

یک روش آماری جهت کوچک کردن تصویر با حفظ محتوا

مهدی نوشیار^۱، استادیار؛ احمد آبستان^۲، دانشجوی کارشناسی ارشد

۱- دانشکده فنی مهندسی - دانشگاه محقق اردبیلی - اردبیل - ایران - nooshyar@uma.ac.ir

۲- دانشکده فنی مهندسی - دانشگاه محقق اردبیلی - اردبیل - ایران - aabsetan@yahoo.com

چکیده: این مقاله یک روش آماری جهت کوچک کردن تصویر با حفظ محتوا پیشنهاد می‌دهد. این روش با استفاده از یک شبکه یکنواخت اولیه تصویر را به ناحیه‌های مربعی یکسان تقسیم می‌کند و سپس با استفاده از نقشه اهمیت برای هر یک از این مربع‌ها یک ضریب اهمیت محاسبه می‌کند. نقشه اهمیت با استفاده از یک ترکیب جدید از چگالی لبه‌ها و نقشه برآمدگی هارل محاسبه می‌شود. الگوریتم پیشنهادی با استفاده از توابع آماری میانگین و واریانس ضرایب اهمیت مربع‌ها، اندازه جدید برای هر یک از مربع‌ها محاسبه می‌کند. برای حفظ نواحی مهم تا حد ممکن و تخریب بیش‌تر نواحی کم‌اهمیت، این روش از توابع آماری برای دادن ارزش بزرگ‌تر به مربع‌های مهم‌تر استفاده می‌کند. برای جلوگیری از خم شدن خطوط افقی، مربع‌هایی که در یک سطر مساوی قرار گرفته‌اند، وقتی اندازه تصویر در جهت عمودی تغییر می‌کند به یک اندازه تغییر می‌کنند؛ و همچنین برای جلوگیری از خم شدن خطوط عمودی تصویر، هر ستون از مربع‌ها به یک اندازه جمع یا کشیده می‌شوند. نتیجه اعمال الگوریتم پیشنهادی بر روی تصاویر مختلف نشان می‌دهد که این الگوریتم هم از نظر کمی و هم کیفیت بصری، عملکرد بهتری نسبت به روش‌های دیگر در حفظ اشیای مهم دارد.

واژه‌های کلیدی: تغییر اندازه تصویر با حفظ محتوا، برجستگی بصری، چگالی لبه، پیچاندن تصویر.

A Statistical Method for Content-Aware Image Shrinking

M. Nooshyar¹, Assistant Professor; A. Absetan², MSc Student

1- Faculty of Engineering, University of Mohaghegh Ardabili, Ardabil, Iran, Email: nooshyar@uma.ac.ir

2- Faculty of Engineering, University of Mohaghegh Ardabili, Ardabil, Iran, Email: aabsetan@yahoo.com

Abstract: This paper proposes a statistical method for content-aware image shrinking. This method divides image into equal quads by an initial uniform grid and computes an important factor for each of quads guided by importance map. The importance map is generated automatically using a novel combination of image edge density and Harel saliency measurement. The method then computes scaling factor for each of quads by using of statistical measures such as the mean and the variance of quads importance factors. To preserve the important areas as far as possible and to destroy the unimportant areas more, this method uses statistical functions to give the larger value to important quads. To prevent bending of straight horizontal lines, in the vertical direction, the algorithm equally resizes quads that are placed in the same row. Also, to prevent bending of straight vertical lines, the algorithm equally squeezes and stretches quads that are placed in the same column. Experimental results of different types of images show that our method can achieve better objective and subjective quality with preserving prominent objects.

Keywords: Content-aware image resizing, visual saliency, edge density, image warping.

تاریخ ارسال مقاله: ۱۳۹۳/۰۶/۰۷

تاریخ اصلاح مقاله: ۱۳۹۳/۰۸/۱۸ و ۱۳۹۳/۱۰/۱۸

تاریخ پذیرش مقاله: ۱۳۹۴/۰۱/۲۵

نام نویسنده مسئول: مهدی نوشیار

نشانی نویسنده مسئول: ایران - اردبیل - بلوار دانشگاه - دانشگاه محقق اردبیلی - دانشکده فنی مهندسی.

۱- مقدمه

دستگاه‌های چندرسانه‌ای مانند تلویزیون‌ها، تلفن‌ها، کامپیوترها و انواع دیگر دستگاه‌ها، دارای صفحات نمایش با اندازه و رزولوشن‌های متفاوت هستند، از این رو تصاویر دیجیتال جهت نمایش در هر یک از این صفحه نمایش‌ها نیاز به تغییر اندازه دارند. روش‌های تغییر اندازه یکنواخت مانند دوخطی و بیکوبیک به محتوای تصویر توجهی نمی‌کنند، بنابراین اعوجاج حاصل از تخریب تصویر را در سراسر آن به صورت یکنواخت پخش می‌کنند و باعث می‌شوند که محتوای تصویر نیز به اندازه بقیه نواحی تصویر تخریب شوند. در برخی موارد محتوای تصویر بیش از اندازه کوچک شده که قابل تشخیص نمی‌باشد. روش‌های بریدن نواحی کم‌اهمیت‌تر اطراف تصویر را برای کوچک کردن تا اندازه دلخواه دور می‌ریزند. بنابراین وقتی اشیاء مهم تصویر در اطراف تصویر واقع شده باشند و یا اندازه نهایی تصویر کوچک‌تر از اندازه اشیاء مهم تصویر باشند، اشیاء مهم نیز حذف می‌شوند.

به دلیل مشکلات ذکر شده بالا، در دهه اخیر الگوریتم‌هایی تحت عنوان «تغییر اندازه تصویر با حفظ محتوا» معرفی شده‌اند. الگوریتم‌های تغییر اندازه تصویر با حفظ محتوا از دو مرحله تشکیل شده‌اند [۱]:

۱. اولین مرحله تشخیص نواحی بااهمیت تصویر است که محتوای تصویر در آن نواحی واقع شده است. خروجی مرحله اول را نقشه اهمیت^۱ تصویر می‌نامند.

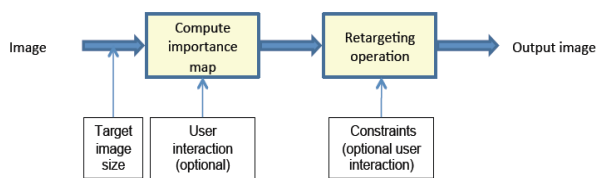
۲. مرحله دوم اجرای الگوریتمی جهت تغییر اندازه تصویر با توجه به نقشه اهمیت که در مرحله یک محاسبه شده است، می‌باشد.

شکل ۱ سیستم تغییر اندازه تصویر با حفظ محتوا را نشان می‌دهد. همان‌طور که در شکل ۱ نیز نشان داده شده است، نقشه اهمیت تصویر در بعضی از کاربردها توسط کاربر و با مشخص کردن نواحی موردنظر مشخص می‌شود. البته اکثر الگوریتم‌ها از محاسبه نقشه اهمیت به صورت اتوماتیک استفاده می‌کنند.

در سال‌های اخیر چندین روش تغییر اندازه با حفظ محتوا معرفی شده‌اند. برخی از روش‌ها که روش‌های گسسته نیز نامیده می‌شوند، پیکسل‌های کم‌اهمیت تصویر را تا رسیدن به اندازه موردنظر حذف می‌کنند. معروف‌ترین روش در این گروه روش Seem Carving (SC) [۲] است. روش SC یک رگه چپ به راست و یا بالا به پایین از پیکسل‌های به هم پیوسته که کم‌ترین انرژی دارند را انتخاب و از تصویر حذف می‌کند. این روش بعداً توسط رابین اشتاین [۳] و همکارانش با تغییر در تابع انرژی تصویر بهبود داده شد. یکی دیگر از روش‌های معروف این گروه شیفت مپ است که تصویر را به صورت یک گراف بیان و آن را حل می‌کند [۴].

