هیچ ساختار داده‌ای خاصی نداریم. بنابراین $p(n) = 0$ بوده و زمان جستجو در تعدادی از المان‌های $q(n) = O(n)$ خطی می‌باشد. در این حالت، $T_{orig}(n) = O(n^2)$ می‌باشد در حالی که $T_{reduce}(m, n) = O(nm)$ است بنابراین افزایش سرعت ضربی از n/m می‌باشد. در حالت‌های عملی موردنظر، $n/m > 100$ بوده و بنابراین این افزایش سرعت کاملاً یک افزایش سرعت موثر می‌باشد.

اکنون می‌توان از ساختارهای داده‌ای پیچیده‌تر برای جستجو استفاده نمود. زمانی که ابعادی که داده‌ها در آن ابعاد وجود دارند $d > 3$ یا از این دست باشند، نمودارهای ورونوی چندان سودمند نخواهند بود چون زمان پیش‌پردازش $p(n) = O(n^{\lceil d/2 \rceil})$ است. سایر ساختارهای داده‌ای متداول همچون درخت‌های kd نیز دارای پیچیدگی فضایی نامایی در d می‌باشند [2]. دسته‌ای از الگوریتم‌های نزدیک‌ترین همسایه تقریبی همچون الگوریتم‌های مبتنی بر درهم‌سازی حساس به مکان [2] وجود دارند که اگر در پی یافتن همسایه‌های نزدیک و نه نزدیک‌ترین همسایه‌ها باشیم به جستجوهای موثرتری منجر می‌شوند. در حقیقت اگر ضریب تقریبی $c \geq 1$ را برگزینیم - یعنی نقاطی را می‌یابیم که فاصله آنها از قسمت مورد جستجو درون بازه‌ای از ضریب c نزدیک‌ترین همسایه می‌باشد - آنگاه $T_{orig}(n) = O(n^{1+1/c^2})$ و $T_{reduce}(m, n) = O(nm^{1/c^2})$ را خواهیم داشت که به افزایش سرعت $O(n/m^{1/c^2})$ منتهی می‌گردد، به عنوان مثال مرجع [2] را ملاحظه نمایید. معمولاً $c = 1.5$ را در نظر می‌گیریم که اگر $n/m = 100$ باشد آنگاه به افزایش سرعتی در حدود 10 منجر می‌شود. (با این وجود خاطر نشان می‌نماییم که علی‌رغم دشوار نبودن نسبی اجرای این ساختارهای داده‌ای در عمل، اما این ساختارها اغلب دارای ویژگی‌های نظری عالی می‌باشند، مرجع [2] را مشاهده نمایید).

2.6. متغیر: قطعه‌بندی نرم یا کارتونها

متغیر زیر را برای الگوریتم خوشه‌بندی انتقال میانگین سریع پیشنهاد می‌نماییم که به گونه‌ای موثر به قطعه‌بندی نرم منجر می‌گردد. شایان ذکر است که تنها مرحله سوم تغییر یافته است؛ برای روشن‌سازی بیشتر، دو مرحله اول را مجدداً ایجاد می‌نماییم.

1. نمونه‌برداری: m نمونه از چگالی f را در نظر می‌گیریم تا $\{\hat{x}_j\}_{j=1}^m$ را بدست آوریم. چگالی جدید $\hat{f}(x) = \sum_{j=1}^m K_{\hat{h}}(x, \hat{x}_j)$ را تشکیل می‌دهیم.

2. انتقال میانگین: روش انتقال میانگین را بر روی هر m نمونه به اجرا درمی‌آوریم: $\hat{x}_j \rightarrow \hat{M}^\infty(\hat{x}_j)$ در اینجا، \hat{M} حاکی از آن است که از \hat{f} (به جای f) برای انتقال میانگین استفاده نموده‌ایم.

3. نگاهت رو به عقب وزنی: برای هر x_i و هر \hat{x}_j ، وزن بین آنها را براساس $\omega_{ij} \propto K_{\hat{h}}(x_i, \hat{x}_j)$ محاسبه می‌نماییم به طوری که مجموع وزن‌ها برابر با 1 باشد: بنابراین $x_i \rightarrow \sum_{j=1}^m \omega_{ij} \hat{M}^\infty(\hat{x}_j)$.

