

فشرده‌سازی تصویر با کمک حذف و کدگذاری هوشمندانه اطلاعات تصویر و بازسازی آن با استفاده از الگوریتم‌های ترمیم تصویر

علی جمشیدی*، مهران یزدی و مریم‌السادات منافی
دانشکده مهندسی برق و کامپیوتر، دانشگاه شیراز، شیراز، ایران



چکیده

روش‌های فشرده‌سازی با اتلاف، به دلیل ایجاد فشرده‌سازی بیشتر، کاربرد گسترده‌تری دارند. اگرچه روش‌های زیادی تا به حال برای فشرده‌سازی تصاویر پیشنهاد شده، اما به استفاده از روش‌های هوشمندانه حذف اطلاعات، کمتر توجه شده است. ترمیم، مجموعه‌ای از روش‌هایی است که اصلاحاتی را بر روی تصاویر انجام می‌دهد؛ با این هدف که بیننده تفاوتی بین تصویر اصلاح‌شده و تصویر اصلی احساس نکند. در این مقاله، پس از بررسی و معرفی بعضی روش‌های ترمیم تصویر و روش‌های فشرده‌سازی تصویر با کمک ترمیم، روش جدیدی پیشنهاد می‌شود که علاوه بر این که باعث فشرده‌گی قابل توجه تصویر در زمان ارسال می‌شود، نتیجه کیفی مناسبی نیز در گیرنده خواهد داشت. در روش پیشنهادی، تصویر به نواحی ساختاری و بافتی تقسیم می‌شود و برای هر ناحیه بلوک‌های قابل حذفی که امکان بازسازی مناسبی در گیرنده با استفاده از روش‌های ترمیم دارند، شناسایی و حذف می‌شوند و اطلاعات کمکی لازم جهت ترمیم بهتر از آنها استخراج می‌شود. این بلوک‌ها به همراه بلوک‌های غیرقابل حذف تصویر پس از گذشتن ارسال می‌شوند و در گیرنده پس از کدگشایی، بلوک‌های از دست‌رفته بازسازی و ترمیم می‌شوند تا در نهایت تصویر اولیه در گیرنده قابل استفاده باشد. ویژگی‌های روش پیشنهادی نخست متغیر بودن اندازه بلوک‌های حذفی است که باعث فشرده‌گی بیشتر می‌شود و ثانیاً ارائه روش جدیدی جهت بازسازی بلوک‌های شامل لبه در گیرنده است که کیفیت بلوک‌های ترمیم‌شده این نواحی را افزایش می‌دهد.

واژگان کلیدی: ترمیم تصویر، فشرده‌سازی تصاویر، حذف هوشمندانه اطلاعات، کدکردن

Image Compression Based on Intelligent Information Removing and Inpainting Reconstruction Algorithms

Ali Jamshidi*, Mehran Yazdi & Maryam-ol-sadat Manafi

School of Electrical and Computer Engineering, University of Shiraz, Shiraz, Iran

Abstract

Compression can be done by lossy or lossless methods. The lossy methods have been used more widely than the lossless compression. Although, many methods for image compression have been proposed yet, the methods using intelligent skipping proper to the visual models has not been considered in the literature. Image inpainting refers to the application of sophisticated algorithms to replace lost or corrupted parts of the data so that visual difference cannot be inferred from the reconstructed image. In this paper, first we review some of the image inpainting algorithms and some of the image compression techniques using the inpainting algorithms, we propose a new inpainting based image compression algorithm that can improve the compression rate considerably. We present image compression system based on the proposed parameter-assistant image inpainting method to more deeply exploit visual redundancy inherent in color images. We have shown that with carefully selected dropped regions and appropriately extracted parameters from them, dropped regions can be satisfactorily restored using the proposed PAI algorithm. Accordingly, our compression scheme has a higher coding performance compared with traditional methods in terms of the perceptual quality. To best represent the target region for inpainting, an effective region classifier is required. A generic solution is to study the distribution of each image region and find the best match among the

* Corresponding author

* نویسنده عهده‌دار مکاتبات

candidates in the predefined model class. For simplicity, in our scheme, an entire image divided into three categories: gradated, structural, and non-featured, at non-overlapping block level of size $S \times S$. The classification is performed based on edge content and color variance in each block. Simulation results show that our proposed method has reasonable visual quality in comparison with the other proposed image compression algorithms.

Keywords: Image Inpainting, Image Compression, Intelligent Information Removing, Coding

خاصی حذف و بازسازی شده و بقیه بخش‌های تصویر نیز با معیارهای مخصوص به خود حذف و ترمیم می‌شوند [13]-[11]. برای اینکه کیفیت تصویر بازسازی‌شده در گیرنده قابل قبول و قابل مقایسه با روش‌هایی همچون JPEG باشد، اطلاعاتی از بلوک‌هایی که قرار است حذف شوند، از آنها استخراج و به‌همراه بقیه اطلاعات ارسال می‌کنند. حجم این اطلاعات در برابر حجم بلوک اصلی بسیار ناچیز است و اگر از روش‌های کدگذاری منبع، جهت فشرده‌کردن آنها استفاده شود، تأثیر چندانی بر فشرده‌گی نخواهند داشت. این در حالی است که کیفیت تصویر بازسازی‌شده را ارتقا می‌دهند. اینکه چه بلوک‌هایی از تصویر باید حذف شوند، چه نواحی‌ای از آنها به‌عنوان اطلاعات کمکی ارسال شود و همچنین در گیرنده چه روش ترمیمی باید مورد استفاده قرار گیرد تا نتایج بهتری در گیرنده داشته باشیم، مسائل چالش برانگیزی هستند که موضوع مقاله‌های بسیاری تاکنون بوده‌اند.

در سال‌های اخیر، پرکاربردترین معیار استفاده‌شده برای محاسبه عملی کیفیت تصاویر نسبت سیگنال به نوفه ($PSNR^2$) و میانگین مربع خطا (MSE^3) بوده‌اند که البته چون نظر مشاهده‌گر در تخمین کیفیت در آنها مطرح نبوده و تنها براساس مقایسه با تصویر مرجع اولیه بنا شده‌اند، مورد انتقاد قرار گرفته‌اند. در چند دهه اخیر، تلاش‌های بسیاری شده است که روش‌های جدیدی جهت سنجش کیفیت ادراکی تصویر براساس دستگاه بینایی انسان ارائه شود. در این روش‌ها، به این نکته توجه شده است که در کاربردهای عملی ما به تصویر اولیه دسترسی نداریم. در این مقاله، در کنار $PSNR$ ، از معیاری که نکات بالا را مورد توجه قرار داده است نیز استفاده می‌شود [14].

در سال ۲۰۰۰ الگوریتم‌هایی جهت ترمیم خودکار و دیجیتال تصاویر ارائه شده است [2]. در روش ارائه‌شده، پس از اینکه کاربر نواحی از تصویر را که باید ترمیم شوند، انتخاب کرد، الگوریتم به‌صورت خودکار با استفاده از اطلاعات نواحی

۱- مقدمه

در روش‌های متداول فشرده‌سازی تصاویر، همه بخش‌های تصویر که به‌طور معمول به‌صورت بلوک‌های مجزا در نظر گرفته می‌شوند، به‌صورت مشابه فشرده می‌شوند. به‌عنوان مثال قسمت عمده روش فشرده‌سازی JPEG اعمال تبدیل DCT^1 بر روی تک‌تک بلوک‌های 8×8 تصویر و ارسال ضرایب مؤثرتر این تبدیل و حذف ضرایب کم اهمیت است. فرآیند بالا به‌طور کاملاً مشابه در مورد تک‌تک بلوک‌ها اعمال می‌شود؛ همچنین، در سامانه‌های بازیابی تصویر بر اساس محتوا، ویژگی‌های دیداری تصاویر پایگاه استخراج‌شده و پایگاه ویژگی‌های دیداری برای یافتن نزدیک‌ترین تصاویر به تصویر پرس و جو، جستجو می‌شود [1].

با معرفی و پیشرفت روش‌هایی مانند ترمیم [2]، که امکان بازسازی نواحی مختلف تصویر را به ما می‌دهند، این ایده مطرح شد که در فرستنده، بخش‌هایی از تصویر که حاوی اطلاعات زیادی نیستند و امکان بازسازی را در گیرنده دارند، حذف و در گیرنده با استفاده از روش ترمیم مناسبی بازسازی شوند [3]. این روش ابتدا جهت بازیابی خودکار قسمت‌هایی از تصویر که به هر دلیلی از بین رفته بودند، مطرح شد [4]-[6]. درحالی‌که تا چند سال بعد کاربردهای زیاد و متنوعی برای آن مطرح و پیشنهاد شد و اهمیت توجه به این روش را افزایش داد. فشرده‌سازی تصاویر نیز از کاربردهای مطرح‌شده برای این روش است [7]-[11]. رفته‌رفته این ایده مورد توجه بیشتری قرار گرفت و نتایج قابل توجهی نیز ارائه شده است [7]-[11]. با توجه به این‌که همه قسمت‌های تصویر از نظر فشرده‌سازی و بازسازی درجه پیچیدگی یکسانی ندارند، برای به‌دست‌آوردن نتیجه بهتر، به‌طور معمول تصویر را به بخش‌هایی تقسیم می‌کنند که نقاط هر بخش ویژگی‌های مشترکی دارند و بازسازی آنها در گیرنده می‌تواند به‌صورت مشابه با هم صورت گیرد. به‌عنوان مثال، بخش‌هایی از تصویر که جزییات بیشتر و تغییرات سریع‌تری دارند و شامل لبه هستند، با معیارهای

² Peak Signal to Noise Ratio

³ Mean Square Error

¹ Discrete Cosine Transform

اینکه باعث فشردگی قابل توجه تصویر در زمان ارسال می‌شود، نتیجه کیفی مناسبی نیز در گیرنده خواهد داشت. تفاوت روش ارائه‌شده در این مقاله با دیگر روش‌ها از جمله مرجع [10] و [25] در این است که: ۱- در روش پیشنهاد شده و در قسمت تخمین بلوک‌های حذف‌شده، ابعاد بلوک‌ها، متغیر در نظر گرفته شده است. ۲- ارائه روش جدیدی جهت بازسازی بلوک‌های شامل لبه در گیرنده که کیفیت بلوک‌های ترمیم‌شده این نواحی را افزایش می‌دهد؛ درحقیقت فرآیند تعیین بلوک‌های لبه‌ای قابل حذف نیز متفاوت است ۳- تعیین بلوک‌های بدون لبه قابل حذف تصویر.