روش‌های پیچاندن تصویر که به روش‌های پیوسته نیز معروفند، تصویر را به شبکه‌ای از مستطیل‌ها تقسیم و با توجه به اهمیت آن‌ها، هر یک را به صورت غیریکنواخت تغییر اندازه می‌دهند. گال و همکارانش روشی پیشنهاد کردند که در آن اشیاء مهم تصویر به صورت دستی مشخص می‌شدند [۵]. به تبعیت از روش گال چندین روش دیگر معرفی شدند: ولف و همکارانش این روش را برای تصاویر ویدئویی توسعه دادند

[۶]. وانگ و همکارانش با اضافه کردن محدودیت‌هایی به روش ولف که از خم شدن خطوط محافظت می‌کرد روشی بنام Optimal Scale-and-Stretch (OSS) معرفی کردند [۷] و همچنین روش ViSizer با توسعه روش وانگ برای تصاویر شامل ابر کلمات، گراف و نقشه پراکندگی پیشنهاد شده است [۸]. این مقاله روش جدیدی برای پیچاندن تصویر ارائه می‌کند که از توابع آماری و یک تابع اهمیت جدید برای تغییر اندازه تصویر با حفظ محتوا استفاده می‌کند.



شکل ۱: مراحل الگوریتم‌های تغییر اندازه تصویر [۱]

بعضی روش‌ها نیز از ترکیب چند روش برای تغییر اندازه تصویر استفاده کرده‌اند، هدف این روش‌ها این است که ترکیبی پیدا کنند که تصویر خروجی بیش‌ترین شباهت با تصویر ورودی داشته باشد. رابین اشتاین و همکارانش با استفاده از نتیجه نظرسنجی از کاربران دریافتند که آن‌ها روش‌هایی که از چند عملگر استفاده کرده‌اند را به روش‌های تک عملگری ترجیح می‌دهند، آن‌ها الگوریتمی ارائه دادند که از ترکیب SC، تغییر اندازه یکنواخت و بریدن برای تغییر اندازه تصویر استفاده می‌کند [۹]. دانگ و همکارانش از ترکیب SC و تغییر اندازه یکنواخت استفاده کرده‌اند و از الگوریتم فاصله تصویر که توسط سیماکوف [۱۰] ارائه شده است برای ارزیابی شباهت تصویر خروجی با ورودی استفاده کردند [۱۱].

روش‌های مبتنی بر قطعه، تصاویر را به قطعات مهم و غیرمهم تقسیم کرده و سپس برای هر قسمت یک ضریب مقیاس گذاری جداگانه محاسبه و آن‌ها را تغییر اندازه می‌دهند. معروف‌ترین روش مبتنی بر قطعه را لیانگ و همکاران [۱۲] ارائه دادند.

هر یک از روش‌های تغییر اندازه با حفظ محتوا مشکلاتی دارند، برای مثال، الگوریتم SC [۲]، در صورتی که اندازه تصویر خروجی کم‌تر از اندازه محتوای تصویر باشد، تصویر را دچار اعوجاج می‌کند. یا روش‌های چند عملگری، چون از بریدن تصویر در هنگام کوچک کردن تصویر استفاده می‌کنند، اطلاعات تصویر را دور می‌ریزند. عملکرد روش‌های موجود روی تصاویر مختلف متفاوت است و ممکن است روشی روی یک تصویر بهترین عملکرد داشته باشد و روی تصویر دیگر عملکرد خوبی نداشته باشند.

در این مقاله یک روش کوچک کردن تصویر با حفظ محتوا پیشنهاد شده است. این روش ابتدا یک تابع اهمیت جدید را معرفی می‌کند که در تشخیص نواحی بااهمیت تصویر عملکرد بهتری نسبت به روش‌های موجود دارد و همچنین بهترین عملکرد بر روی الگوریتم ارائه شده دارد. این روش ابتدا تصویر را با یک شبکه یکنواخت می‌پوشاند و سپس با استفاده از توابع اهمیت برای هر یک از مربع‌های شبکه یک فاکتور

اهمیت هر پیکسل استفاده کرده‌اند. گرادیان تصویر به صورت یکی از روابط (۲) یا (۳) محاسبه می‌شود.

$$E_1(I) = \left| \frac{\partial}{\partial x} I \right| + \left| \frac{\partial}{\partial y} I \right| \quad (2)$$

$$E_2 = \sqrt{\left(\frac{\partial}{\partial x} I \right)^2 + \left(\frac{\partial}{\partial y} I \right)^2} \quad (3)$$

در سال ۲۰۱۲، لیانگ و همکاران [۱۲]، نقشه اهمیت جدیدی با استفاده از ترکیب گرادیان و نقشه برآمدگی تصویر ارائه دادند:

$$I_{imp} = 10\alpha I_s + (1 - \alpha)I_e \quad (4)$$

در این رابطه I_s نقشه برآمدگی هارل [۱۴] است که به وسیله زنجیره مارکوف قسمت‌های برجسته تصویر را مشخص می‌کند و I_e گرادیان تصویر است که با استفاده از عملگر سوبل محاسبه می‌شود. α یک ضریب ثابت است که در الگوریتم لیانگ ۰/۵ در نظر گرفته شده است.

در اینجا یک نقشه اهمیت جدید ارائه می‌شود که محتوای تصویر را بهتر تشخیص داده و عملکرد بهتری در الگوریتم پیشنهادی دارد. با الهام از روش وانگ [۷]، که نقشه اهمیت تصویر را با استفاده از ضرب نقشه برآمدگی در گرادیان تصویر به دست می‌آورد، نقشه اهمیت پیشنهادی نیز از ضرب نقشه برآمدگی هارل [۱۴] در نقشه چگالی لبه‌ها به دست می‌آید. برای نمایان کردن ساختار تصویر، روش پیشنهادی از نقشه چگالی لبه‌ها و برای مشخص کردن نواحی جذاب که توجه انسان را به خود جلب می‌کند از نقشه برآمدگی استفاده می‌کند. روزن هول و همکاران در سال ۲۰۰۵، سه معیار برای درهمی بصری ارائه دادند، شامل خصوصیات، آنتروپی و چگالی لبه‌ها. چگالی لبه‌ها، برای نشان دادن نواحی از تصویر که در آن تعداد اشیاء تصویر زیاد است و درواقع تراکم لبه‌ها زیاد است، تلاش می‌کند [۱۵]. در نقشه اهمیت پیشنهادی، از این معیار برای مشخص کردن ساختار اشیاء موجود در تصویر استفاده شده است.

تیزی لبه‌های تشخیص داده شده توسط چگالی لبه‌ها کم‌تر از روش گرادیان است و به همین دلیل در موقعی که تصویر حاصل از این معیار در نقشه برآمدگی تصویر برای محاسبه نقشه اهمیت ضرب می‌شود، قسمت‌های برجسته تصویر در نقشه اهمیت حاصل از این ضرب بهتر حفظ می‌شود.

اگر تصویر ورودی I باشد، نقشه اهمیت تصویر ورودی برابر است با:

$$I_{imp} = I_s \times I_d \quad (5)$$

I_s نقشه برآمدگی هارل و I_d نقشه چگالی لبه‌ها است. ابعاد ماتریس‌های روابط بالا هم اندازه با ابعاد تصویر اصلی هستند. نقشه محاسبه شده I_{imp} برای کاربرد در الگوریتم پیشنهادی، باید نرمال شود و اندازه هر پیکسل از آن در محدوده صفر و یک قرار گیرد:

اهمیت محاسبه می‌کند. جهت جلوگیری از خم شدن خطوط عمودی و افقی، مربع‌هایی که در یک ستون قرار دارند به طور مساوی در جهت افقی و مربع‌هایی که در یک سطر قرار دارند به طور مساوی در جهت عمودی تغییر اندازه می‌دهند. الگوریتم پیشنهادی با استفاده از توابع آماری میانگین و واریانس، یک فاکتور اهمیت برای هر سطر و ستون محاسبه می‌کند و سپس با استفاده از این فاکتورها، اندازه جدید هر مربع از تصویر را محاسبه می‌کند. سپس با استفاده از روش بیکویک هر یک از مربع‌ها را تغییر اندازه می‌دهد تا به اندازه تصویر موردنظر برسد. در ادامه، بخش ۲ روش پیشنهادی را تشریح می‌کند، بخش ۳ نتایج شبیه‌سازی بررسی می‌کند.