این روش به جای قطعه‌بندی ثابت تکه‌ای، قطعه‌بندی هموار یا کارتونها مانند را نتیجه خواهد داد. این روش با الگوی مامفورد-شاه [16] مشابه است که در جستجوی تقریب هموار تکه‌ای برای تصویر اصلی می‌باشد. نمونه‌هایی از هر دو قطعه‌بندی نرم و سخت را در بخش 3 نشان خواهیم داد.

2.7. پهنای باند بهینه

اکنون به عبارت موجود برای فاصله L_2 بین f KDE و \hat{f} KDE کاهیده که در عبارت (2) داده شده بود، بازمی‌گردیم. فرض کنید که ما به حالت مجانبی می‌نگریم که در آن m به اندازه کافی بزرگ بوده و \hat{h} به اندازه کافی کوچک می‌باشد؛ در این حالت، حد بالایی مربوط به J به خوبی به وسیله رابطه زیر تقریب زده شده است

$$L = 4A\hat{h} + \frac{B}{m\hat{h}^d}$$

اکنون تلاش می‌کنیم تا پهنای باندهای را بیابیم که با ثابت نگه داشتن m ، L را به حداقل مقدار خود برساند؛ \hat{h} در L محدب بوده و با برابر قرار دادن مشتق آن با صفر، به رابطه زیر می‌رسیم

$$\hat{h}^*(m) = C_1 m^{-\frac{1}{\alpha+1}} \quad \text{and} \quad L^*(m) = C_2 m^{-\frac{1}{\alpha+1}}$$

که C_1 و C_2 در آن ثابت‌های (نه چندان جالب توجه) می‌باشند. با داشتن پهنای باند اصلی h ، روش ساده انتخاب \hat{h} برای KDE کاهیده را در اختیار خواهیم داشت؛ با حذف C_1 به رابطه زیر خواهیم رسید