بخش‌های باقیمانده از مقاله به شرح زیر سامان‌دهی شده است. در بخش دوم ابتدا درخصوص روش‌های فشرده‌سازی تصویر مبتنی بر ترمیم بحث می‌کنیم؛ سپس، در بخش سوم، روش پیشنهادی را مطرح می‌کنیم. در بخش چهارم شبیه‌سازی‌ها و تحلیل نتایج آورده شده و درنهایت، بخش پنجم به نتیجه‌گیری اختصاص یافته است.

۲- فشرده‌سازی تصویر مبتنی بر الگوریتم‌های ترمیم تصویر

با توجه به وجود روش‌های ذکر شده در قسمت قبل و نیاز به فشرده‌سازی تصویر تا حد ممکن، به دلیل محدودیت‌های پهنای باند، این ایده ایجاد شد که بخش قابل توجهی از تصاویری که می‌خواهند ارسال شوند، حذف شده و در گیرنده به کمک ترمیم تصویر بازسازی شود [3]. از آن جا که روش‌های ترمیم تصویر قادرند لبه‌ها و بافت‌های از بین رفته را بازسازی کنند، ما می‌توانیم به دلخواه بعضی از بخش‌های تصویر، شامل ساختار یا بافت را حذف کنیم. پس از حذف عمدی این قسمت‌ها، می‌توان اطلاعات باقی‌مانده را به همراه ویژگی‌هایی از قسمت‌های به‌عمد حذف‌شده، کد و ارسال کرد. درنهایت در گیرنده پس از کدگشایی اطلاعات با کمک الگوریتم‌های مناسب ترمیم، ساختارها و بافت‌های از دست‌رفته را تخمین زده و بازسازی می‌کنیم. اگر اطلاعات به صورت بلوکی ارسال می‌شوند، حذف می‌تواند به صورت بلوکی صورت گیرد و در گیرنده بلوک‌های حذف‌شده بازسازی شوند. با این کار ضریب فشرده‌سازی تا حد زیادی می‌تواند بهبود یابد و در نتیجه نرخ انتقال اطلاعات به میزان مناسبی افزایش می‌یابد.

روش فشرده‌سازی با کمک ترمیم براساس لبه در [9]

همسایه، این نواحی را ترمیم می‌کند. به طور دقیق‌تر، در این الگوریتم، با استفاده از معادلات دیفرانسیل با مشتقات جزئی، اطلاعات نواحی همسایه در راستای امتدادشان به درون ناحیه از دست‌رفته کشیده می‌شوند. معادله، دیفرانسیلی که در گام نخست استفاده شد، لاپلاسیان^۱ بود؛ اما رفته‌رفته پژوهش‌گران دیگر از معادلات درجه بالاتری (تا درجه ۳ و ۴) برای بهتر شدن نتیجه استفاده کردند [17]-[16]. البته به دلیل استفاده از معادلات دیفرانسیل با مشتقات جزئی، این روش به محاسبات پیچیده‌تری نیاز دارد. محدودیت این روش در این است که اگر مساحت ناحیه از دست‌رفته زیاد باشد، نتیجه مناسبی نخواهیم داشت و لبه‌ها تار می‌شوند. روش‌هایی نیز مانند روش‌های مراجع [20]-[17] معرفی شدند که در همگی آنها با استفاده از معادلات دیفرانسیل با مشتقات جزئی به ترمیم تصویر پرداخته می‌شود. برای به دست آوردن نتایج بهتر، ایده تلفیق روش‌های موجود مورد توجه قرار گرفت [15]. در این ایده، تصویر به بخش‌های مجزایی تقسیم شده و برای هر بخش بررسی می‌شود که چه روشی آن بخش را بهتر ترمیم می‌کند. نتیجه این بررسی به تعریف مناطق ساختاری^۲ و مناطق بافتی^۳ در تصویر منجر شد. نواحی ساختاری، نواحی‌ای هستند که سطح رنگ تصویر (چه در تصویر رنگی و چه در تصویر خاکستری) در آنها تغییرات سریعی دارند؛ یعنی نقاطی شامل لبه، نواحی کم‌عرض و همسایه‌های آنها؛ اما، تغییرات مؤلفه‌های تصویر در نواحی بافتی، کند و یکنواخت است. با توجه به این تعریف، نواحی ساختاری به طور معمول نواحی باریک و کم‌عرض هستند و نواحی بافتی، سطوح وسیع‌تری از تصویر را به خود اختصاص می‌دهند.

در مرجع [24]، نویسندگان، یک بررسی اجمالی از روش‌های توأم پنهان‌سازی داده و روش‌های فشرده‌سازی با استفاده از ترمیم تصاویر را ارائه کرده‌اند.

در [25]، با تجزیه تصویر به نواحی بافتی و هندسی و سپس با استفاده از روش‌های ترمیم تصویر مبتنی بر لبه، فشرده‌سازی بخش‌های بافتی و هندسی به صورت جداگانه انجام می‌شود.

در این مقاله، پس از بررسی و معرفی بعضی روش‌های فشرده‌سازی تصویر با کمک ترمیم، که توسط دیگران به کار گرفته شده است، روش جدیدی پیشنهاد می‌شود که علاوه بر

¹ Laplacian

² Structural Regions

³ Texture Regions

ارائه شده است. در این روش، پس از استخراج لبه‌ها، انتخاب بلوک‌های الگو^۱ - یعنی بلوک‌هایی که از نوع ساختاری یا بافتی غیرقابل حذف هستند - براساس این لبه‌ها صورت می‌گیرد. برای این منظور، ابتدا تصویر به بلوک‌های ۸×۸ که با هم هم‌پوشانی ندارند، تقسیم می‌شود و به هر بلوک براساس فاصله‌اش تا لبه‌ها برچسب ساختار یا بافتی اختصاص داده می‌شود. به صورت جزئی‌تر، اگر بیش از یک چهارم نقاط یک بلوک فاصله کمی (۵ نقطه) تا لبه‌ها داشته باشند، بلوک مورد نظر برچسب ساختاری و در غیر این صورت برچسب بافتی می‌خورد.

در تصویر شکل (۱) لبه‌ها قبلاً مشخص و با خطوط خاکستری تیره مشخص شده‌اند. بلوک‌های خاکستری روشن ساختار و بلوک‌های سفید و سیاه بافت هستند. بلوک‌هایی که در حاشیه ناحیه بافتی (شامل نواحی سفید و سیاه) قرار دارند، یعنی بلوک‌های سفید بافت‌های ضروری هستند که در همسایگی عمودی یا افقی بلوک‌های ساختاری قرار دارند. این بلوک‌ها چون جزء ناحیه‌های گذار هستند، قابل حذف نمی‌باشند. از بین بلوک‌های سیاه تعدادی به عنوان بافت اضافه (به منظور ارتقای کیفیت دیداری در بازسازی) و به عنوان نماینده‌ای از تغییرات واریانس در نواحی بافتی - که تغییرات در آنها کند است - نگه داشته می‌شوند و بقیه حذف می‌شوند.



(شکل-۱): بلوک‌بندی یک تصویر نمونه جهت توضیح بلوک‌های

غیر قابل حذف ساختاری و بافتی [9]

(Figure-1): An exemplar block selection to describe necessary blocks for textural and structural region [9]

در مرجع [10] فشرده سازی تصویر با کمک ترمیم پارامتری (PAI^۲) ارائه شده است. در این روش، ابتدا تصویر به

بلوک‌های ۸×۸ بدون هم‌پوشانی با یکدیگر تقسیم می‌شود و بعد به هر بلوک آن بر حسب ویژگی‌های تصویر یکی از برچسب‌های گرادینانی، ساختاری و غیر خصیصه‌ای^۳ اختصاص می‌یابد. برچسب اختصاص داده شده به هر بلوک براساس لبه‌های موجود در آن و واریانس رنگی آن صورت می‌گیرد. ابتدا فرآیند استخراج لبه‌ها در تصویر صورت می‌گیرد و هر بلوکی که شامل لبه بود برچسب بلوک ساختاری می‌گیرد؛ سپس برای بقیه بلوک‌ها واریانس رنگی بر مبنای مؤلفه‌های رنگی RGB محاسبه می‌شود. بلوک‌هایی که واریانس آنها کمتر از یک مقدار آستانه باشد، برچسب بلوک گرادینانی و بلوک‌های باقی‌مانده برچسب غیرخصیصه‌ای می‌گیرند. برای نواحی در تصویر که بلوک‌های برچسب گرادینانی دارند، رنگ یا شدت روشنایی به طور یکنواخت و کند تغییر می‌کند؛ بنابراین، مهم‌ترین ویژگی که باید استخراج شود، الگوی گرادینانی در این نواحی است که با استفاده از گرادینان بلوک‌ها شکل می‌گیرد. برای هر بلوک بردار گرادینانی برای هر یک از مؤلفه‌های RGB مستقلاً به دست می‌آید.

برای بازسازی بلوک‌های گرادینانی، روش PAI یک بلوک گرادینانی حذف‌شده را با استفاده از اطلاعات نقاط معلوم همسایه و با کمک گرادینان آن بلوک که در فرستنده محاسبه و ارسال شده است، ترمیم و بازسازی می‌کند. در این روش، برای یک ناحیه شامل بلوک‌های گرادینانی ابتدا بلوک‌های با بیشترین همسایه معلوم (از بین چهار همسایگی ممکن) بازسازی می‌شوند، تا در نهایت نوبت به بلوک‌های با همسایه‌های کمتر برسد.

برای بلوک‌هایی از تصویر که برچسب ساختاری دارند و درون آنها لبه وجود دارد، محل نقاطی که لبه‌ها را می‌سازند و گرادینان لبه‌ای یعنی گرادینان در طول لبه پارامترهای کلیدی هستند. محل لبه‌ها در طول فرایند طبقه‌بندی بلوک‌های تصویر، به دست آمده است. بازسازی بخش‌های ساختاری حذف‌شده از تصویر در گیرنده به همان روش مرجع [9] صورت می‌گیرد. بلوک‌های الگو شامل بلوک‌های پیچیده و بلوک‌های الگوی ساختاری و گرادینانی، که باید خودشان به طور کامل از سال شوند، به روش JPEG فشرده می‌شوند. پارامترهای کمی شامل محل بلوک‌هایی که حذف شده‌اند، نوع بلوک حذفی (ساختاری یا گرادینانی) و محل لبه‌ها در بلوک‌های ساختاری که باید حذف شوند

³ Non-Feature

¹ Exemplar Blocks

² Parameter-Assistant Inpainting

۱- اطلاعات دریافت شده کدگشایی می‌شوند. به این صورت که کدگشای یک بلوک‌های حذف‌نشده تصویر را که تنها کدگذاری JPEG روی آنها اعمال شده کدگشایی کرده و کدگشای ۲ اطلاعات کمکی بلوک‌های حذف‌شده تصویر را کدگشایی می‌کند.