۲- روش پیشنهادی

این بخش اسلوب^۲ و مراحل انجام روش پیشنهادی را ارائه می‌کند. شکل (۲) شمای کلی الگوریتم پیشنهادی را نشان می‌دهد. در روش پیشنهادی ابتدا نقشه اهمیت تصویر محاسبه می‌شود (شکل (۲) قسمت (ب)). بعد از محاسبه نقشه اهمیت تصویر ورودی توسط یک شبکه یکنواخت از چهارگوش‌ها پوشیده می‌شود (شکل (۲) قسمت (د)). این شبکه به صورت $G = \{Q\}$ که Q مجموعه چهارگوش‌ها می‌باشد، نشان می‌دهیم. برای تغییر اندازه یک تصویر $m \times n$ پیکسلی به اندازه دلخواه $m' \times n'$ پیکسل، برای سادگی محاسبات، می‌خواهیم که همه چهارگوش‌های ابتدایی به صورت مربعی باشند. برای محقق شدن این خواسته، قبل از اجرای الگوریتم ما ابتدا تصویر ورودی را به اندازه جدید $m_I \times n_I$ که m_I و n_I به صورت رابطه (۱) محاسبه می‌شوند، می‌رسانیم.

$$m_I = \left\lfloor \frac{m}{n_q} \right\rfloor \times n_q \quad n_I = \left\lfloor \frac{n}{n_q} \right\rfloor \times n_q \quad (1)$$

که در اینجا n_q اندازه ضلع مربع‌ها در شبکه ابتدایی است که تصویر ورودی را می‌پوشاند.

الگوریتم پیشنهادی از پنج مرحله تشکیل شده است:

- مرحله اول: محاسبه نقشه اهمیت تصویر.
- مرحله دوم: تقسیم تصویر ورودی به شبکه‌ای از مربع‌ها و محاسبه ضریب اهمیت برای هر یک از آن‌ها.
- مرحله سوم: محاسبه ضریب اهمیت برای هر سطر و ستون از مربع‌ها.
- مرحله چهارم: محاسبه اندازه نهایی برای هر مربع بر اساس ضریب اهمیت سطرها و ستون‌ها.
- مرحله پنجم تغییر اندازه هر یک از مربع‌ها به اندازه جدید محاسبه شده.

۲-۱- مرحله اول: محاسبه نقشه اهمیت تصویر

نقشه اهمیت تصویر معیاری^۳ است به صورت تصویر و درواقع هسته اصلی الگوریتم‌های تغییر اندازه تصویر با حفظ محتوا است [۸]. روش‌های تغییر اندازه قبلی از معیارهای مختلف برای تشخیص اهمیت پیکسل‌های تصویر استفاده کرده‌اند. اویدان و شمیر [۲]، از گرادیان برای محاسبه



شکل ۲: شمای کلی الگوریتم پیشنهادی؛ (الف) تصویر اصلی، (ب) نقشه اهمیت (مرحله اول)، (ج) تصویر پوشیده شده با شبکه‌ای از چهارگوش‌ها (مرحله دوم)، (د) محاسبه ضرایب برای سطر و ستون‌ها (مرحله سوم)، (ه) محاسبه اندازه نهایی هر چهارگوش (مرحله چهارم)، (و) تصویر نهایی همراه با شکل تغییر یافته شبکه اولیه (مرحله پنجم) و (ز) تصویر خروجی

۲-۲- مرحله دوم: تقسیم تصویر به شبکه‌ای از مربع‌ها محاسبه ضریب اهمیت برای هر یک از آن‌ها

بعد از محاسبه نقشه اهمیت تصویر، الگوریتم پیشنهادی تصویر ورودی و همچنین نقشه اهمیت محاسبه شده در مرحله قبل را به وسیله شبکه یکسانی از چهارگوش‌ها می‌پوشاند. البته همان‌طور که قبلاً اشاره شده این تصویر ابتدا با استفاده از رابطه (۱) به اندازه جدید $m_I \times n_I$ تبدیل شده است. این اندازه جدید تفاوت چندانی با اندازه اصلی تصویر ندارد و فقط برای اینکه محاسبات راحت‌تر باشد و همه چهارگوش‌ها مربعی باشند، تصویر را به اندازه‌ای که بر n_q بخش پذیر شود تغییر داده می‌شود. تعداد سطر و ستون‌های شبکه‌ای که با آن تصویر پوشانده می‌شود، به ترتیب برابر است با n_r و n_c که به صورت زیر محاسبه می‌شوند:

$$n_r = \frac{m_I}{n_q} \quad n_c = \frac{n_I}{n_q} \quad (7)$$

در این رابطه n_q اندازه هر ضلع از مربع‌های شبکه است که در الگوریتم پیشنهادی برابر ۲۰ است. ماتریس F یک ماتریس $n_r \times n_c$ است و هر درایه از آن معرف اهمیت مربع متناظرش در شبکه می‌باشد. برای محاسبه ضریب اهمیت هر مربع، میانگین پیکسل‌های مربع متناظر با آن در نقشه اهمیت محاسبه و در درایه متناظرش در ماتریس F قرار می‌دهیم:

$$F(i, j) = 1/n_q^2 \left(\sum_{k=1}^{n_q} \sum_{l=1}^{n_q} S_{i,j}(k, l) \right) \quad (8)$$

$S_{i,j}$ مربع متناظر با مربع $q_{i,j}$ در نقشه اهمیت I_{imp} می‌باشد.

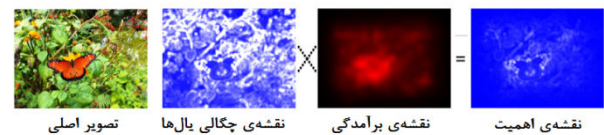
چون اشیاء مهم تصویر معمولاً چندین چهارگوش از تصویر را می‌پوشانند، بنابراین ضریب اهمیت هر چهارگوش با ضریب اهمیت چهارگوش‌های همسایه‌اش تقریباً مشابه است. بنابراین درایه‌های به دست آمده در رابطه (۸) با رابطه (۹) به روز می‌شود.

$$F(i, j) = 1/9 \left(\sum_{k=1}^3 \sum_{l=1}^3 N_{i,j}(k, l) \right) \quad 1 \leq i, j \leq n_q \quad (9)$$

در رابطه بالا $N_{(i,j)}$ پنجره 3×3 پیکسلی است که $F(i,j)$ در مرکز آن قرار می‌گیرد. بدین ترتیب ضریب اهمیت هر چهارگوش به دست می‌آید و در موقع تغییر اندازه تصویر سعی می‌شود که چهارگوش‌هایی که ضریب اهمیت بالاتری دارند کم‌تر دچار تخریب شوند. قسمت (ج) شکل ۵ شبکه اولیه است که تصویر قسمت (الف) که تصویر ورودی است را می‌پوشاند. قسمت (ب) شکل ۵ نقشه اهمیت محاسبه شده به روش پیشنهادی است.

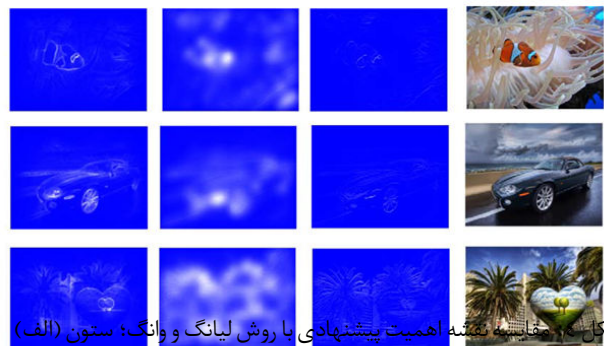
$$I_{imp} = \frac{I_{imp}}{\max(I_{imp})} \quad (6)$$

در رابطه بالا $\max(I_{imp})$ بزرگ‌ترین پیکسل I_{imp} است. پیکسل با اندازه کوچک‌تر به این معنی است که پیکسل متناظر با آن در تصویر ورودی دارای ارزش کم‌تری است. شکل ۳ نحوه محاسبه نقشه اهمیت پیشنهادی معادل رابطه (۷) را نشان می‌دهد.