$$\hat{h} = (n/m)^{\frac{1}{\alpha+1}} h$$

3. آزمون‌های قطعه‌بندی

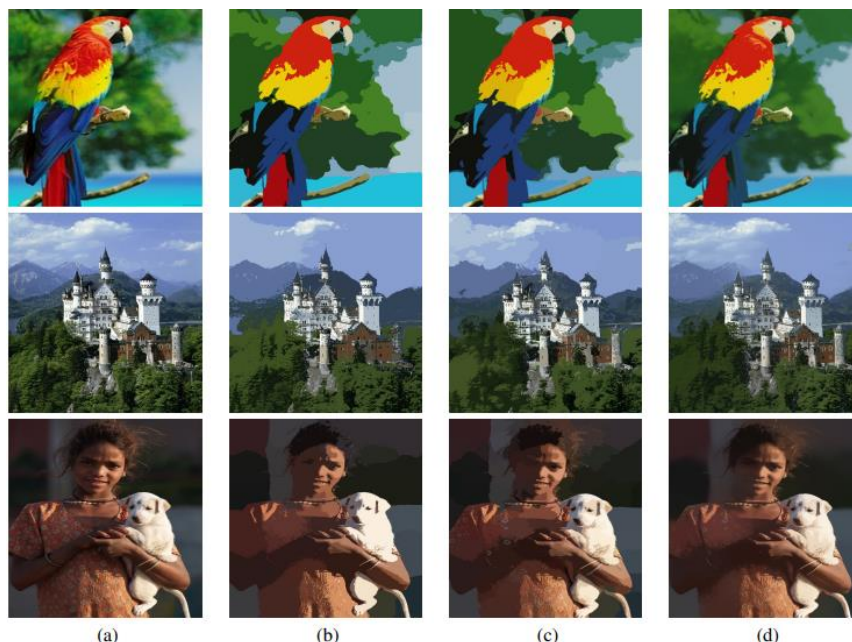
3.1. قطعه‌بندی تصویر

در آزمون قطعه‌بندی تصویر، عملکرد روش انتقال میانگین سریع پیشنهادی را با روش انتقال میانگین استاندارد در مورد سه تصویر با هم مقایسه می‌کنیم. بردارهای ویژگی 5 بعدی در نظر گرفته شده و از سه بردار رنگ Lab پیکسل‌های منفرد در تصویر و دو مختصه فضایی مربوطه (x و y) ساخته شده‌اند. در جدول 1، اطلاعات مربوط به زمان‌بندی و خود تصاویر در شکل 1 نشان داده شده‌اند. در هر مثال، ضریب نمونه‌برداری برابر $\frac{n}{m} = 1024$ گذاشته شده است. تحلیل پیچیدگی انجام شده ما نشان می‌دهد که باید انتظار داشتن ضریب تسریعی در حدود 1024 را داشته باشیم؛ در حقیقت علی‌رغم استفاده از جستجوهای سریع نزدیک‌ترین همسایه‌ها (براساس درخت kd که علی‌رغم بحث‌های موجود در خصوص آن، در ابعاد پایین به خوبی عمل می‌نماید)، ضریب تسریعی در دو حالت از سه حالت موجود از این ضریب نمونه‌برداری تجاوز می‌نماید که در جدول 1 نشان داده شده است.

	Parrot	Castle	Girl
Image Size	844 × 636	960 × 784	551 × 735
Sampling m/n	1024	1024	1024
Speed-up Factor	401	1160	1543

جدول 1. نتایج زمان‌بندی برای قطعه‌بندی تصویر

در شکل 1، نتایج روش انتقال میانگین استاندارد (ستون دوم)، روش انتقال میانگین سریع (ستون سوم) و متغیر قطعه‌بندی نرم روش انتقال میانگین سریع (ستون چهارم) را نشان داده‌ایم. بدیهی است که نتایج روش قطعه‌بندی سخت بدست آمده با استفاده از روش‌های انتقال میانگین اصلی و سریع کاملاً با همدیگر همسان به نظر می‌رسند. مطابق انتظار روش قطعه‌بندی نرم، نوعی کارتون هموار تکه‌ای از تصویر اصلی با جزئیات کوچکی است که از تصویر اصلی برداشته شده است. بنابراین برای کاربردهای کاهش نویز می‌تواند مناسب باشد.

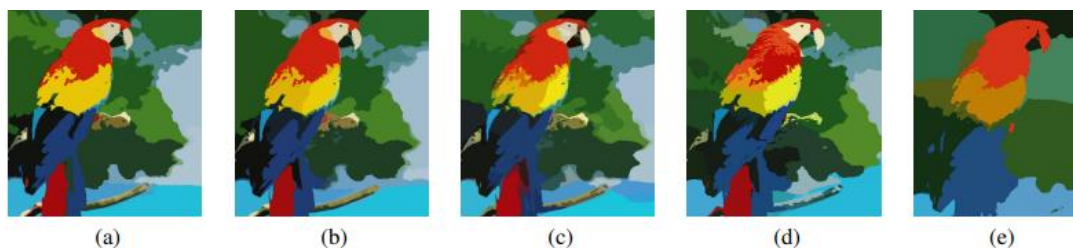


شکل 1. مثالی از قطعه‌بندی تصویر: (الف) تصویر اولیه، (ب) نتیجه قطعه‌بندی انتقال میانگین استاندارد، (ج) نتیجه قطعه‌بندی انتقال میانگین سریع، (د) گونه قطعه‌بندی نرم انتقال میانگین نرم. به نتایج مشابه قطعه‌بندی سخت در (ب) و (ج) توجه کنید؛ روش پیشنهادی از روش انتقال میانگین اصلی، 1000 برابر سریع‌تر می‌باشد.