۲- پس از کدگشایی اطلاعات کمکی بلوک‌های حذف شده، این بلوک‌ها ترمیم و بازسازی می‌شوند.

۳- در مرحله آخر این بلوک‌های باز سازی شده با بلوک‌های اصلی تصویر ترکیب می‌شوند و در کنار هم قرار می‌گیرند تا تصویر نهایی قابل استفاده توسط کاربر به‌دست آید.

در زیربخش‌های بعدی در مورد هر یک از قسمت‌های بالا به تفصیل توضیح داده می‌شود و در بخش بعد نتایج ارائه خواهد شد.

۳-۱- فرستنده

برخلاف روش‌هایی که در مقاله‌های قبلی توضیح داده شد، در این روش ابعاد بلوک‌هایی که حذف می‌شوند، متغیر هستند و موقع فشرده‌سازی یک تصویر بلوک‌های حذفی می‌توانند 8×8 ، 16×16 یا 32×32 باشند. حال به توضیح روش پیشنهادی می‌پردازیم. در مرحله نخست با استفاده از روش 'canny' محل لبه‌های تصویر به‌دست می‌آیند. بعد از معلوم شدن لبه‌ها ابتدا تمام بلوک‌های لبه‌ای 8×8 ، 16×16 و 32×32 که باید حذف شوند، به صورت زیر مشخص می‌شوند و در مرحله بعد به تعیین بلوک‌های گرادیانی قابل حذف می‌پردازیم.

بصورت نقشه بیتی^۱ به روش JBIG^۲ فشرده می‌شوند. گرادیان بلوک‌ها نیز با روشی شبیه^۳ DPCM فشرده می‌شوند.

۳- روش پیشنهادی

در شکل (۲) نمودار جعبه‌ای کلی روش پیشنهادی رسم شده است که در ادامه به بررسی بیشتر آن می‌پردازیم.

در بخش تحلیل تصویر مشخص می‌کنیم، کدام نواحی تصویر باید حذف شوند و کدام نواحی غیرقابل حذف هستند. پس از مشخص شدن بلوک‌هایی که باید حذف شوند، اطلاعاتی که باید از طریق فرستنده ارسال شوند به چند گروه تقسیم می‌شوند:

۱- قسمت‌هایی از تصویر که حذف نمی‌شوند و خودشان باید ارسال شوند. این اطلاعات وارد کدکننده ۱ می‌شوند.

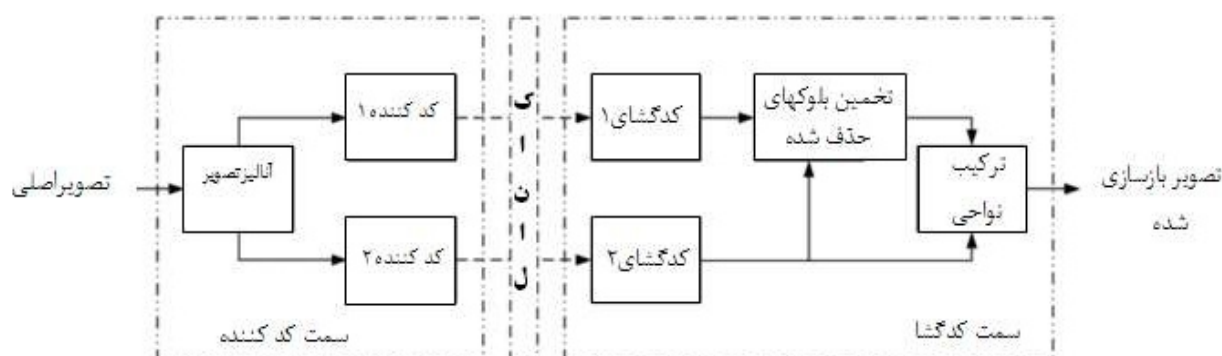
۲- اطلاعات کمکی بلوک‌های حذف‌شده که وارد کدکننده ۲ می‌شوند و شامل موارد زیر هستند:

■ بلوک‌هایی از تصویر که شامل لبه بوده و شرایط لازم برای حذف را داشته و در نتیجه تنها محل لبه‌ها در آنها باید ارسال شود.

■ بلوک‌هایی از تصویر که بدون لبه هستند و شرایط لازم را برای حذف دارند و گرادیان آنها باید ارسال شود.

■ محل بلوک‌هایی از تصویر که حذف شده‌اند.

پس از ارسال این اطلاعات کد شده از طریق کانال به گیرنده می‌رسیم که در آن طی مراحل زیر تصویر دریافت شده بازسازی می‌شود:



(شکل-۲): نمودار جعبه‌ای روش پیشنهادی
(Figure-2): Our Proposed Block Diagram

¹ Bitmap

² Joint Bi-level Image Experts Group

³ Differential Pulse-Code Modulation

۳-۱-۱- چگونگی به دست آوردن لبه‌ها در تصویر

هر یک از مؤلفه‌های RGB لبه‌های مخصوص خودشان را دارند که می‌توانند مشترک یا متمایز باشند؛ اما اگر بخواهیم این لبه‌های متمایز را در نظر بگیریم، برای هر یک از مؤلفه‌ها، بلوک‌های حذفی را باید جداگانه پیدا کنیم؛ در حالی که اگر تنها یک دسته لبه را طوری به دست آوریم که جمع لبه‌های هر سه مؤلفه باشد، برای هر سه مؤلفه بلوک‌ها یکسان حذف می‌شوند و به جای ارسال محل لبه‌ها در بلوک‌های حذفی و نقشه محل بلوک‌های حذفی برای هر یک از مؤلفه‌ها به‌طور جداگانه، این اطلاعات به خاطر مشترک بودن در هر سه مؤلفه، تنها باید یک بار ارسال شوند و در نتیجه اطلاعات از سالی به میزان زیادی کاهش می‌یابند. اگر برای پیدا کردن لبه‌ها از مقیاس خاکستری تصویر اصلی استفاده کنیم، جامعیتی که در بالا به آن اشاره شد، حفظ نمی‌شود و این امکان وجود دارد که در جایی از تصویر تنها یکی از مؤلفه‌ها لبه‌ای داشته باشد؛ در حالی که دو مؤلفه دیگر فاقد لبه باشند و این لبه در مقیاس خاکستری 'canny' آن قدر ضعیف باشد که در فرآیند لبه‌یابی پیدا نشود. برای حل این مشکل ضرایب مؤلفه‌ها در فرآیند محاسبه مقیاس خاکستری استاندارد آن قدر با سعی و خطا تغییر داده شدند تا نتیجه قابل قبولی به دست آمد. برای این منظور، ابتدا چندین بلوک دارای لبه مشخص به‌عنوان داده‌های آموزشی انتخاب شدند، به طوری که مؤلفه‌های R، G و B آنها دارای لبه‌های یکسانی باشند؛ سپس مقادیر اولیه ضرایب تبدیل مشابه نرم‌افزار متلب تعیین شدند و بعد در هر مرحله بهینه‌سازی لبه‌های به دست آمده از الگوریتم لبه‌یابی تصویر خاکستری با لبه‌های مشخص تعیین شده و مشخص بلوک‌ها مقایسه شده و در صورت وجود خطا ابتدا ضریب را کاهش یا افزایش می‌دهیم تا اینکه خطای کوچکی حاصل شود. رابطه ضرایب در فرم مقیاس خاکستری تعریف شده در متلب به صورت زیر است:

$$0.2989 \times R + 0.5870 \times G + 0.1140 \times B$$

که پس از بهینه‌سازی ضرایب زیر به دست می‌آیند:

$$0.2 \times R + 0.6 \times G + 0.2 \times B$$

با استفاده از این ضرایب در به دست آوردن فرم مقیاس خاکستری تصویر جهت پیدا کردن لبه‌ها به نتیجه مناسب و قابل قبولی برای هر سه مؤلفه می‌رسیم.

۳-۱-۲- تعیین بلوک‌های لبه‌ای قابل حذف تصویر

برای تعیین بلوک‌های لبه‌ای قابل حذف، تصویر در سه مرحله بررسی می‌شود: در مرحله نخست بلوک‌های 32×32 و در مرحله دوم بلوک‌های 16×16 و در مرحله سوم بلوک‌های 8×8 قابل حذف مشخص می‌شوند. حال به توضیح مرحله نخست می‌پردازیم. در این مرحله، تصویر به بلوک‌های 32×32 که با هم هم‌پوشانی ندارند، تقسیم می‌شود و برای هر بلوک در صورتی که شامل لبه باشد، صلاحیت آن برای حذف شدن مورد بررسی قرار می‌گیرد؛ یعنی خصوصیتی از آن بررسی می‌شود که نشان می‌دهند این بلوک در صورت حذف شدن پس از بازسازی کیفیت مناسبی خواهد داشت یا خیر؟ در گام نخست تمام دسته لبه‌های موجود در این بلوک برجسته زده می‌شوند و اگر تنها یک لبه در این بلوک موجود باشد، بقیه شروط برای حذف این لبه بررسی خواهد شد؛ در غیر این صورت به سراغ پردازش بلوک 32×32 بعد می‌رویم و شرایط حذف شدن زیر بلوک‌های این بلوک را در مراحل بعد بررسی می‌کنیم.

گام دوم، بررسی این نکته است که تک‌لبه موجود در این بلوک، فقط و فقط دو بار دو حاشیه متفاوت را از بین چهار حاشیه موجود در این بلوک قطع کرده باشد. به‌عنوان مثال در شکل (۳) بلوک سمت چپ به این دلیل که تک‌لبه موجود در بلوک تنها یک بار یکی از حاشیه‌های بلوک را قطع می‌کند، و بلوک سمت راست به این دلیل که لبه در هر دو بار تنها یک حاشیه بلوک را قطع کرده، شرایط حذف شدن را ندارند. دلیل قراردادن این شروط به روش ترمیم تصویر در گیرنده باز می‌گردد که براساس الگو است و الگوی این نوع لبه‌ها برای بقیه نقاط بلوک نمی‌تواند، استفاده شود؛ اما این شرایط هنوز برای حذف بلوک کافی نیستند. به‌عنوان مثال دو ویژگی بالا برای بلوک شکل (۴) صادق است؛ اما همان‌طور که می‌بینید، این لبه بسیار پیچیده است و مسلماً در صورت حذف آن، ترمیم در گیرنده نتیجه قابل قبولی نخواهد داشت.