شکل ۳: نحوه محاسبه نقشه اهمیت پیشنهادی معادل با رابطه (۵)

در مقایسه با الگوریتم‌هایی که برای محاسبه نقشه اهمیت تصویر توضیح داده شد، نقشه اهمیت پیشنهادی در تشخیص محتوای تصویر بسیار بهتر عمل می‌کند. همان‌طور که در شکل ۴ نشان داده شده است، نقشه اهمیت پیشنهادی نسبت به دوروشی که توسط وانگ [۷] و لیانگ [۱۲] پیشنهاد شده‌اند، بهتر عمل کرده است. روش لیانگ نقشه اهمیت را به صورت مات^۴ نشان می‌دهد و ساختار تصویر در این روش به خوبی نمایان نیست و این روش بیش‌تر شبیه به همان نقشه برآمدگی تصویر است. در روش دانگ نیز به دلیل ضرب نقشه برآمدگی در گرادیان، قسمت‌های برجسته تصویر حذف می‌شود، چون در گرادیان نواحی غیر از لبه‌ها به صورت صفر نشان داده می‌شود و لبه‌ها خیلی تیز و با عرض کم هستند.



شکل ۴: مقایسه نقشه اهمیت پیشنهادی با روش لیانگ و وانگ؛ (الف) تصویر اصلی، (ب) نقشه اهمیت محاسبه شده به روش لیانگ، (ج) نقشه اهمیت محاسبه شده به روش وانگ و (د) نقشه اهمیت پیشنهادی

درواقع ضریب اهمیت هر سطر میانگین ضرایب اهمیت چهارگوش‌های واقع در آن سطر و ضریب اهمیت هر ستون میانگین ضرایب اهمیت چهارگوش‌های واقع در آن ستون است.

نتایج حاصل از شبیه‌سازی‌های متعدد به ما نشان می‌دهد که برای محافظت از نواحی مهم تصویر، دو پارامتر می‌تواند مهم باشد: نسبت اندازه تصویر ورودی به تصویر خروجی و واریانس ضرایب اهمیت جهتی که تصویر تغییر اندازه می‌دهد. برای روشن شدن چگونه عمل کردن الگوریتم پیشنهادی در این مرحله، فقط کوچک کردن تصویر در جهت افقی بررسی می‌شود. برای کوچک کردن یک تصویر $m \times n$ پیکسلی به اندازه جدید $m' \times n'$ پیکسلی، اینکه نسبت اندازه n به n' چقدر باشد، مهم است. هرچه قدر این اندازه بزرگ‌تر باشد، درواقع باید از محتوای تصویر بیش‌تر محافظت شود. اگر واریانس ضرایب اهمیت ستون‌ها که با استفاده از رابطه (۱۱) به دست آمده است، بزرگ باشد، بدین معنی است که محتوای تصویر ستون‌های کم‌تری از تصویر را پوشانده‌اند. بنابراین می‌توان نواحی مهم تصویر را با جمع کردن نواحی غیر مهم تصویر بیش‌تر حفظ کرد. در الگوریتم پیشنهادی ستون‌های مهم‌تر، کم‌تر و ستون‌های غیرمهم، بیش‌تر تخریب می‌شوند. با توجه به توضیحات بالا، ضرایب اهمیت ستون‌ها را به‌صورت رابطه (۱۲) اصلاح می‌شود.

$$F_c(i) = (F_c(i))^{v+r} \quad (12)$$

متغیرهای v و r در رابطه (۱۲) به وسیله رابطه (۱۳) محاسبه می‌شوند.

$$v = 1000(\text{variance}(F_c)) , \quad r = \frac{n}{n'} \quad (13)$$

درواقع متغیر v حاصل ضرب عدد ۱۰۰۰ در واریانس ماتریس F_c و متغیر r نسبت اندازه تصویر ورودی به تصویر خروجی است. این تابع نمایی باعث می‌شود که فاصله بین ضرایب ستون‌های مهم و غیرمهم تصویر به‌صورت نمایی افزایش پیدا کند. یادآور می‌شویم که ستون‌هایی که ضریب بزرگ‌تری داشته باشند، مهم‌تر می‌باشند و ضرایب، عددی بین صفر و یک هستند.

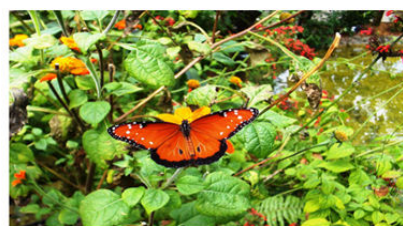
۲-۴- مرحله چهارم: محاسبه اندازه نهایی برای هر مربع بر اساس

ضریب اهمیت سطرها و ستون‌ها

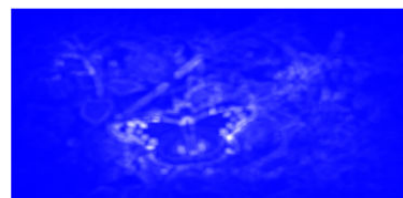
در اینجا با استفاده از ماتریس‌های F_c که با استفاده از رابطه (۱۲) به دست آمده است، اندازه نهایی هر یک از چهارگوش‌ها محاسبه خواهد شد. شبکه اولیه که تصویر را می‌پوشاند و تصویر را به چهارگوش‌هایی که همه آن‌ها در ابتدا مربعی هستند تقسیم می‌کند، به‌صورت غیریک‌نواخت تغییر اندازه داده خواهد شد.

برای تغییر اندازه تصویر، ابتدا تصویر را در جهت افقی و سپس به‌صورت عمودی کوچک می‌کنیم. چون نحوه کوچک کردن تصویر در هر جهت مشابه جهت مخالف است، در اینجا فقط نحوه تغییر اندازه تصویر به‌صورت افقی را شرح خواهیم داد.

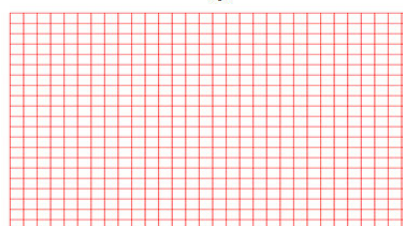
فرض کنید تصویر ورودی $m \times n$ باشد و می‌خواهیم آن را تا اندازه جدید $m \times n'$ تغییر اندازه دهیم. واضح است که n' از n کوچک‌تر



(الف)



(ب)



(ج)

شکل ۵: شبکه اولیه متناظر با تصویر ورودی؛ (الف) تصویر اصلی، (ب) نقشه اهمیت و (ج) شبکه متناظری که این دو را می‌پوشاند.

۲-۳- مرحله سوم: محاسبه ضریب اهمیت برای هر سطر و ستون از مربع‌ها

برای تغییر اندازه تصویر، الگوریتم پیشنهادی دو هدف را دنبال می‌کند: اول اینکه تا جایی که ممکن است از نواحی با اهمیت تصویر در موقع تغییر اندازه تصویر محافظت کند و دوم اینکه از خم شدن خطوط عمودی و افقی تصویر جلوگیری کند، چون خم شدن خطوط عمودی و افقی تخریب و اعوجاج حاصل از تغییر اندازه تصویر را بیش‌تر نمایان می‌کند. برای تحقق هدف دوم، در این الگوریتم برای هر سطر از چهارگوش‌ها در موقع کوچک کردن تصویر به‌صورت عمودی و برای هر ستون از چهارگوش‌ها در موقع کوچک کردن به‌صورت افقی ضریب مقیاس‌گذاری یکسانی در نظر گرفته می‌شود. برای تحقق هدف اول از توابع آماری برای دادن ضریب مقیاس‌گذاری بزرگ‌تر به چهارگوش‌های مهم در ماتریس F که بعداً ما در محاسبه اندازه نهایی هر چهارگوش از آن استفاده می‌کنیم، استفاده می‌شود.

با توجه به توضیحات بالا، در این روش دو ماتریس F_r و F_c تعریف می‌شود که دو ماتریس یک‌بعدی به ترتیب برابر n_r و n_c می‌باشند. درایه‌های F_r ضریب اهمیت هر سطر از چهارگوش‌ها را و درایه‌های F_c ضریب اهمیت هر ستون از چهارگوش‌ها را مشخص می‌کنند. هر یک از درایه‌های ماتریس F_r و F_c برابر است با:

$$F_r(i) = \text{mean}(\text{row}_i) \quad (11)$$

$$F_c(i) = \text{mean}(\text{col}_i) \quad (12)$$

- می باشد. بنابراین باید $n - n'$ پیکسل را از هر سطر از پیکسل های تصویر حذف شود. به عبارت دیگر، باید تصویر به اندازه $n - n'$ پیکسل در جهت افقی کوچک شود. همان طور که قبلاً نیز اشاره شد، در هنگام تغییر اندازه تصویر در جهت افقی، چهارگوش های واقع در یک ستون باید دارای ضریب مقیاس گذاری یکسانی باشند.
- برای تغییر اندازه تصویر در جهت افقی، از ضرایب موجود در ماتریس F_c پیکسل هایی که باید در جهت افقی از تصویر حذف شوند، بین ستون های شبکه ای که تصویر را پوشانده است، تقسیم می شوند. واضح است که از ستون هایی که ضریب اهمیت بزرگتری دارند، پیکسل های کمتری حذف می شود.
- این مرحله از الگوریتم به صورت شبه کدی که در زیر نشان داده شده است، ارائه می شود. خروجی این الگوریتم گراف G' با چهارگوش های با اندازه جدید است.