3.2. تغییر ضریب نمونه‌برداری n/m

برای درک تاثیر تغییر ضریب نمونه‌برداری بر قطعه‌بندی، الگوریتم انتقال میانگین سریع با ضریب نمونه‌برداری مختلف n/m را در مورد تصویر طوطی بخش قبلی مورد استفاده قرار داده‌ایم. تصاویر در شکل 2 نشان داده شده‌اند. قابل ذکر است که الگوریتم در مواردی که ضریب نمونه‌برداری تا $n/m = 4096$ افزایش می‌یابد نیز به خوبی عمل می‌نماید

که این مورد در شکل (د) نشان داده شده است؛ از آنجایی که خود تصویر حاوی کمی بیش از 5×10^5 پیکسل است که با بکارگیری تنها $m = 131$ نمونه انتخاب شده به صورت تصادفی برای ساخت KDE متناظر می‌باشد که بسیلر قابل ملاحظه است. الگوریتم سریع در $n/m = 16384$ با مشکل مواجه می‌شود؛ که البته چندان شگفت‌آور نیست چون این حالت با استفاده از تنها 32 نمونه انتخاب شده به صورت تصادفی جهت ساخت KDE متناظر می‌باشد. در حقیقت در این نرخ نمونه‌برداری فرعی بالا، الگوریتم قطعاتی کاملاً متفاوت از آنچه که در ادامه اجرا می‌شود را تولید می‌نماید.



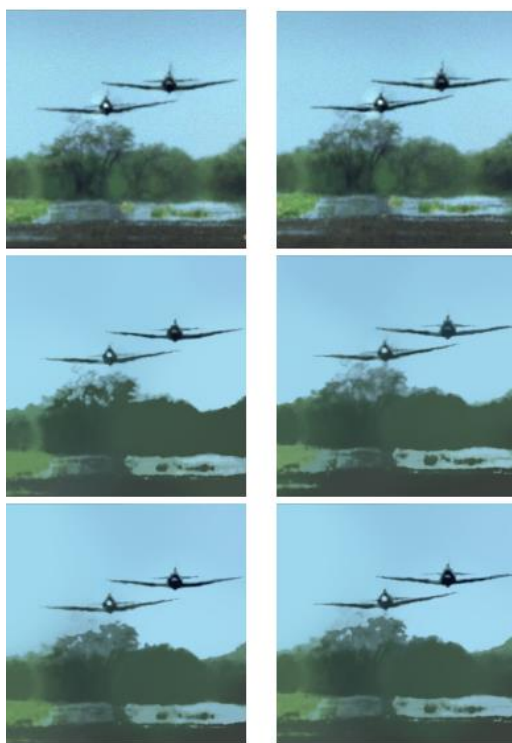
شکل 2. تاثیر تغییر شریب نمونه‌برداری n/m . (الف) $n/m = 64$ (ب) $n/m = 256$ (ج) $n/m =$

1024، (د) $n/m = 4096$ (ه) $n/m = 16384$

3.3. قطعه‌بندی ویدیو

در شکل 3، نتایج قطعه‌بندی ویدیو در یک تکرار 10 فریمی برای دو فریم متوالی از زنجیره ویدیو نشان داده شده است (ردیف بالا را مشاهده کنید). در این آزمون، استفاده از روش انتقال میانگین سریع پیشنهادی در دو آرایش را با همدیگر مقایسه نموده‌ایم. در نخستین آرایش (ردیف میانی)، قطعه بندی فریم به فریم انجام شده است. در آرایش دوم، داده‌های حاصل از پنجره زمانی 10 فریمی در مجموعه داده‌های منفرد بزرگی ترکیب شده و برای اجرای دو مرحله نخست (نمونه‌برداری و انتقال میانگین) در الگوریتم بخش 2.4، با استفاده از ضریب نمونه‌برداری جنبی 1024، مورد استفاده قرار گرفته است. (شایان ذکر است که در این مثال، امکان استفاده از ضریب نمونه‌برداری جنبی 5000 نیز وجود دارد که نتایج حاصل از آن تا حدودی دارای شفافیت کمتری بوده و نقص بیشتری دارند.) بنابراین مرحله سوم (نگاشت رو به عقب) لز بخش 2.4 در مورد هر فریم بکار گرفته شده است. نتایج نسخه "پنجره‌دار (محدود شده: Windowed)"،

واضح تر به نظر می‌رسد؛ به عنوان مثال این نسخه همانند روش فریم به فریم، رنگ هواپیما را نامناسب طبقه‌بندی ننموده است. گزینه دیگر همانا استفاده از محور زمانی به عنوان محور ویژگی اضافی است که در کاربردهایی همچون مسیریابی هدف سودمند می‌باشد. بدیهی است که روش سریع پیشنهادی، احتمالات جدیدی را برای مواجهه با مجموعه داده‌های فضایی- زمانی بزرگ را فراروی ما قرار می‌دهد.