برای جلوگیری از حذف چنین بلوک‌هایی در فرستنده، تمام زیر بلوک‌های هر بلوک را نیز بررسی می‌کنیم. در صورتی که هر یک از این زیر بلوک‌ها یکی از دو حالت زیر را داشته باشند، بلوک اصلی قابل حذف است:

الف: زیر بلوک موجود دو ویژگی بالا را داشته باشد؛ یعنی تنها شامل یک لبه و این تک‌لبه فقط و فقط دو بار دو حاشیه متفاوت از بین چهار حاشیه موجود در بلوک را قطع کرده باشد.

ب: این زیر بلوک فاقد لبه باشد.

بعد از اینکه تمامی بلوک‌های 32×32 بررسی شدند، نوبت به مرحله دوم بررسی بلوک‌های لبه‌ای قابل حذف می‌رسد. در این مرحله، تصویر را به بلوک‌های 16×16 تقسیم می‌کنیم و برای بلوک‌های شامل لبه‌ای که نخست این که هیچ یک از نقاط آنها در مرحله قبل برچسب حذفی نخورده‌اند و دوم این که بخشی از آنها جزء حاشیه بلوک‌های حذف‌شده در مرحله قبل نیست، حذفی بودن یا نبودن آن بررسی می‌شود. نحوه بررسی مشابه بررسی بلوک‌های 32×32 است. به همین ترتیب بلوک‌های 8×8 را بررسی می‌کنیم.

۳-۱-۳- تعیین بلوک‌های بدون لبه قابل حذف تصویر
 بعد از اینکه تمام بلوک‌های لبه‌ای قابل حذف 32×32 ، 16×16 و 8×8 تصویر مشخص شدند، نوبت به مشخص کردن بلوک‌های بدون لبه قابل حذف تصویر می‌رسد. در این جا هم در مرحله نخست بلوک‌های 32×32 ، در مرحله دوم بلوک‌های 16×16 و در نهایت بلوک‌های 8×8 قابل حذف مشخص می‌شوند. برای این منظور ابتدا تصویر را به بلوک‌های 32×32 بدون هم‌پوشانی تقسیم و برای هر بلوک بدون لبه، واریانس آن را با استفاده از رابطه زیر محاسبه می‌کنیم:

$$V = \sum_{y=1}^S \sum_{x=1}^S [(R_{xy} - \bar{R}_{xy})^2 + (G_{xy} - \bar{G}_{xy})^2 + (B_{xy} - \bar{B}_{xy})^2] \quad (1)$$

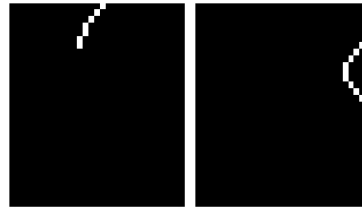
$$\bar{R} = \frac{1}{S^2} \sum_{y=1}^S \sum_{x=1}^S R_{xy} \quad (2)$$

$$\bar{G} = \frac{1}{S^2} \sum_{y=1}^S \sum_{x=1}^S G_{xy} \quad (3)$$

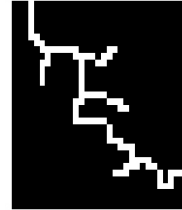
$$\bar{B} = \frac{1}{S^2} \sum_{y=1}^S \sum_{x=1}^S B_{xy} \quad (4)$$

که در آن S ابعاد بلوک و برابر 32 و RGB مؤلفه‌های رنگی و \bar{R} و \bar{G} و \bar{B} میانگین مؤلفه‌ها در هر بلوک هستند. همچنین، x و y نیز اندیس نقاط هستند. حال واریانس به دست آمده را با واریانس آستانه بلوک 32×32 مقایسه می‌کنیم. مقادیر آستانه برای بلوک‌های با ابعاد مختلف متفاوت است؛ ولی این مقادیر با هم رابطه دارند. به صورت تجربی مقدار آستانه بلوک‌های 16×16 برابر بیست‌هزار در نظر گرفته می‌شود و با توجه به این مقدار آستانه بلوک‌های $S \times S$ از رابطه زیر محاسبه می‌شود:

$$V_T = \frac{(20000 \times S^2)}{16^2} \quad (5)$$



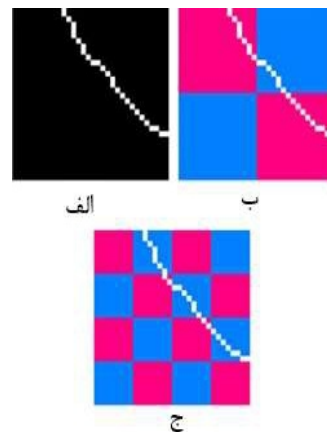
(شکل-۳): بلوک‌های غیر قابل حذف در یک تصویر نمونه
 (Figure-3): An example of necessary blocks



(شکل-۴): بلوک‌های قابل حذف در یک تصویر نمونه [9]
 (Figure-4): An example of removed blocks [9]

برای یک بلوک تقسیم بندی به زیربلوک‌ها به صورتی که در شکل (۵) دیده می‌شود، تعریف می‌شوند. در نهایت، بلوک اولیه‌ای که هر یک از زیر بلوک‌هایش یکی از دو حالت گفته شده در بالا را داشته باشد، به عنوان بلوک حذفی در نظر گرفته می‌شود و محل لبه‌ها در آن به عنوان اطلاعات کمکی نگه داشته می‌شوند.

با توجه به اینکه برای بازسازی بلوک‌های شامل لبه‌ای که حذف شده‌اند به تمام نقاط چهار حاشیه معلوم آنها نیاز داریم، اگر بلوکی شرایط حذف شدن را داشت؛ تمام زیربلوک‌های آن برچسب غیر حذفی می‌خورند تا مراحل بعدی پردازش که بلوک‌های کوچک‌تر بررسی می‌شوند با آنها کاری نداشته باشیم. البته در مورد بلوک‌های گردایانی که قرار است حذف شوند، این کار لازم نیست.



(شکل-۵): مراحل تقسیم بندی یک بلوک به زیربلوک‌های کوچک‌تر، الف: یک بلوک بزرگ، ب: تقسیم به ۴ زیربلوک ج: تقسیم به ۱۶ زیربلوک
 (Figure-5): The procedure of dividing a large block to small sub blocks, a) a large block, b) dividing to 4 sub blocks, c) dividing to 16 sub blocks

روش جهت تخمین مؤلفه‌های گرادیانی هر بلوک به شرح زیر است:

$$\hat{g}_y = \frac{\sum_{i=1}^S \left(i - \frac{S+1}{2}\right) (U_{i,V} - \bar{U}_V) / \sum_{i=1}^S \left(i - \frac{S+1}{2}\right)^2}{1} \quad (8)$$

$$\hat{g}_x = \frac{\sum_{i=1}^S \left(i - \frac{S+1}{2}\right) (U_{i,H} - \bar{U}_H) / \sum_{i=1}^S \left(i - \frac{S+1}{2}\right)^2}{1} \quad (9)$$

که در آن \bar{U}_V و \bar{U}_H میانگین بردارهای H و V هستند. برای هر بلوک بردار گرادیانی (\hat{g}_x, \hat{g}_y) برای هر یک از مؤلفه‌های RGB مستقلاً به دست می‌آید. حال به ادامه بررسی روش ارائه شده می‌پردازیم.

بعد از اینکه تمامی بلوک‌های 32×32 بررسی شدند، نوبت به مرحله دوم بررسی بلوک‌های بدون لبه قابل حذف می‌رسد. در این مرحله، تصویر را به بلوک‌های 16×16 تقسیم می‌کنیم و حذفی بودن یا نبودن بلوک‌های بدون لبه‌ای که هیچ یک از نقاط آنها در مراحل قبل برچسب حذفی نخورده‌اند، بررسی می‌شود. نحوه بررسی مشابه بررسی بلوک‌های 32×32 است و مقدار واریانس و آستانه آن از روابط (۱) و (۵) به دست می‌آید که در آنها S برابر ۱۶ است. در صورتی که بلوکی 16×16 در این مرحله قابل حذف تشخیص داده شد، گرادیان هر یک از مؤلفه‌های آن در جهت‌های x و y آن با استفاده از روابط (۸) و (۹) محاسبه شده و به عنوان اطلاعات کمکی ارسال می‌شود.

برای به دست آوردن نتیجه بهتر در بازسازی بلوک‌های گرادیانی در گیرنده، بلوک‌های گرادیانی که شرایط حذف داشته‌اند، اما یکی از هشت همسایگی آنها بلوک لبه‌ای است، جزء بلوک‌های غیر قابل حذف در نظر می‌گیریم. با قراردادن این شرط، همسایه‌های بلوک‌های لبه‌ای قابل حذف جزء بلوک‌های ارسالی و غیر قابل حذف قرار می‌گیرند و در ترمیم این بلوک‌ها هر چهار همسایه موجود هستند. نکته مهمی که درباره ایده پیشنهاد شده برای متغیر بودن ابعاد بلوک‌ها وجود دارد، این است که ممکن است، بلوکی شامل لبه پیچیده‌ای باشد؛ اما زیر بلوک‌های لبه‌ساده داشته باشند یا اصلاً لبه نداشته باشند و واریانس آنها از آستانه کمتر باشد و در نتیجه قابل حذف باشند. بنابراین، کار میزان فشرده‌سازی در تصویر بیشتر می‌شود.

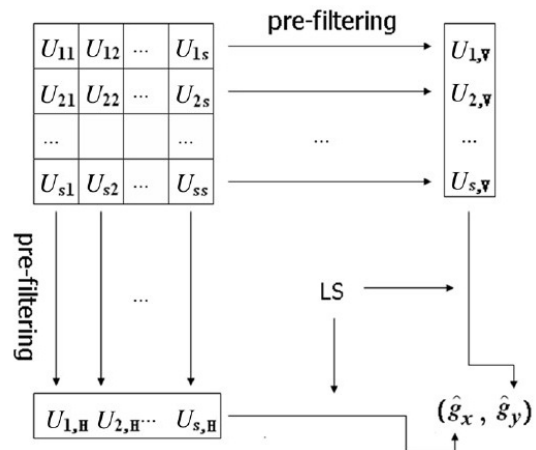
در مورد بلوک‌های بدون لبه‌ای هم که واریانس آنها از آستانه بیشتر است و در نتیجه قابل حذف نیستند، ممکن است،

در صورتی که واریانس بلوک از مقدار آستانه کمتر باشد، این بلوک، به عنوان بلوک گرادیانی 32×32 در نظر گرفته می‌شود و باید حذف شود. مقدار گرادیان هر یک از مؤلفه‌های این بلوک در جهت‌های x و y محاسبه می‌شود تا به عنوان اطلاعات کمکی ارسال شود. در غیر این صورت این بلوک نگه داشته می‌شود تا در مرحله بعدی پردازش درباره زیر بلوک‌های 16×16 آن یا زیر بلوک‌های 8×8 تصمیم گرفته شود.