می باشد. بنابراین باید $n - n'$ پیکسل را از هر سطر از پیکسل های تصویر حذف شود. به عبارت دیگر، باید تصویر به اندازه $n - n'$ پیکسل در جهت افقی کوچک شود. همان طور که قبلاً نیز اشاره شد، در هنگام تغییر اندازه تصویر در جهت افقی، چهارگوش های واقع در یک ستون باید دارای ضریب مقیاس گذاری یکسانی باشند.

این مرحله از الگوریتم به صورت شبه کدی که در زیر نشان داده شده است، ارائه می شود. خروجی این الگوریتم گراف G' با چهارگوش های با اندازه جدید است.

Algorithm 1: calculating final size of each quads

```

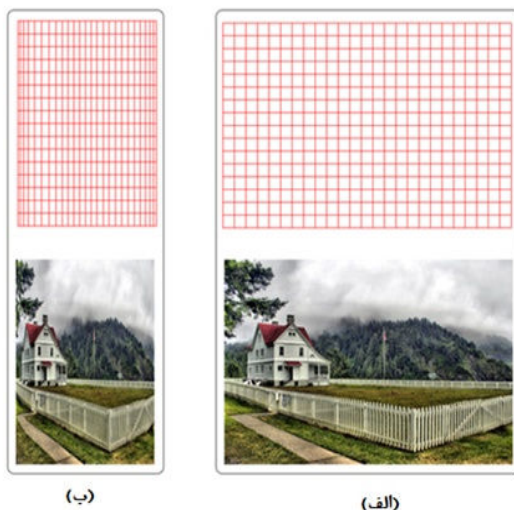
M ← n - n'
Removed ← 0
for each coli ∈ G' do
    s ← M * (Fc(l) / Sum(Fc))
    if s < nq then
        coli.horiz_size ← nq - s
        Removed ← Removed + s
    else
        coli.horiz_size ← 1
        Removed ← Removed + nq - 1
    end if
end for
if Removed < M then
    while Removed ≠ M repeat
        for i = 1 to nc do
            if coli.horiz_size > 1 then
                coli.horiz_size = coli.horiz_size - 1
                Removed = Removed + 1
            end if
            if Rq = M then
                break
            end if
        end for
    end while
end if
for each q' ij ∈ Q' do
    q' ij.horiz_size ← colj.horiz_size
end for
return G'

```

۵-۲- مرحله پنجم تغییر اندازه هر یک از مربع ها به اندازه جدید محاسبه شده

در این مرحله هر یک از چهارگوش ها به اندازه نهایی محاسبه شده در مرحله قبل تغییر اندازه داده می شوند. این کار با استفاده از روش درونیابی بیکویک انجام می شود.

شکل ۶ تصویر ورودی و خروجی و شبکه اولیه متناظر با تصویر ورودی و شبکه تغییر اندازه داده شده متناظر با تصویر خروجی را نشان می دهد.



شکل ۶: تصویر ورودی و خروجی و شبکه متناظر با آن ها؛ (الف) تصویر ورودی و (ب) تصویر خروجی

۳- نتایج شبیه سازی

برای تست الگوریتم پیشنهادی، آن را بر روی یک کامپیوتر شخصی با پردازنده Intel(R) core(TM) i3-2120 CPU 3.30 GHz با ۳/۴ گیگابایت حافظه RAM پیاده سازی کرده ایم. برای تغییر اندازه یک تصویر ورودی به اندازه ۱۰۲۴×۶۴۶ پیکسل به یک تصویر خروجی به اندازه ۵۱۲×۶۴۶ پیکسلی، این الگوریتم ۰/۸۷ ثانیه زمان می برد که نسبت به خیلی از الگوریتم ها سریع تر می باشد و این الگوریتم قابل استفاده برای کاربردهای زمان واقعی می باشد.

در روش های مقیاس گذاری، اگر تصویر به یک نسبت در طول و عرض تغییر اندازه دهد، طبیعت تصویر حفظ می کند. این روش ها چون برای

توضیح متغیرها و حلقه های به کاررفته در شبه کد بالا به صورت زیر است:

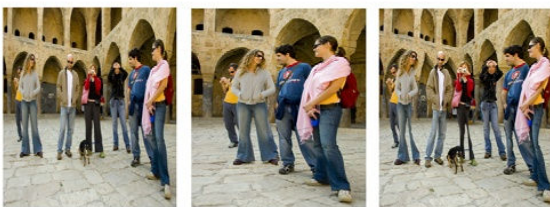
- M : تعداد پیکسل هایی است که باید در جهت افقی از تصویر حذف شوند.
- $Removed$: شمارشگری برای تعداد پیکسل های حذف شده.
- حلقه For اول تعداد پیکسل هایی که از هر ستون باید حذف شود را محاسبه می کند.
- متغیر s در هر بار اجرای حلقه For، تعداد پیکسل هایی که باید از ستون i ام حذف شود را در بر دارد.

روش شیفت مپ [۴] نواحی از تصویر که به هم مرتبط هستند و اهمیت کمتری دارند را بر اثر شیفت نواحی دیگر بر روی آن حذف می‌کند. این کار باعث حذف اطلاعات تصویر می‌شود. نتیجه الگوریتم شیفت مپ مانند روش‌های بریدن می‌باشد که اطلاعات تصویر را حذف می‌کند. قسمت (د) شکل ۸ با روش شیفت مپ کوچک شده است. همان‌طور که در شکل می‌بینید بعضی از افراد از تصویر حذف شده‌اند. قسمت (ه) شکل ۸ با روش پیشنهادی کوچک شده است که نسبت به سه روش دیگر طبیعی‌تر و دارای اعوجاج کم‌تر می‌باشد.



(ب)

(الف)



(ه)

(د)

(ج)

شکل ۸: مقایسه الگوریتم پیشنهادی با الگوریتم‌های گسسته؛ (الف) تصویر اصلی، (ب) روش SC [۱]، (ج) روش بهبود یافته SC [۲]، (د) روش شیفت مپ [۳] و (ه) روش پیشنهادی. اندازه تصویر از 1024×683 به اندازه 512×341 تغییر کرده است.



(ب)

(الف)



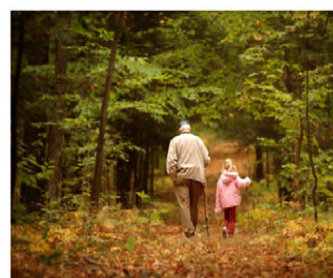
(د)

(ج)

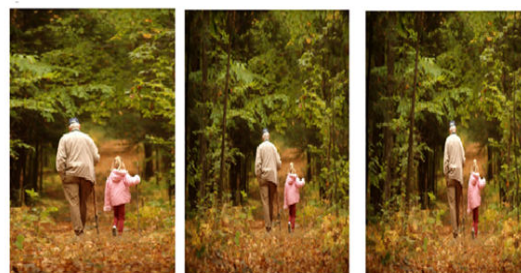
شکل ۹: مقایسه الگوریتم پیشنهادی با الگوریتم‌های پیوسته؛ (الف) تصویر اصلی، (ب) روش ولف [۶]، (ج) روش وانگ [۷] و (د) روش پیشنهادی. اندازه تصویر از 1024×697 به اندازه 512×348 تغییر کرده است.

همه پیکسل‌های تصویر اهمیت یکسانی می‌دهد و به محتوای تصویر توجهی نمی‌کنند، باعث می‌شوند که زمانی که تصویر کوچک می‌شود، محتوای تصویر گاهی اوقات آن‌قدر کوچک شوند که به درستی قابل تشخیص نباشد. برای مثال، قسمت (ب) شکل ۷ تصویر به روش مقیاس‌گذاری بیکویک کوچک شده است. قسمت (ج) شکل ۷ که به روش پیشنهادی کوچک شده است، محتوا هم بهتر حفظ شده و هم نسبت طول و عرض بهتر حفظ شده است.