شکل 3. مثالی از قطعه‌بندی ویدیو. ردیف بالا: دو فریم اصلی از رشته ویدیو. ردیف میانی: نتیجه قطعه‌بندی فریم به فریم با استفاده از روش پیشنهادی. ردیف پایین: نتیجه قطعه‌بندی مبتنی بر پنجره 10 فریمی با استفاده از روش پیشنهادی.

4. کاربردها: سلسله مراتب گراف

در این بخش، نحوه استفاده از الگوریتم انتقال میانگین سریع جهت ایجاد سلسله مرتبه گراف‌های متناظر با یک تصویر خاص نشان داده شده است. به دلیل درگیر بودن با تعدادی از مسائل بینایی رایانه‌ای همچون مسائل گراف و حل آنها با استفاده از الگوریتم‌های گرافی همچون برش‌های گراف [15]، داشتن ساختار چندمقیاسی در گراف‌های تصویر

می‌تواند بسیار سودمند باشد. مثال‌هایی از این مسائل عبارتند از استریو، قطعه‌بندی نیمه‌خودکار و جریان اپتیکی و همچنین مسائل متعددی که به صورت بیشینه برآورد استقرایی بر روی میدان‌های تصادفی مارکوف با آنها مواجه هستیم. برای کمک به فرایند تسریع راه‌حل در این مسائل در روش چندمقیاسی معمول می‌توان از سلسله مراتب گراف چندمقیاسی استفاده نمود: مسئله موردنظر نخست در مقیاس درشت‌تر و دانه‌ای حل شده است؛ سپس راه‌حل این مسئله برای شروع افزایش مقیاس بعدی مورد استفاده قرار گرفته است؛ و غیره. این روش علاوه بر افزایش سرعت راه‌حل می‌تواند به نمونه‌هایی با جواب‌های دقیق‌تر منتهی گردد. برای مشاهده مثال مربوط به گونه‌ای دیگر از سلسله مرتبه گراف که بر روش‌های چندشبکه‌ای جبری استوار است، به مرجع [1] مراجعه نمایید.

4.1. سلسله مرتبه گراف

با استفاده از انتقال میانگینی که در ادامه به آن پرداخته خواهد شد، سلسله مراتب پیوسته را تعریف می‌نماییم. همانند قبل، داده‌های ما حاوی مجموعه‌ای از بردارهای ویژگی $\{x_i\}_{i=1}^m$ است که در بیشتر کاربردهای موردنظر، هر نقطه داده با پیکسلی از تصویر متناظر می‌باشد. با مشخص نمودن $f_h(x)$ KDE با ماتریس پهنای باند $\mathbf{H} = h^2 \mathbf{I}$ و با تعریف x به عنوان مدی از $f_h(x)$ حوزه موردنظر x می‌باشد یعنی $B(x) = \{y \in \mathbb{R}^d : M^\infty(y) = x\}$. گراف متناظر با پهنای باند h به صورت زیر داده می‌شود

$$G_h = (V_h, E_h, W_h), \text{ where}$$

$$V_h = \{x \in \mathbb{R}^d : x \text{ is a mode of } f_h(\cdot)\}$$

$$E_h = \{(u, v) : u, v \in V(h), B(u) \cap B(v) \neq \emptyset\}$$

و وزن‌های لبه به صورت زیر داده می‌شوند

$$W_h(u, v) = \int_{x \in B(u)} \int_{y \in B(v)} K_h(x, y) dx dy$$

تعریف مجموعه محدب و مجموعه لبه از لحاظ انتقال میانگین، کاملاً طبیعی می‌باشند: مجموعه محدب دقیقاً همان مجموعه مدها و مجموعه لبه همان مجموعه از جفت مدهایی است که حوزه‌های موردنظر در فضای ویژگی همجوار می‌باشند. وزن‌های لبه نیز به شیوه‌ای منطقی و با اجتماع شباهت بین نقاط موجود در حوزه‌های موردنظر همسایه

تعریف شده‌اند و بدین وسیله بسیاری از تعاریف دیگر نیز ممکن می‌باشند. در عمل، انتگرال موجود در تعریف وزن لبه را با مجموع متناظر آن تقریب خواهیم زد.