در ادامه نحوه محاسبه گرادیان در بلوک‌های بدون لبه را توضیح می‌دهیم. برای این منظور، با توجه به شکل ۶ ابتدا برای هر بلوک میانگین اندازه نقاط را برای هر سطر و ستون محاسبه می‌کنیم و سپس، بردارهای U و H را طبق رابطه زیر می‌سازیم:

$$V = (U_{1,V}, \dots, U_{S,V}), U_{i,V} = \frac{1}{S} \sum_{x=1}^S U_{x,V} \quad (6)$$

$$H = (U_{1,H}, \dots, U_{S,H}), U_{i,H} = \frac{1}{S} \sum_{y=1}^S U_{y,H}, i = 1, \dots, S \quad (7)$$



(شکل-۶): چگونگی محاسبه گرادیان در بلوک‌های بدون لبه

[10]
(Figure-6): Gradient evaluation in the edge blocks [10]

حال با استفاده از روش برازش LS^1 مؤلفه‌های گرادیانی هر بلوک را در جهت‌های x و y تخمین می‌زنیم، در این روش، بهترین مدل برازش شده بر مجموعه‌ای از داده‌ها، مدلی است که در آن مجموع مربع باقی‌مانده‌ها کمینه باشد. منظور از باقی‌مانده‌ها، اختلاف بین داده مشاهده شده و مقداری است که از مدل به دست می‌آید. روابط حاصل از این

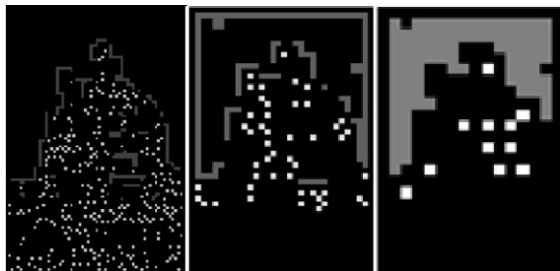
¹ Least Squares

مؤلفه‌های RGB در دو جهت x و y به گرادینان نیاز داریم؛ یعنی برای هر بلوک شش مقدار گرادینانی باید ارسال شوند.

با توجه به اینکه پراکندگی مقادیر گرادینانی بسیار کمتر از پراکندگی مقادیر مؤلفه‌ها است، برای کاهش تعداد بیت لازم جهت ارسال گرادینان‌های هر بلوک از همبستگی این بلوک‌ها با هم استفاده می‌کنیم. برای این منظور پس از چندی‌سازی^۱ اندازه گرادینان‌ها، ابتدا کدگذاری DPCM^۲ و سپس کدگذاری هافمن بر روی این مقادیر چندی‌سازی شده اعمال می‌شود.

(پ) برای کد کردن محل بلوک‌ها، برای هر یک از ابعاد بلوک مطابق شکل (۸) یک نقشه در نظر می‌گیریم.

برای بلوک‌های گرادینانی و لبه‌ای در نقشه مقدار غیر صفر در نظر می‌گیریم. حال با استفاده از روش Run Length هر نقشه را کد می‌کنیم با این تفاوت که در کد کردن بلوک‌های شامل لبه، نمادهای ارسالی تنها صفر و یک بودند و برای ارسال هر یک از نمادها تنها به یک بیت نیاز داشتیم؛ به عبارتی کدگذاری دو سطحی بود؛ اما در اینجا بلوک‌های حذفی در هر یک از نقشه‌ها خود دو نوع لبه‌ای و غیرلبه‌ای هستند و بخش‌های دیگر نقشه هم مقدار صفر را دارند. بنابراین در هر نقشه سه نماد داریم و کدگذاری ما، سه سطحی است. در نتیجه برای کد کردن هر یک از نمادها به دو بیت نیاز داریم.



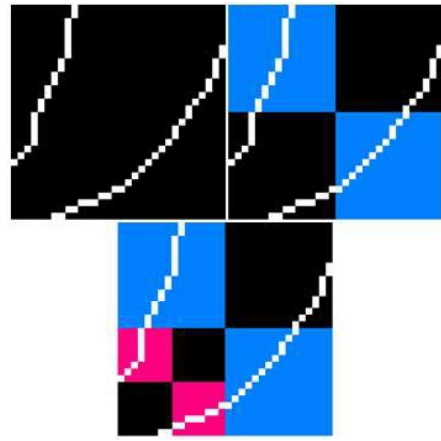
(شکل-۸): نقشه بلوک‌های حذفی 32×32 (راست)، 16×16 (وسط)،

و 8×8 (چپ)

(Figure-8): 32×32 skipped blocks map (right), 16×16 (center), and 8×8 (left)

۳-۲- گیرنده

در گیرنده ابتدا اطلاعات دریافت شده کدگشایی می‌شوند. به این صورت که بلوک‌های حذف‌نشده تصویر که تنها



(شکل-۷): مثالی از بلوک لبه‌ای غیرقابل حذف با زیربلوک‌های لبه‌ای و گرادینانی قابل حذف

(Figure-7): An example of necessary edge block with edge and gradient sub blocks

تغییرات مؤلفه‌های RGB در زیر بلوک‌ها کمتر باشد. بنابراین، ممکن است واریانس آنها کمتر از مقدار آستانه شده و این زیر بلوک‌ها قابل حذف شوند. مطالب بالا در شکل (۷) نشان داده شده‌اند. به عنوان مثال با تقسیم بلوک 32×32 بالا سمت چپ-که خود شرایط حذف را ندارد- به بلوک‌های 16×16 تنها بلوک پایین سمت چپ شرایط حذف را ندارد و سه بلوک دیگر می‌توانند حذف شوند. این بلوک هم در صورتی که به زیربلوک‌های 8×8 تقسیم شود، زیربلوک‌های آن به صورت جداگانه به عنوان بلوک‌های 8×8 قابل حذف می‌شوند.

۳-۱-۴- نحوه کد کردن اطلاعات ارسالی

با توجه به اینکه هدف ما فشرده کردن تصویر و کاهش اطلاعات ارسالی تا حد ممکن است، بر روی هر دسته از اطلاعات که در بخش قبل معرفی شدند، پیش از ارسال، کدگذاری منبع با نرخ فشرده‌سازی مناسب به ترتیب زیر اعمال می‌کنیم:

۱- برای قسمت‌هایی از تصویر که حذف نمی‌شوند، کدگذاری JPEG ساده اعمال می‌شود.

۲- برای بلوک‌هایی از تصویر که حذف می‌شوند، اطلاعات کمکی شامل موارد زیر کد می‌شوند:

الف) در بلوک‌های شامل لبه قابل حذف، محل لبه‌ها را با استفاده از روش Run Length کد کرده و ارسال می‌کنیم.

ب) در بلوک‌هایی از تصویر که بدون لبه بوده و شرایط لازم برای حذف را داشته‌اند، در هر بلوک برای هر یک از

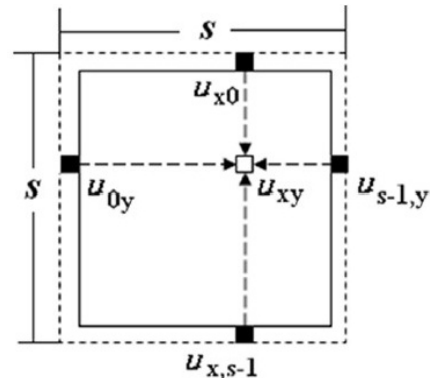
¹Quantization

² Differential Pulse Code Modulation

کدگذاری JPEG روی آنها اعمال شده و اطلاعات کمکی بلوک‌های حذف شده شامل محل لبه‌ها و گرادیان‌ها و محل بلوک‌های حذفی تصویر کدگشایی می‌شوند و در مرحله بعد فرآیند ترمیم برای بلوک‌های لبه‌ای و گرادیانی انجام می‌شود. در زیر بخش‌های بعدی به بررسی قسمت‌های مختلف گیرنده می‌پردازیم.

۳-۲-۱- نحوه بازسازی بلوک‌های بدون لبه

نخستین مرحله در گیرنده ترمیم بلوک‌های گرادیانی تصویر است. برای این منظور از روش PAI با کمی تغییر استفاده می‌کنیم. در این روش بلوک گرادیانی حذف شده با استفاده از اطلاعات نقاط معلوم همسایه و با کمک گرادیان آن بلوک که در فرستنده محاسبه و ارسال شده است، ترمیم و بازسازی می‌شود. نحوه بازسازی به این شکل است که برای یک ناحیه شامل بلوک‌های گرادیانی، ابتدا بلوک‌های با بیشترین همسایه معلوم (از بین چهار همسایگی ممکن) بازسازی می‌شوند و در مراحل بعد نوبت به بلوک‌های با همسایه‌های کمتر می‌رسد. در شکل (۹) حالتی نشان داده شده است که برای یک بلوک $S \times S$ تمام نقاط چهارهمسایگی در دسترس هستند. توجه کنید که $S \times S$ ابعاد بلوک و حاشیه موجود آن است. به عبارتی اگر بلوک حذفی 32×32 باشد، مقدار S برابر ۳۴ است.



(شکل-۹): یک بلوک $S \times S$ با چهار همسایه‌ی مرزی [9]
(Figure-9): An $S \times S$ block with four border pixels

اندازه هر نقطه نامعلوم u_{xy} درون این بلوک که در آن $(2 \leq x, y \leq S - 1)$ با استفاده از چهار نقطه متناظر آن در حاشیه بلوک و از روابط زیر تخمین زده می‌شود:

$$u_{xy} = k_1 p(u_{x,0}) + k_2 p(u_{x,s-1}) + k_3 p(u_{0,y}) + k_4 p(u_{s-1,y}) \quad (10)$$