روش‌های بریدن تصویر، سعی می‌کنند که قسمت‌هایی از تصویر که اهمیت کمتری دارند را از تصویر حذف کنند. این کار باعث حذف کلی اطلاعاتی می‌شود که در قسمت‌های حذف شده وجود دارند. قسمت (د) شکل ۷ به روش بریدن کوچک شده است. همان‌طور که در شکل مشخص است، قسمت‌های از چپ و راست تصویر حذف شده‌اند.



(الف)



(د)

(ج)

(ب)

شکل ۷: مقایسه الگوریتم پیشنهادی با روش‌های مقیاس‌گذاری و بریدن؛ (الف) تصویر اصلی، (ب) به روش بیکویک، (ج) به روش پیشنهادی و (د) به روش بریدن کوچک شده است. اندازه تصویر از 600×429 به اندازه 300×214 تغییر کرده است.

صرف‌نظر از زمان اجرای^۵ بالا، الگوریتم SC [۲] زمانی که نواحی یکنواخت تصویر زیاد باشد، خوب عمل می‌کند. اما زمانی که اندازه محتوای تصویر از اندازه تصویر خروجی بزرگ‌تر باشد، این روش محتوای تصویر را تخریب و باعث ایجاد اعوجاج در تصویر خروجی می‌شود. در قسمت (ب) شکل ۸ تصویر ورودی با روش SC کوچک شده است. همان‌طور که در شکل مشاهده می‌شود، چون محتوای تصویر که افراد می‌باشند، تقریباً سرتاسر تصویر را پوشانده‌اند، این روش محتوای تصویر را تخریب و باعث اعوجاج در تصویر خروجی شده است. برای مثال پای افراد در تصویر تخریب شده و تصویر غیرطبیعی به نظر می‌رسد.

روش بهبود یافته SC [۳] نیز که در قسمت (ج) شکل ۸ نشان داده شده است، هرچند نسبت به روش SC بهتر عمل کرده است ولی همچنان باعث تخریب محتوای تصویر شده است.

پیشنهادی کوچک شده است. همان طور که می بینید طبیعی تر است. برای مثال صورت خرس و پسر بچه که محتوای مهم تصویر هستند، نسبت به دوروش طبیعی تر است.

بیشتر روش های به صورت بصری، نتیجه روش خود را با دیگر روش ها مقایسه می کنند. چون تمرکز اصلی این الگوریتم ها بر حفظ محتوا تصویر است، این مسئله خود باعث می شود که فاصله تصویر خروجی از تصویر اصلی زیاد شود. اما در اینجا الگوریتم پیشنهادی را بر اساس معیار شباهت پیشنهاد شده توسط سیماکوف [۱۰] با معروف ترین روش های تغییر اندازه تصویر با حفظ محتوا مقایسه شده است. الگوریتم سیماکوف برای ارزیابی عملکرد الگوریتم تغییر اندازه تصویر، فاصله تصویر خروجی را از تصویر ورودی محاسبه می کند. فاصله کم تر نشان دهنده عملکرد بهتر الگوریتم می باشد. در اینجا، پنج تصویر استاندارد (شکل های ۱۵-۱۱) که از پایگاه داده استاندارد که رابین اشتاین و همکاران جمع آوری کرده و اکثر روش ها نیز از این تصاویر استفاده کرده اند، انتخاب شده و تصویر را با روش پیشنهادی کوچک شده و سپس با استفاده از الگوریتم سیماکوف با دیگر الگوریتم ها مقایسه می شود. در شکل ها و نمودارهایی که در ادامه آمده است، از علائم اختصاری برای بیان هر روش استفاده شده است که به صورت زیر می باشند.

- Original Image: تصویر اصلی.
- SCL: روش مقیاس گذاری بیکوبیک.
- CR: روش بریدن.
- SC: روش Seam Carving [۲].
- ISC: روش بهبود یافته [۳].
- SM: روش شیفت مپ [۴].
- NHVR: روش ولف [۶].
- OSS: روش وانگ [۷].
- MO: روش چند عملگری رابین اشتاین [۹].
- SV: [۱۶].
- Ours: روش پیشنهادی.

در شکل های که در ادامه آمده است (شکل های ۱۵-۱۱) پنج مقایسه جامع با دیگر الگوریتم های معروف انجام شده است. همان طور که در نمودارها مشخص است خروجی الگوریتم پیشنهادی از نظر کمی نیز در مقایسه با اکثر الگوریتم ها نتیجه بهتر و در واقع فاصله کمتری نسبت به سایر روش ها به تصویر اصلی دارد.

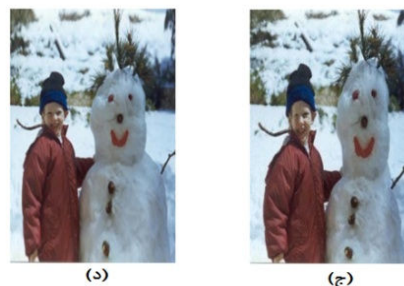
جدول ۱ با استفاده از روش سیماکوف [۱۰] فاصله خروجی های روش های مختلف تغییر اندازه تصویر را با تصویر ورودی محاسبه می کند. روش سیماکوف [۱۰] در ضمیمه تشریح شده است. همان طور که در شکل مشخص است خروجی الگوریتم پیشنهادی از نظر کمی نیز در هر مقایسه از اکثر الگوریتم ها بهتر عمل کرده است. یادآور می شویم که هر چه فاصله تصویر ورودی و خروجی کم تر باشد، عملکرد الگوریتم بهتر و تصویر خروجی به تصویر ورودی بیش تر شباهت خواهد داشت.

الگوریتم پیشنهادی ولف [۶] تصویر را بدون توجه به خطوط مستقیم موجود در تصویر می پیچاند. قسمت (ب) شکل ۹ تصویر به روش ولف کوچک شده است. نواحی داخل مستطیل نشان می دهد که این روش خطوط مستقیم موجود در تصویر را دچار اعوجاج می کند.

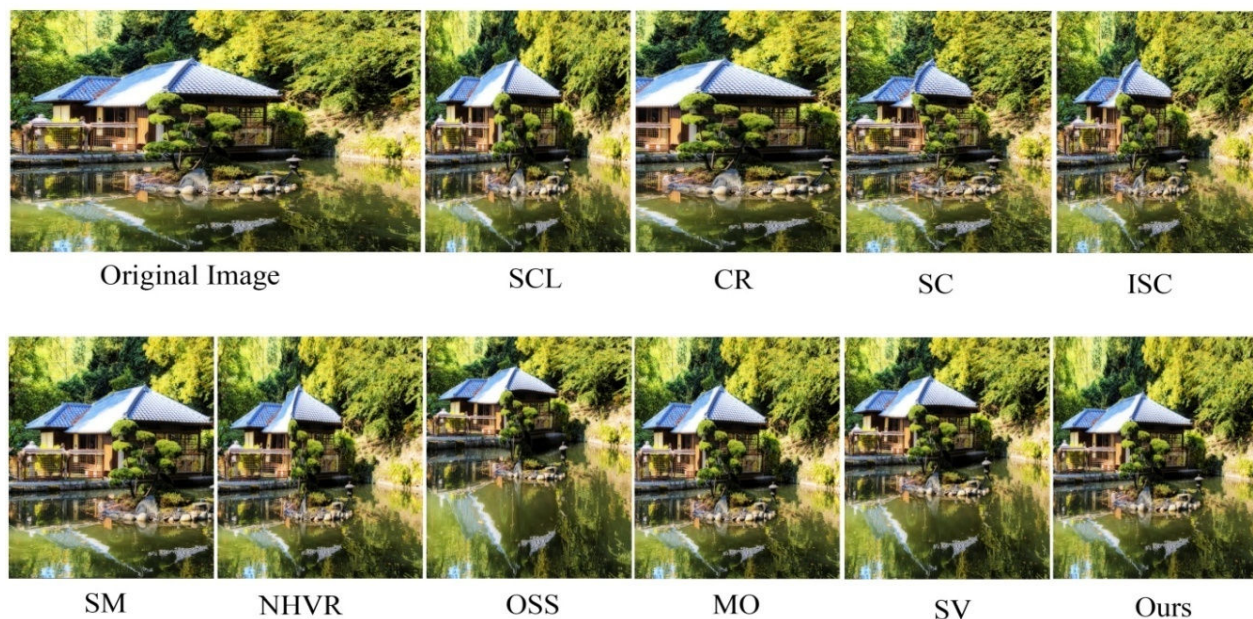
الگوریتم پیشنهادی وانگ [۷] که در قسمت (ج) شکل ۹ نشان داده شده است، هر چند که با اعمال محدودیت هایی در الگوریتم، باعث می شود که خطوط مستقیم عمودی و افقی کم تر دچار اعوجاج شوند، اما این روش نیز این کار را به طور کامل انجام نمی دهد. الگوریتم پیشنهادی برای حفظ نسبت طول و عرض محتوای تصویر از نواحی یکنواخت سرتاسر تصویر استفاده می کند و باعث می شود تصویر طبیعی تر به نظر برسد. برای جلوگیری از خم شدن خطوط مستقیم، الگوریتم پیشنهادی برای هر سطر و ستون از چهار گوش ها ضریب مقیاس گذاری یکسانی در نظر می گیرد. قسمت (ج) شکل ۹ با روش دانگ و قسمت (د) با روش پیشنهادی کوچک شده اند. برای مقایسه چگونگی حفظ خطوط مستقیم به نواحی داخل مستطیل نگاه کنید.