بدیهی است که این سلسله مراتب گراف پیوسته دارای ویژگی‌های مطلوب زیر می‌باشد:

قضیه 3. سلسله مراتب گراف را به صورتی که در بالا تعریف شده است در نظر می‌گیریم. بنابراین گرفت متناظر با $h = 0$ دارای ویژگی $V_0 = \{x_i\}_{i=1}^m$ است در حالی که گراف متناظر با $h = \infty$ دارای ویژگی $|V_\infty| = 1$ است.

اثبات: مرجع [9] را ملاحظه نمایید.

به عبارت دیگر از گرافی که مجموعه رئوس آن از پیکسل‌های اصلی تشکیل شده است به گرافی با یک رأس منفرد منتقل می‌شویم. این دقیقاً همان چیزی است که از سلسله مراتب چندمقیاسی انتظار می‌رود.

حال برای تبدیل از سلسله مراتب گراف پیوسته به سلسله مراتب گراف گسسته، تعداد پهنای باند ثابت را در نظر می‌گیریم یعنی $G_\ell = G_{h_\ell}$ ، که در آن ترازها یا سطوح از $\ell = 0$ (ریزترین) تا $\ell = L$ (درشت‌ترین) اجرا می‌شوند.

به ویژه در فضای d بعدی، انتخاب $h_\ell = 2^{\frac{1}{d}} h_{\ell-1}$ طبیعی می‌باشد به گونه‌ای که حجم پوشش در هر مرحله دو برابر شده است. در این حالت انتظار داریم که $L \approx \log n$ باشد؛ با استفاده از بحثی مشابه با آنچه که در بخش 2.5 آمده

است می‌توان نشان داد که در محاسبه این سلسله مراتب گراف گسسته با استفاده از انتقال میانگین معمولی به زمان $O(Ln^2) = O(n^2 \log n)$ نیازمندیم. (در اینجا ساختار داده‌های نزدیک‌ترین همسایه‌های خام در نظر گرفته شده

است، برای دسترسی به مطالب با جزئیات بیشتر در این زمینه به بخش 2.5 مراجعه نمایید.)

برای استفاده از الگوریتم انتقال میانگین سریع، دو ملاحظه وجود دارد. نخست، برای سطح ابتدایی ($\ell = 0$) سلسله مراتب، از نمونه‌های $n \gg m_0$ می‌توان استفاده نمود؛ این دقیقاً همان چیزی است که در بخش‌های 2 و 3 به ترتیب

از نقطه نظر نظری و تجربی نشان داده‌ایم. دومین ملاحظه آن است که برای تمامی سطوح بعدی به دلیل افزایش پهنای باند این سطوح می‌توان از نمونه‌های کمتری استفاده نمود. $h_\ell \approx 2^{\frac{1}{d}} h_{\ell-1}$ را داشته و می‌توان برای استفاده در هر

سطح از گراف و با استفاده از نتایج ارائه شده در بخش 2.7 برای انتخاب پهنای باند، مباحث ارائه شده اینجا در خصوص پهنای باندها را به تعدادی از نمونه‌ها تعمیم داد. به ویژه اگر m_ℓ تعداد نمونه‌های موردنیاز در سطح $m_\ell \approx$

$2^{-\frac{d+1}{d}} m_{\ell-1}$ باشد آنگاه را داشته و یا برای d بزرگ، $m_{\ell} \approx \frac{1}{2} m_{\ell-1}$ را خواهیم داشت. بار دیگر و با استفاده از منطق ارائه شده در بخش 2.5، پیچیدگی نسخه سریع سلسله مراتب گراف گسسته را به صورت