در رابطه (۱۰)، داریم:

$$p(u_{i,j}) = u_{i,j} + (x - i) \hat{g}_x + (y - i) \hat{g}_y \quad (11)$$

$$k_1 = \frac{1}{2} \left(1 - \frac{y}{S-1} \right) \quad (12)$$

$$k_2 = \frac{1}{2} - k_1 \quad (13)$$

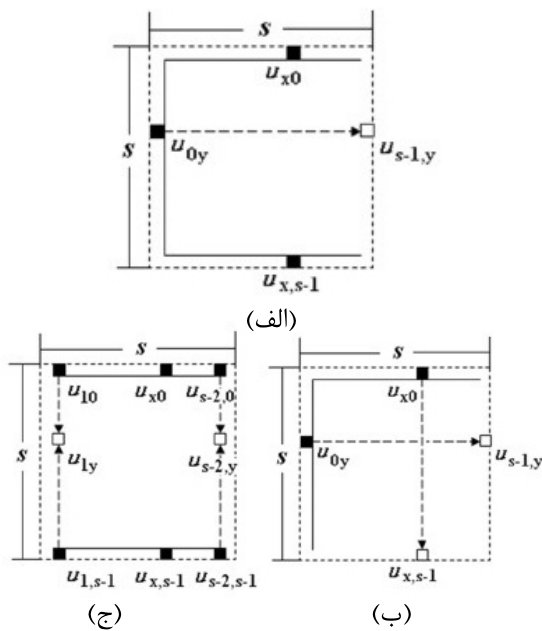
$$k_3 = \frac{1}{2} \left(1 - \frac{x}{S-1} \right) \quad (14)$$

$$k_4 = \frac{1}{2} - k_3 \quad (15)$$

که در آن $p(u_{i,j})$ تخمینی است که با استفاده از نقطه $u_{i,j}$ در حاشیه و با کمک گرادیان (\hat{g}_x, \hat{g}_y) برای نقطه u_{xy} به دست می‌آید. در شکل (۱۰) حالت‌های ممکن نشان داده شده‌اند. شکل الف حالتی را نشان می‌دهد که در آن سه حاشیه از این چهار حاشیه در دسترس هستند و در شکل‌های (ب) و (ج) تنها دو حاشیه موجود هستند. در این شرایط ابتدا تخمینی از حاشیه نامعلوم را به دست می‌آوریم. به‌عنوان مثال در حالت نخست که تنها به سه حاشیه دسترسی داریم، بعضی اطلاعات لازم را برای استفاده از رابطه (۷) در اختیار نداریم. بنابراین نقاط موجود در حاشیه حذف شده از رابطه زیر تخمین زده می‌شوند:

$$u_{s-1,y} = p(u_{0,y}) = u_{0,y} + (S-1) \hat{g}_x \quad (16)$$

در دو حالت بعد هم حاشیه‌های نامعلوم با روش‌های مشابهی تخمین زده می‌شوند.



(شکل-۱۰): تخمین حاشیه‌های بلوک گرادیانی حذف شده [9]
(Figure-10): Border pixels restoration

$$\omega_i = \frac{1}{d(x^{P_1}, x^P)}, \quad i \in \{1, 2\} \quad (18)$$

در روابط بالا، λ^{P_1} و λ^{P_2} اندازه نقاط حاشیه‌ای و معلوم P_1 و P_2 بوده و ω_1 و ω_2 به ترتیب معکوس فاصله نقطه P از نقاط P_1 و P_2 است. با استفاده از روابط بالا، مقدار نقطه P از ناحیه بلوک مشخص می‌شود. برای تمامی نقط نامعلوم روند بالا را جهت محاسبه مقدار نقاط تکرار می‌کنیم.

۳-۲-۳- نحوه پیدا کردن مسیرهای موازی لبه

برای دانستن مسیرهای موازی با لبه ابتدا باید الگوی لبه را به دست آوریم. برای این کار به عنوان نمونه برای یک بلوک 8×8 شده یک بلوک 10×10 شامل بلوک حذف شده و حاشیه موجود آن در نظر می‌گیریم، حال به جستجوی نخستین نقطه لبه موجود در این بلوک می‌پردازیم، به این صورت که یک جستجو در حاشیه بلوک انجام می‌دهیم و نخستین نقطه غیر صفر را به عنوان نخستین نقطه لبه موجود در بلوک انتخاب می‌کنیم. حال برای این نقطه در بین پنج همسایگی‌اش در بلوک 10×10 (از بین هشت همسایگی موجود) جستجو می‌کنیم تا نقطه لبه بعدی را پیدا کنیم. این نقطه دومین مقدار در بردار الگو است. حال هشت همسایگی این نقطه را بررسی می‌کنیم و به همین ترتیب بقیه مقادیر بردار الگو را به دست می‌آوریم و این بردار را کامل می‌کنیم. حال تمام نقاط حاشیه بلوک را به عنوان نقاط ابتدایی الگو در نظر گرفته و مسیرهای موازی را با توجه به رابطه تفاضلی مقادیر بردار الگوی به دست آمده برای هر یک از آنها می‌سازیم. بردار تفاضلی حاصل از این بردار که رابطه نقاط این الگو را نشان می‌دهد، را به دست می‌آوریم. البته اولویت در اینجا با نقاطی از حاشیه است که در همان سمت نخستین نقطه لبه باشند. بدین ترتیب با شروع از یک نقطه حاشیه‌ای با اولویت مسیری را که از این نقطه به موازات مسیر لبه است؛ پیدا می‌کنیم و براساس آن مسیر حذف شده بازیابی می‌شوند.

۴- شبیه‌سازی و تحلیل نتایج

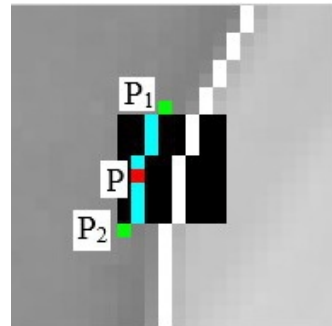
در این قسمت به بررسی نتایج شبیه‌سازی‌های انجام شده می‌پردازیم. در شکل (۱۲) هشت تصویری که در بررسی الگوریتم پیشنهاد شده در این مقاله مورد استفاده قرار گرفته، معرفی شده‌اند. در جدول (۱) نیز ویژگی‌های این تصاویر داده شده است. تعداد بیت بر پیکسل در تمامی تصاویر ۲۴

برخلاف روش PAI حالت وجود تک‌همسایه در اینجا در نظر گرفته نمی‌شود و آن قدر بلوک‌های با ۲ و ۳ و ۴ همسایه بازسازی می‌شوند تا دست کم یکی دیگر از همسایه‌های این نوع بلوک‌ها بازسازی شوند و بعد این بلوک‌ها با توجه به تعداد همسایه‌هایشان که از ۱ بیشتر شده بازسازی می‌شوند.

۳-۲-۲- نحوه بازسازی بلوک‌های شامل لبه

روش پیشنهادی در بازسازی بلوک‌های شامل لبه، از روش پیشنهاد شده در مقاله [21] الهام گرفته شده است. البته در اینجا مکان لبه‌ها به عنوان یکی از اطلاعات کمکی در دسترس هستند و نیازی به تخمین آنها نداریم. در روش قبل شکل لبه‌های موجود در مصنوعات^۱ دایره در نظر گرفته شده و از دایره‌های هم‌مرکز با دایره‌ای که لبه بخشی از آن دایره است، برای بازسازی نقاط مصنوعات استفاده می‌شد؛ اما ویژگی روش پیشنهادی در اینجا این است که الگوی اصلی لبه‌ها حفظ می‌شود و بقیه نقاط بلوک حذف‌شده بر روی منحنی انتقال یافته منحنی لبه اصلی ترمیم می‌شوند.

برای بازسازی هر نقطه حذف شده داخل بلوک، مانند نقطه P ، مسیر موازی با لبه که الگوی لبه را به طور دقیق دنبال کرده و از نقطه P نیز می‌گذرد، در نظر می‌گیریم و نقاط حاشیه‌ای و معلوم P_1 و P_2 بلوک را که جزئی از این الگو هستند، به عنوان نقاط مرجع جهت بازسازی نقطه P داخل بلوک در نظر می‌گیریم.



(شکل-۱۱): بازسازی نقطه P با استفاده از نقاط کمکی P_1 و P_2
(Figure-11): P pixel restoration using auxiliary pixels P_1 and P_2

حال با استفاده از رابطه زیر جهت تخمین نقاط مصنوعات که شامل تنها یک جفت لبه است و با دانستن محل و مقدار نقاط مرجع P_1 و P_2 ، به تخمین این نقطه می‌پردازیم:

$$\lambda^P = \frac{\omega_1 \lambda^{P_1} + \omega_2 \lambda^{P_2}}{\omega_1 + \omega_2} \quad (17)$$

که در آن داریم:

^۱ Artifact

کمی سازی کیفیت بدون داشتن تصویر مرجع است، به صورت زیر تعریف می شود:

فرض کنید تصویر دارای ابعاد $M \times N$ بوده و نقاط سیگنال به صورت $x(n, m)$ باشند، ابتدا سیگنال تفاضلی افقی به صورت زیر تعریف می شود:

$$d_h(m, n) = x(m, n+1) - x(m, n), \quad n \in [1, N-1] \quad (19)$$

سیس، اثر بلوکی شدن در راستای افقی بر اساس رابطه زیر تعریف می شود:

$$B_h = \frac{1}{M \left(\lfloor \frac{N}{8} \rfloor - 1 \right)} \sum_{i=1}^M \sum_{j=1}^{\lfloor \frac{N}{8} \rfloor} |d_h(i, 8j)| \quad (20)$$

برای تخمین میزان مات شدگی تصویر بدون دسترسی به تصویر مرجع از رابطه زیر که با اندازه اعوجاج بلوکی تلفیق شده است، استفاده می شود:

$$A_h = \frac{1}{7} \left[\frac{8}{M(N-1)} \sum_{i=1}^M \sum_{j=1}^{N-1} |d_h(i, j)| - B_h \right] \quad (21)$$

برای سنجش روشنی تصویر نیز از معیار عبور از صفر استفاده می شود، که به صورت زیر تعریف شده است:

$$Z_h = \frac{1}{M(N-2)} \sum_{m=1}^M \sum_{n=1}^{N-2} Z_h(m, n) \quad (22)$$

که در رابطه (22)، $Z_h(m, n)$ به صورت زیر است:

$$Z_h(m, n) = \begin{cases} 1 & \text{: Zero Crossing for } d_h(m, n) \\ 0 & \text{: Otherwise} \end{cases} \quad (23)$$

اکنون، مقدار پارامترهای داده شده در (22)-(20) برای راستای عمودی نیز محاسبه شده و میانگین راستای افقی و عمودی به عنوان پارامترهای A, Z, B در مدل نمره کیفیت QS زیر قرار داده می شود:

$$Qs = \alpha + \beta B^{\gamma_1} A^{\gamma_2} Z^{\gamma_3} \quad (24)$$

در مرجع [14] پارامترهای $\alpha, \beta, \gamma_1, \gamma_2, \gamma_3$ براساس داده های آموزشی (تصاویر معین) و نمره دهی گروهی به کیفیت تصاویر و در نهایت برازش داده ها به مدل بالا، به صورت زیر به دست آمده است:

$$\alpha = -245.9, \quad \beta = 261.9, \quad (25)$$

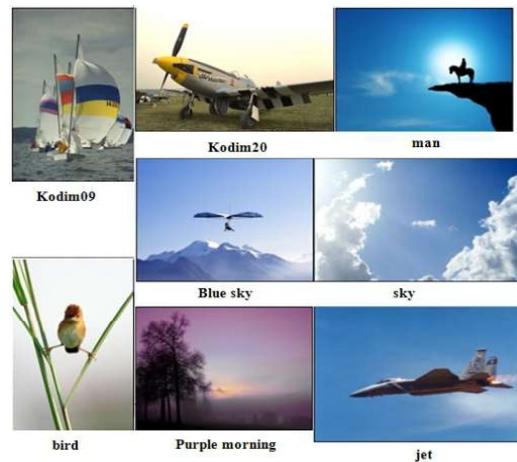
$$\gamma_1 = -0.024, \quad \gamma_2 = 0.016, \quad \gamma_3 = 0.0064$$

مقدار عددی QS عددی بین صفر تا شانزده است که هر چه بیشتر باشد، بیانگر کیفیت ادراکی بهتر تصویر است.