روش چند عملگری که توسط رابین اشتاین و همکاران معرفی شد [۹]، به دلیل استفاده از روش بریدن قسمت هایی از تصویر را حذف می کند. در شکل ۱۰ قسمت (ب) تصویر با این روش کوچک شده است. همان طور که در تصویر می بینید قسمت راست تصویر کاملاً حذف شده است.

به دلیل استفاده از روش SC، الگوریتم دانگ [۱۱] باعث اعوجاج در تصویر خروجی می شود. برای مثال به شکل ۱۰ قسمت (ج) نگاه کنید. همان طور که می بینید، قسمت سمت راست بدن خرس تخریب و باعث اعوجاج تصویر شده است.



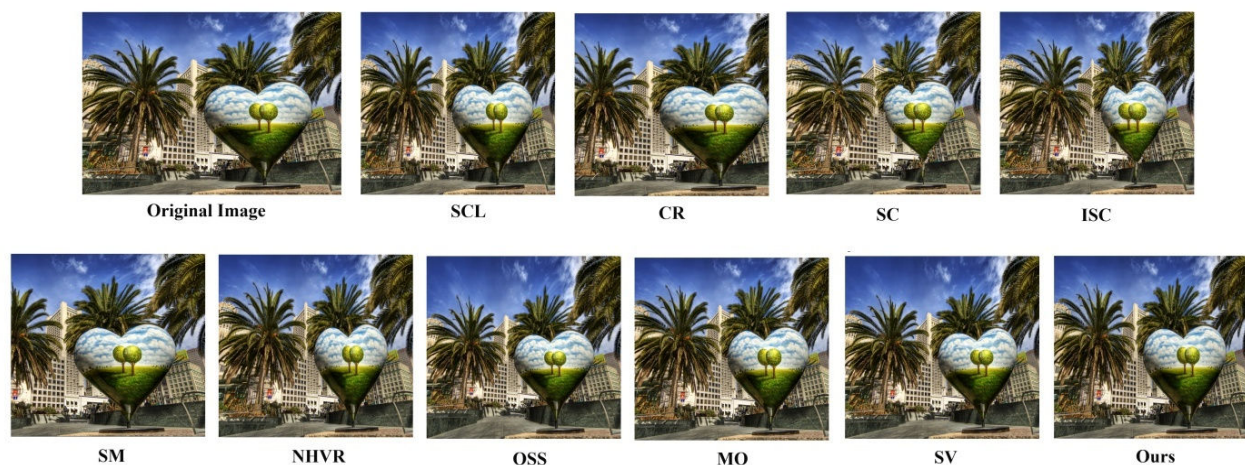
شکل ۱۰: مقایسه الگوریتم پیشنهادی با روش های چند عملگری؛ (الف) تصویر اصلی، (ب) روش رابین اشتاین [۹]، (ج) روش دانگ [۱۱] و (د) روش پیشنهادی. اندازه تصویر از ۳۱۸×۴۰۰ به اندازه ۲۰۰×۳۱۸ تغییر کرده است. این روش ها به دلیل استفاده از تابع فاصله و همچنین استفاده از عملگر SC بسیار کند هستند. شکل ۱۰ قسمت (د) با استفاده از روش



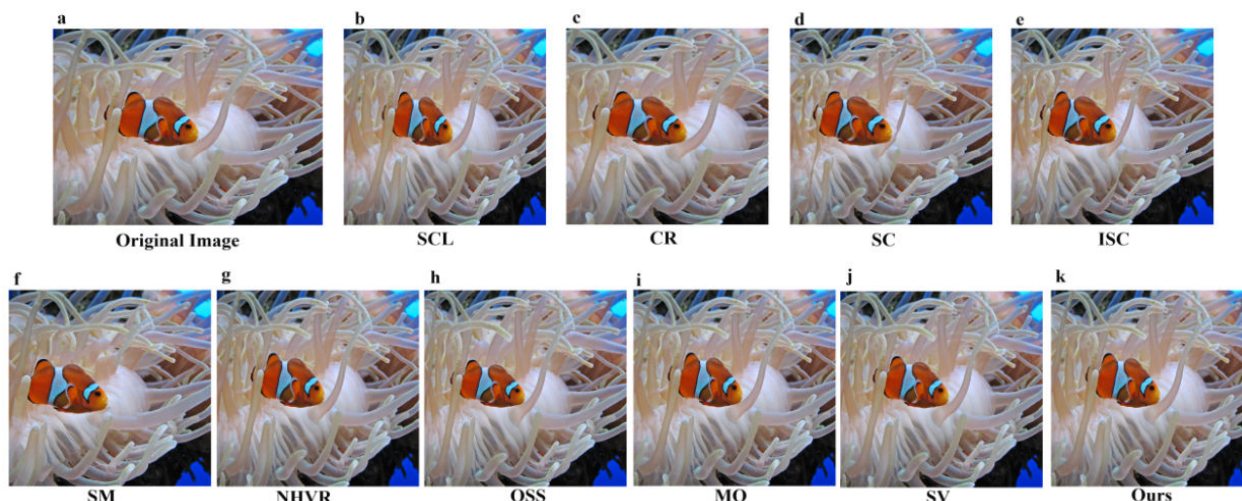
شکل ۱۱: مقایسه الگوریتم پیشنهادی با دیگر الگوریتم‌ها



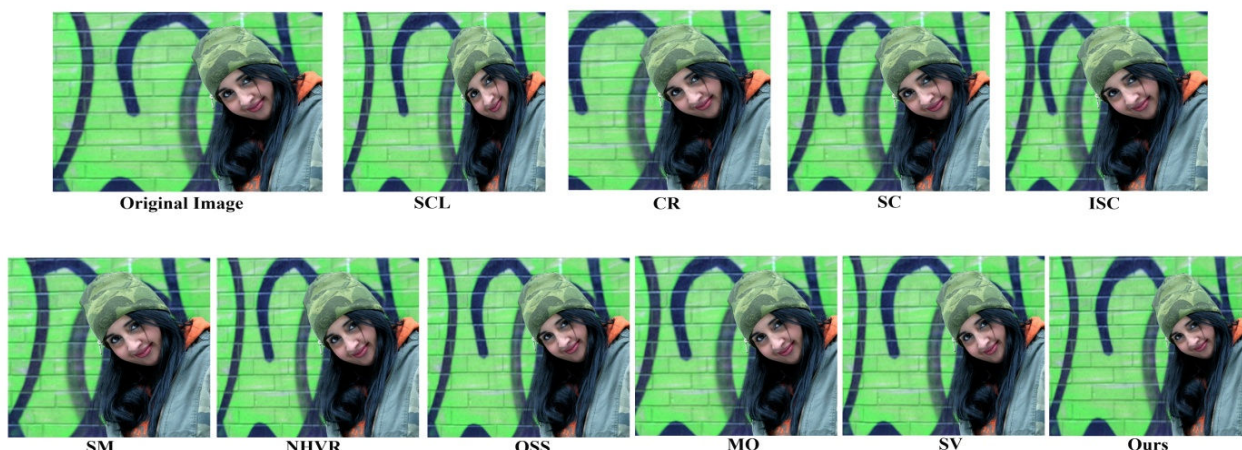
شکل ۱۲: مقایسه الگوریتم پیشنهادی با دیگر الگوریتم‌ها