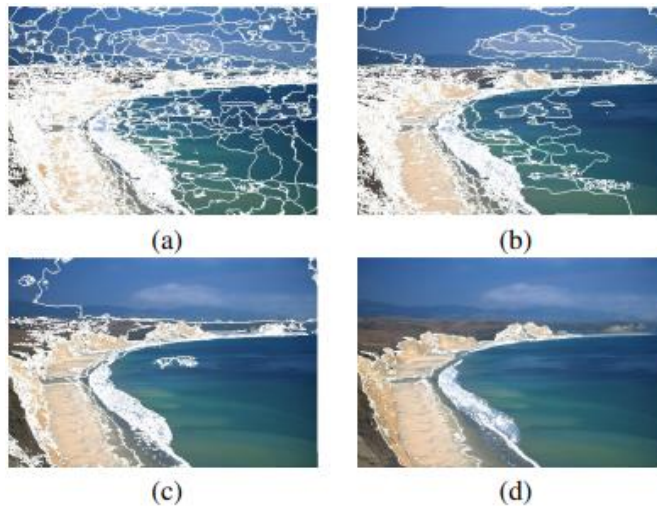
$$O\left(\sum_{\ell=0}^L n 2^{-\ell} m_0\right) = O(nm_0)$$

خواهیم داشت.

شایان ذکر است که ضریب تسریع در محاسبه سلسله مراتب گراف حتی از ضریب تسریع خود انتقال میانگین بهتر می‌باشد؛ یعنی $(n/m) \log n$ در مقابل n/m برای ضریب تسریع انتقال میانگین. در حالت‌های موردنظر که در آنها $n \approx 10^7$ یا 10^8 می‌باشد، ضریب چندان غیرقابل چشم‌پوشی نیست (یعنی 25-20).

4.2. مثال

در شکل 4، بخشی از سلسله مرتبه گراف برای تصویر ساحل دریا نشان داده شده است که با استفاده از روش سریعی که در بالا به آن اشاره شده است، محاسبه گردیده است. چهار تصویر را برای پهنای باندهای 8، 16، 32 و $h = 64$ نشان داده‌ایم. تعداد رئوس موجود برای این گراف‌ها به ترتیب برابر 266، 59، 11 و 3 می‌باشند. با احاطه هر ناحیه همجوار با رنگ سفید، گراف‌ها را مجسم می‌نماییم؛ با این وجود شایان ذکر است که این نواحی همجوار، خود چند قطعه نبوده و اغلب یک یک قطعه (و بنابراین یک راس) شامل چند عدد از این نواحی همجوار می‌باشد. این امر ناشی از این حقیقت می‌باشد که گراف‌ها لزوماً مسطح یا مستوی نیستند. این موضوع در مثال ارائه شده ما نیز صادق می‌باشد چون بردار ویژگی ما به جای بافت، رنگی بدون افزودن موقعیت می‌باشد؛ بنابراین ناحیه بافت‌دار ماسه‌ای به صورت ناحیه‌های با گره‌های متعدد مشاهده می‌شود در حالی که در حقیقت این ناحیه تنها شامل چند گره غیرمجاور است. حتی با در نظر گرفتن این اثر نیز می‌توان دید که ساده‌سازی گراف همانن پهنای باند افزایش می‌یابد.



شکل 4. تجسمی از سلسه مراتب گراف. (الف) $h = 8$, $|V| = 226$, (ب) $h = 16$, $|V| = 59$, (ج) $h = 32$, $|V| = 11$, (د) $h = 64$, $|V| = 3$. مباحث مربوطه در متن مقاله را ملاحظه نمایید.

5. نتیجه گیری

براساس محاسبه KDE کوچک شده و با استفاده از روش نمونه برداری، نسخه سریعی از الگوریتم انتقال میانگین را ارائه نموده ایم. نزدیکی KDE کوچک شده به KDE اصلی و همچنین پیچیدگی فوق العاده الگوریتم انتقال میانگین سریع در مقایسه با انتقال میانگین استاندارد را به صورت نظری نشان داده ایم. سرعت الگوریتم را به صورت تجربی اثبات نموده و کاربردپذیری بالقوه آن در خوشه بندی مجموعه داده های بزرگ همچون ویدیو و در محاسبه ساختار داده های پیچیده ای همچون سلسله مرتبه گراف نشان داده شده را نشان داده ایم. امیدواریم که بتوان این الگوریتم را در کاربردهایی همچون قطعه بندی سریع مجموعه داده های بزرگ موجود همچون تصاویر پزشکی خیلی زود مورد استفاده قرار داد.