در جدول (2) نتایج اجرای الگوریتم پیشنهادی بر تصاویر معرفی شده ارائه شده است. در این جدول، میانگین بیت

ارسالی لازم در هر پیکسل، ضریب فشردگی، نسبت سیگنال به نوفه و کیفیت ادراکی تصویر دریافت شده آورده شده است.

همان طور که ملاحظه می شود، روش ارائه شده دارای bpp



(شکل-12): تصاویر استفاده شده در الگوریتم پیشنهادی (Figure-12): Images considered in our proposed algorithm

(جدول-1): ویژگی های تصاویر استفاده شده در بررسی الگوریتم

پیشنهاد شده

(Table-1): Considered images parameters for evaluation of the proposed algorithm

تصویر	ابعاد تصویر	حجم
Kodim09	۷۶۸×۵۱۲	۵۸۲۸۹۹
Kodim20	۵۱۲×۷۶۸	۴۹۲۴۶۲
jet	۵۰۰×۷۳۲	۹۸۱۸۰۴
bird	۷۰۰×۴۶۵	۲۰۷۵۵۷
man	۷۶۸×۱۰۲۴	۴۲۸۸۲۹
sky	۴۳۸×۷۰۰	۲۱۴۷۰۲
Blue sky	۷۶۸×۱۰۲۴	۱۱۹۳۶۴۹
Purple morning	۷۶۸×۱۰۲۴	۱۲۰۴۰۸۱

بیت بوده است.

جهت مقایسه نتایج شبیه سازی ها از معیارهای مختلفی استفاده کرده ایم. معیار بیت بر پیکسل BPP که برابر میانگین بیت ارسالی لازم در هر پیکسل است. معیار ضریب فشردگی^۱ (Cr) که این پارامتر نشان دهنده میزان فشردگی تصویر است و به صورت نسبت کل بیت های لازم برای ارسال تصویر اصلی و کل بیت های لازم برای ارسال یا ذخیره تصویر فشرده شده تعریف می شود. معیار نسبت سیگنال به نوفه^۲ (PSNR) که در حقیقت اگر متوسط مربعات خطای بین تصویر اصلی و تصویر بازسازی شده را به عنوان توان نوفه در نظر گرفته و بیشینه مقدار پیکسل تصویر را به توان دو رسانده (به عنوان توان سیگنال) و آن را به نوفه به دست آمده تقسیم کنیم مقدار PSNR به دست می آید. در نهایت، کیفیت ادراکی تصویر دریافت شده^۳ (Qs) از جمله معیارهای مورد استفاده در مقاله هستند [22]-[23]. این معیار که بیانگر یک روش

¹ Compression Rate (Cr)

² Peak Signal to Noise Ratio

³ Perceptual quality assessment

بسیار کمی است، به عبارتی، فشرده‌سازی قابل ملاحظه‌ای بدست آمده است.

(جدول-۲): نتایج اجرای الگوریتم پیشنهادی

(Table-2): Simulation results in our proposed algorithm

	BPP	Cr	PSNR	Qs
Kodim09	۰/۰۹۴۷	۲۵۳	۳۵/۸۲۹۶	۱۰/۵۵۶۰
Kodim20	۰/۰۵۲۲	۴۶۰	۳۶/۱۷۲۶	۱۰/۶۳۳۱
jet	۰/۰۶۳۷	۳۷۷	۳۵/۸۷۵۳	۱۱/۹۰۷۴
bird	۰/۱۱۴۹	۲۰۹	۳۵/۸۵۹۱	۹/۳۷۷۸
man	۰/۰۶۱۵	۳۹۰	۳۵/۰۰۹۷	۹/۱۴۰۰
sky	۰/۰۷۲۴	۳۲۴	۴۰/۷۱۹۳	۹/۹۵۰۹
Blue sky	۰/۰۴۶۷	۵۱۴	۳۵/۲۵۸۰	۱۰/۸۰۳۴
Purple morning	۰/۰۷۰۶	۳۴۰	۳۸/۲۸۴۶	۱۰/۸۴۶۷

جهت مقایسه کیفیت، در جدول (۴) مقادیر مربوط به معیار نسبت سیگنال به نوفه آورده شده است. همان‌طور که از نتایج این جدول مشخص است، اندکی کاهش کیفیت از دید مقایسه پارامتر PSNR در روش پیشنهادی نسبت به روش JPEG دیده می‌شود که البته این افت کیفیت در ازای حصول فشرده‌سازی قابل توجه به دست آمده است. همچنین، باید توجه شود که معیار PSNR بیان‌گر معیار بصری کیفیت نیست. لازم به ذکر است که در این مقایسه، میزان فشرده‌سازی روش ارائه‌شده بسیار بیشتر از روش استاندارد JPEG بوده است.

برای مقایسه روش پیشنهادی با روش‌های ارائه‌شده در دیگر مقالات، به دلیل پایه‌ای بودن مرجع [9]، مقایسه‌ای با روش PAI انجام داده‌ایم. در جدول (۵) میانگین بیت بر پیکسل، PSNR و معیار کیفیت ادراکی در روش‌های PAI، JPEG و روش پیشنهادی مقایسه شده‌اند. همان‌طور که جدول ۴-۵ نشان می‌دهد از نظر نرخ فشرده‌سازی، روش پیشنهادی بسیار بهتر عمل کرده است. اگرچه، همان‌طور که در نتایج جدول (۴) نیز مشاهده شد، ممکن است معیار کیفیت PSNR در مقایسه با روش‌های استاندارد چون JPEG کمتر باشد؛ اما، میزان فشرده‌سازی در روش پیشنهادی بسیار بالاتر است. به علاوه، جهت مقایسه کیفیت بصری، نتایج تصویری در شکل‌های (۲) و (۳) نشان داده شده است. نتایج تصویری نشان می‌دهند که نتایج کیفی مشابه همراه با فشرده‌گی بهتری از روش PAI به دست آمده است. ضریب کیفیت تبدیل JPEG برابر ۷۵ در مقایسه‌ها در نظر گرفته شده است.

برای اینکه با روش استاندارد فشرده‌سازی، مقایسه‌ای انجام شود، در جدول (۳) به ازای بیت بر پیکسل یکسان، نتایج مربوط به PSNR و Qs روش استاندارد JPEG و روش پیشنهادی ارائه شده است. همچنین، جهت مقایسه پژوهش‌های انجام‌شده در این زمینه، نتایج مربوط به روش‌های PAI و روش ارائه‌شده در مرجع [25] نیز در این جدول آورده شده است. همان‌طور که در جدول نیز مشخص است، روش پیشنهادی در بیش‌تر موارد به ازای bpps یکسان دارای نسبت سیگنال به نوفه و معیار کیفیت ادراکی Qs بالاتری است. البته همان‌طور که در قبل نیز اشاره شد، نکته مهم در روش ارائه‌شده، قابلیت فشرده‌سازی قابل توجه در نسبت سیگنال به نوفه و کیفیت ادراکی به‌طور تقریبی یکسان در مقایسه با روش‌های دیگر است.

(جدول-۳): میانگین بیت بر پیکسل الگوریتم پیشنهادی و الگوریتم [25]

(Table-3): bpp in our proposed algorithm and algorithm given by [25]

	bpp	JPEG		PAI		Proposed		Reference [25]	
		PSNR	Qs	PSNR	Qs	PSNR	Qs	PSNR	Qs
Kodim09	۰/۴	۳۸/۶۴	۹/۷۳	۳۸/۲۳	۱۰/۳۴	۴۱/۹۵	۱۱/۲۴	۳۹/۶۱	۱۰/۸۳
Kodim20	۰/۲	۳۷/۱۲	۹/۵۲	۳۶/۴۲	۱۰/۱۵	۳۸/۴۵	۱۰/۷۱	۳۷/۲۵	۱۰/۲۰
jet	۰/۱۵	۳۵/۸۷	۶/۱۲	۳۶/۱۲	۷/۱۸	۳۷/۲۷	۱۲/۳۱	۳۵/۶۸	۸/۱۲
bird	۰/۱۵	۴۱/۲۸	۷/۶۴	۴۱/۸۶	۸/۱۲	۴۲/۸۷	۱۰/۴۸	۴۱/۵۵	۸/۹۴
man	۰/۲۵	۴۴/۴۱	۷/۲۶	۴۳/۸۷	۹/۱۲	۴۵/۰۸	۱۰/۶۳	۴۵/۱۲	۹/۸۹
sky	۰/۲۵	۴۵/۱۷	۹/۱۴	۴۵/۲۴	۱۰/۱۲	۴۵/۴۷	۱۱/۰۱	۴۵/۳۸	۱۰/۸۲
Blue sky	۰/۱۵	۳۷/۴۳	۸/۱۵	۳۶/۸۸	۹/۸۷	۳۸/۸۱	۱۱/۳۷	۳۷/۱۶	۱۰/۱۱
Purple morning	۰/۲۵	۴۰/۲۸	۹/۱۸	۴۰/۵۷	۹/۲۸	۳۹/۴۸	۱۰/۹۳	۳۹/۷۱	۹/۸۲

(جدول-۴): نسبت سیگنال به نوفه در الگوریتم پیشنهادی و الگوریتم

JPEG

(Table-4): SNR in our proposed algorithm and JPEG algorithm

	Proposed method	Jpeg
Kodim09	۳۵/۸۲۹۶	۳۹/۴۸۶۷
Kodim20	۳۶/۱۷۲۶	۴۲/۹۸۸۴
jet	۳۵/۸۷۵۳	۳۷/۱۱۹۰
bird	۳۵/۸۵۹۱	۴۳/۲۵۲۸
man	۳۵/۰۰۹۷	۴۴/۴۰۵۷
sky	۴۰/۷۱۹۳	۴۵/۱۷۵۳
Blue sky	۳۵/۲۵۸۰	۳۷/۴۳۰۶
Purple morning	۳۸/۲۸۴۶	۴۰/۷۷۸۸