شکل ۱۳: مقایسه الگوریتم پیشنهادی با دیگر الگوریتم‌ها



شکل ۱۴: مقایسه الگوریتم پیشنهادی با دیگر الگوریتم‌ها



شکل ۱۵: مقایسه الگوریتم پیشنهادی با دیگر الگوریتم‌ها

جدول ۱: فاصله تصویر خروجی حاصل از روش‌های مختلف از تصویر

ورودی

	شکل ۱۱	شکل ۱۲	شکل ۱۳	شکل ۱۴	شکل ۱۵
SCL	۲۰۳۵	۱۰۹۱	۲۴۰۷	۱۱۰۷	۸۸۵
CR	۲۱۴۹	۱۷۲۶	۲۲۱۴	۱۴۸۸	۸۹۴
SCL	۲۰۸۸	۱۲۱۹	۲۶۳۵	۱۱۵۹	۹۶۲
ISC	۲۰۱۸	۱۱۲۰	۲۵۸۵	۱۱۴۲	۸۹۴
SM	۱۸۷۹	۱۲۴۲	۲۳۳۳	۱۱۲۳	۸۹۹
NHVR	۱۹۳۱	۹۶۴	۲۴۱۸	۹۹۱	۷۹۶
OSS	۱۷۵۴	۱۳۶۹	۲۴۴۵	۱۰۲۳	۸۸۱
MO	۲۰۸۴	۱۶۶۳	۲۴۶۸	۱۱۹۸	۱۲۵۲
SV	۱۹۰۵	۱۵۳۳	۲۴۸۲	۱۱۳۰	۸۸۴
Ours	۱۹۵۴	۱۲۰۳	۲۳۹۷	۱۱۲۷	۸۸۰

موجود در تشخیص نواحی مهم تصویر دارد. این الگوریتم برای هر قسمت از تصویر یک ضریب اهمیت محاسبه کرده که نواحی مهم‌تر دارای ضریب بزرگ‌تری هستند و در هنگام تغییر اندازه تصویر این نواحی کم‌تر تخریب می‌شوند. چون کج شدن خطوط مستقیم عمودی و افقی تصویر، اعوجاج حاصل از تخریب آن را بیش‌تر نمایان می‌کند، ما هر سطر و ستون از چهارگوش‌ها را به یک اندازه تغییر داده و از خم شدن خطوط جلوگیری کردیم. سرانجام نتیجه شبیه‌سازی این الگوریتم و مقایسه با سایر الگوریتم‌ها نشان داد که الگوریتم پیشنهادی هم سرعت قابل‌توجهی دارد و هم در مقایسه با الگوریتم‌های موجود دارای ارزش کمی و کیفی بهتری است.

پیوست: معیار شباهت دوطرفه

فرض کنید تصویر S ، تصویر ورودی و تصویر T ، تصویر خروجی حاصل از اجرای الگوریتم تغییر اندازه تصویر با حفظ محتوا باشد. فرض کنید P یک قطعه 7×7 از تصویر ورودی بوده، هدف پیدا کردن یک قطعه از تصویر خروجی است که کم‌ترین فاصله 2 با تصویر ورودی داشته باشد و برعکس. اگر N_S تعداد قطعات تصویر ورودی (S) و N_T تعداد

۴- نتیجه‌گیری

در این مقاله یک روش تغییر اندازه تصویر با حفظ محتوا پیشنهاد شد که از توابع آماری برای محاسبه اهمیت هر قسمت از تصویر استفاده می‌کند. برای تشخیص نواحی با اهمیت تصویر، یک نقشه اهمیت جدید پیشنهاد دادیم که عملکرد بهتری نسبت به نقشه‌های اهمیت

- [14] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," *Proceedings of the NIPS*, 2006.
- [15] R. Rosenholtz, Y. Li, and L. Nakano, "Measuring visual clutter," *Journal of Vision*, vol. 7, no. 2, pp. 1–22, 2007.
- [16] P. Krähenbühl, Manuel Lang, A. Hornung, M. Gross, and D. Research, "A system for retargeting of streaming video," *ACM Transactions on Graphics*, vol. 28, no. 5, 2009.

زیرنویس‌ها

- ^۱ Importance map
- ^۲ Methodology
- ^۳ Measure
- ^۴ Blur
- ^۵ Execution time
- ^۶ Patch
- ^۷ Distance

قطعات تصویر خروجی (T) باشد، فاصله تصویر ورودی از تصویر خروجی با رابطه زیر به دست می‌آید [۱۰].

$$d(S, T) = \frac{1}{N_S} \sum_{P \in S} \min_{Q \in T} D(P, Q) + \frac{1}{N_T} \sum_{Q \in T} \min_{P \in S} D(Q, P)$$

برای مثال، برای هر قطعه $Q \subset T$ دنبال یک قطعه $P \subset S$ می‌گردیم که بیش‌ترین شباهت با آن داشته باشد و برعکس. تابع $D(.,.)$ فاصله بین دو قطعه در دو تصویر را محاسبه می‌کند. تابع $D(.,.)$ (Sum of Squared Distances) SSD دو قطعه از تصویر را محاسبه می‌کند. به‌طور خلاصه الگوریتم شباهت دوطرفه فاصله دو تصویر که ممکن است در اندازه متفاوت باشند را محاسبه می‌کند. تصویری که دارای فاصله کم‌تر باشد به این مفهوم است که آن تصویر به تصویر ورودی شباهت بیش‌تری دارد.

مراجع

- [1] D. Vaquero, M. Turk, K. Pulli, M. Tico, and N. Gelfand, "A survey of image retargeting techniques," *Proc. SPIE 7798, Applications of Digital Image Processing XXXIII*, 2010.
- [2] S. Avidan, and A. Shamir, "Seam carving for content-aware image resizing," *ACM Transactions on Graphics*, vol. 26, no. 3, pp. 267–276, 2007.
- [3] M. Rubinstein, A. Shamir, and S. Avidan, "Improved seam carving for video retargeting," *ACM Transactions on Graphics*, vol. 27, no. 3, pp. 1–9, 2008.
- [4] Y. Pritch, E. Kav-Venaki, and S. Peleg, "Shift-map image editing," *IEEE International Conference on Computer Vision*, pp. 151–158, 2009.
- [5] R. Gal, O. Sorkine, and D. Cohen-Or, "Feature aware texturing," *Proceedings of Euro graphics Symposium on Rendering*, pp. 297–303, 2006.
- [6] L. Wolf, M. Guttman, and D. Cohen-Or, "Non-homogeneous content-driven video retargeting," *IEEE International Conference on Computer Vision*, pp. 1–6, 2007.
- [7] Y.S. Wang, C.L. Tai, O. Sorkin, and T.Y. Lee, "Optimized scale-and-stretch for image resizing," *ACM Transactions on Graphics*, vol. 27, no. 5, 2008.
- [8] Y. Wu, X. Liu, S. Liu, and Kwan-Liu Ma, "ViSizer: a visualization resizing framework," *IEEE Transactions on Visualization and Computer Graphics*, pp. 278–290, 2013.
- [9] M. Rubinstein, A. Shamir, and S. Avidan, "Multi-operator media retargeting," *ACM Transactions on Graphics*, vol. 28, no. 3, pp. 1–11, 2009.
- [10] D. Simakov, Y. Caspi, E. Shechtman, and M. Irani, "Summarizing visual data using bidirectional similarity," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2008.
- [11] W. Dong, N. Zhou, J.C. Paul, and X. Zhang, "Optimized image resizing using seam carving and scaling," *ACM Transactions on Graphics*, vol. 28, no. 5, pp. 1–10, 2009.
- [12] Y. Liang, Z. Su, and X. Luo, "Patchwise scaling method for content-aware image resizing," *Signal Processing*, vol. 92, pp. 1243–1257, 2012.
- [13] L. Itti, C. Koth, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.