References

- [1] S. Alpert, M. Galun, R. Basri, and A. Brandt. Image Segmentation by Probabilistic Bottom-Up Aggregation and Cue Integration. In Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on, pages 1–8, 2007.

- [2] A. Andoni and P. Indyk. Near-Optimal Hashing Algorithms for Approximate Nearest Neighbor in High Dimensions. *Communications of the ACM*, 51(1):117, 2008.
- [3] D. Barash and D. Comaniciu. A common framework for nonlinear diffusion, adaptive smoothing, bilateral filtering and mean shift. *Image and Vision Computing*, 22(1):73–81, 2004.
- [4] D. Comaniciu. An Algorithm for Data-Driven Bandwidth Selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 281–288, 2003.
- [5] D. Comaniciu and P. Meer. Mean Shift: A Robust Approach Toward Feature Space Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 603–619, 2002.
- [6] D. Comaniciu and V. Ramesh. Mean shift and optimal prediction for efficient object tracking. In *Proceedings of the International Conference on Image Processing*, volume 3, 2000.
- [7] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-Based Object Tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 564–577, 2003.
- [8] D. Dementhon. Spatio-temporal segmentation of video by hierarchical mean shift analysis. In *Center for Automat. Res., U. of Md, College Park*, 2002.
- [9] D. Freedman and P. Kisilev. Fast Mean Shift. Technical report, Hewlett-Packard Laboratories, 2009.
- [10] K. Fukunaga and L. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory*, 21(1):32–40, 1975.
- [11] B. Georgescu, I. Shimshoni, and P. Meer. Mean shift based clustering in high dimensions: A texture classification example. In *International Conference on Computer Vision*, pages 456–463, 2003.
- [12] I. Grabec. Self-organization of neurons described by the maximum-entropy principle. *Biological Cybernetics*, 63(5):403–409, 1990.
- [13] H. Guo, P. Guo, and H. Lu. A Fast Mean Shift Procedure with New Iteration Strategy and Re-sampling. In *IEEE International Conference on Systems, Man and Cybernetics, 2006. SMC'06*, volume 3, 2006.
- [14] K. Kim, K. Jung, and J. Kim. Texture-based approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(12):1631–1639, 2003.
- [15] V. Kolmogorov and R. Zabih. What Energy Functions Can Be Minimized via Graph Cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 147–159, 2004.
- [16] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Comm. Pure Appl. Math*, 42(5):577–685, 1989.
- [17] S. Paris and F. Durand. A topological approach to hierarchical segmentation using mean shift. In *Proc. CVPR*, 2007.
- [18] H. Traven. A neural network approach to statistical pattern classification by semiparametric estimation of probability density functions. *Neural Networks, IEEE Transactions on*, 2(3):366–377, 1991.
- [19] A. Vedaldi and S. Soatto. Quick shift and kernel methods for mode seeking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2008.
- [20] J. Wang, B. Thiesson, Y. Xu, and M. Cohen. Image and Video Segmentation by Anisotropic Kernel Mean Shift. *Lecture Notes in Computer Science*, pages 238–249, 2004.
- [21] P. Wang, D. Lee, A. Gray, and J. Rehg. Fast mean shift with accurate and stable convergence. In *Workshop on Artificial Intelligence and Statistics (AISTATS)*, 2007.
- [22] J. Wu and C. Chan. A three-layer adaptive network for pattern density estimation and classification. *International Journal of Neural Systems*, 2(3):211–220, 1991.
- [23] C. Yang, R. Duraiswami, N. Gumerov, and L. Davis. Improved fast Gauss transform and efficient kernel density estimation. In *Ninth IEEE International Conference on Computer Vision, 2003. Proceedings*, pages 664–671, 2003.