(جدول-۵): میانگین بیت بر پیکسل روش پیشنهادی با روش‌های JPEG، PAI و H.264

(Table-5): average bpb of our proposed method with JPEG, PAI, and H.264

	JPEG	Proposed method (Based on JPEG)	Bit Rate Saving	H.264	Proposed method (Based on H.264)	Bit Rate Saving	PAI	Bit Rate Saving in comparison with Our method (JPEG)	Bit Rate Saving in comparison with Our method (H.264)
Kodim09(BPP)	-/۵۲۲۵	-/۰۹۴۷	۸۱/۸۷	-/۴۸۰۲	-/۰۸۴۳	۸۲/۴۵	-/۴/۱۲	۲۳/۶۳	۴۷/۱
Kodim20(BPP)	-/۳۱۳۴	-/۰۵۲۲	۸۳/۳۴	-/۲۸۷۵	-/۰۴۴۳	۸۴/۶۱	-/۰۸۰	۵۳/۲۶	۸۰/۵۸
Kodim09(PSNR)	-/۵۲۲۵	-/۰۹۴۷	۸۱/۸۷	-/۴۸۰۲	-/۰۸۴۳	۸۲/۴۵	-/۱۲۴	۲۳/۶۳	۴۷/۱
Kodim20(PSNR)	-/۳۱۳۴	-/۰۵۲۲	۸۳/۳۴	-/۲۸۷۵	-/۰۴۴۳	۸۴/۶۱	-/۰۸۰	۵۳/۲۶	۸۰/۵۸
Kodim09(Qs)	-/۵۲۲۵	-/۰۹۴۷	۸۱/۸۷	-/۴۸۰۲	-/۰۸۴۳	۸۲/۴۵	-/۱۲۴	۲۳/۶۳	۴۷/۱
Kodim20(Qs)	-/۳۱۳۴	-/۰۵۲۲	۸۳/۳۴	-/۲۸۷۵	-/۰۴۴۳	۸۴/۶۱	-/۰۸۰	۵۳/۲۶	۸۰/۵۸



(ج) روش پیشنهادی

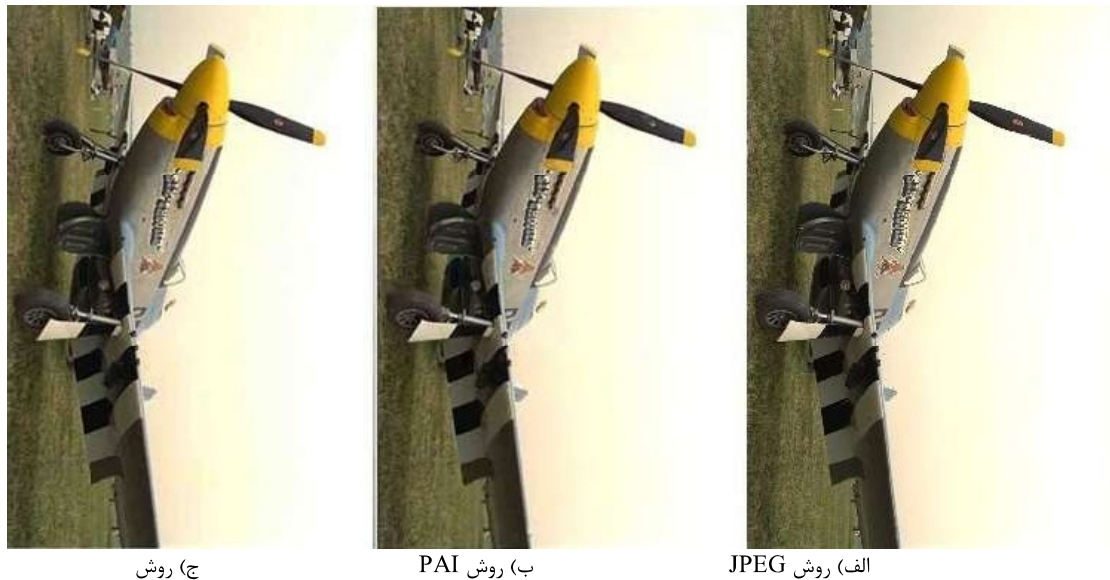
(ب) روش PAI

(الف) روش JPEG

(شکل-۱۳): نتایج فشرده‌سازی تصویر kodim09. از چپ به راست: JPEG، PAI، روش پیشنهادی و مقادیر معیارهای مقایسه برای روش

پیشنهادی، BPP=0.0947, PSNR=35.8296, Qs=10.556

(Figure-13): Compression results for kodim09 image. From left to right: JPEG, PAI, and our proposed method. Parameter values are: BPP=0.0947, PSNR=35.8296, Qs=10.556



پیشنهادی

(شکل-۱۴): نتایج فشرده‌سازی تصویر kodim20. از چپ به راست: JPEG، PAI، روش پیشنهادی و مقادیر معیارهای مقایسه

برای روش پیشنهادی،

BPP=0.0522, PSNR=36.1726, Qs=10.6331

(Figure-14): Compression results for kodim20 image. From left to right: JPEG, PAI, and our proposed method.

Parameter values are:

BPP=0.0522, PSNR=36.1726, Qs=10.6331

تشخیص صحیح تر بلوک‌های لبه‌ای حذفی و غیر حذفی و در نتیجه ترمیم بهتر تصویر فراهم شود. به علاوه، در بخش کدکردن اطلاعات کمکی، در صورت یافتن روش کدگذاری مناسب، امکان به دست آوردن میانگین بیت بر پیکسل بهتر وجود دارد. بنابراین، امکان بهبود کیفیت تصاویر بازسازی‌شده در روش پیشنهادی نیز با یافتن روش‌های ترمیم کارآمد در آینده وجود دارد.

۵- نتیجه‌گیری

در این مقاله، با استفاده از روش‌های حذف اطلاعات به صورت هوشمندانه و با استفاده از روش‌های ترمیم تصویر، روش جدیدی جهت فشرده‌سازی تصاویر پیشنهاد شد. این روش، علاوه بر این که باعث فشردگی قابل توجه تصویر می‌شود، نتیجه کیفی مناسبی نیز در گیرنده دارد. همان‌طور که در بخش نتایج دیده شد، از نقطه نظر معیارهای کیفیت مانند PSNR و Qs در بیش تر تصاویر انتخاب شده تفاوت قابل توجهی با روش JPEG و روش PAI دیده نمی‌شود و در روش پیشنهاد شده، مقدار نرخ فشرده‌سازی بسیار قابل توجه است. بنابراین، تصاویر فشرده و بازسازی‌شده با روش پیشنهادی بهتر یا در حد نتایج تصویری تصاویر فشرده و بازسازی‌شده با روش‌های دیگر است. نتایج ارائه شده در جدول معیار ادراکی کیفیت نیز این نتیجه را به طور کامل تأیید می‌کنند. این بهتر بودن کیفیت، زمانی که در کنار نسبت فشردگی قرار می‌گیرد، مزیت روش پیشنهادی را نسبت به روش‌های JPEG و PAI و دیگر روش‌های مطالعه شده تأیید می‌کند. همچنین، میانگین بیت بر پیکسل ارسالی با روش پیشنهادی در همه موارد از روش‌های دیگر بهتر است. می‌توان در ادامه روش‌های بهتر برای یافتن نوع خاکستری تصویر یا روش‌های لبه‌یابی استفاده کرد تا امکان

6. References

۶-مراجع

- [۱] شمسی گو شکی، ا.، نظام آبادی پور، ح.، سریزدی، س.، کبیر، ا.، "روشی برای بازخورد ربط براساس بهبود تابع شباهت در بازیابی تصویر بر اساس محتوا" فصل نامه علمی پژوهشی «پردازش علائم و داده‌ها»، دوره ۱۱، شماره ۲، ۱۳۹۳-۱۲، صفحات ۴۳ تا ۵۵
- [1] Shamsi gooshki A, Nezamabadi-pour H, Saryazdi S, Kabir E. "a relevance feedback approach based on similarity refinement in content based image retrieval". Journal of Signal and Data Processing, JSDP, vol. 11 (2) pp:43-55, 2015.
- [2] Bertalmio, Marcelo, Guillermo Sapiro, Vincent Caselles, and Coloma Ballester. "Image inpainting." In Proceedings of the 27th annual conference on Computer graphics and interactive techniques, ACM Press/Addison-Wesley

- on spatial redundancy removal and image inpainting." *Journal of Zhejiang University SCIENCE C*, vol. 11, no. 2 pp: 92-100, January 2010.
- [13] Guillemot, Christine, and Olivier Le Meur. "Image inpainting: Overview and recent advances." *IEEE Signal Processing Magazine*, vol. 31, no. 1 pp: 127-144, January 2014.
- [14] Wang, Zhou, Hamid R. Sheikh, and Alan C. Bovik. "No-reference perceptual quality assessment of JPEG compressed images." In *Image Processing 2002, International Conference on IEEE*, vol. 1, 2002, pp. 472-477
- [15] Bertalmio, Marcelo. "Strong-continuation, contrast-invariant inpainting with a third-order optimal PDE." *IEEE Transactions on Image Processing*, vol. 15, no. 7 pp: 1934-1938, July 2006.
- [16] Chen, Peiying, and Yuandi Wang. "Fourth-order partial differential equations for image inpainting." *ICALIP 2008. International Conference on Audio, Language and Image Processing, IEEE, 2008*, pp. 1713-1717.
- [17] Richard, Manuel M. Oliveira Brian Bowen, and McKenna Yu-Sung Chang. "Fast digital image inpainting." Appeared in the *Proceedings of the International Conference on Visualization, Imaging and Image Processing (VIIP 2001)*, Marbella, Spain, pp. 106-107.
- [18] Shen, Jianhong, and Tony F. Chan. "Mathematical models for local nontexture inpaintings." *SIAM Journal on Applied Mathematics* vol. 62, no. 3 pp: 1019-1043, 2002.
- [19] Chan, Tony F., and Jianhong Shen. "Nontexture inpainting by curvature-driven diffusions." *Journal of Visual Communication and Image Representation* vol. 12, no. 4 pp: 436-449, December 2001.
- [20] Bertalmio, Marcelo, Luminita Vese, Guillermo Sapiro, and Stanley Osher. "Simultaneous structure and texture image inpainting." *IEEE Transactions on Image Processing*, vol. 12, no. 8 pp: 882-889, August 2003.
- [21] Rareş, Andrei, Marcel JT Reinders, and Jan Biemond. "Edge-based image restoration." *IEEE Transactions on Image Processing*, vol. 14, no. 10 pp: 1454-1468, October 2005.
- [22] Ma, Wenjuan, Maolin Hu, and Pengyong Hu. Publishing Co., 2000, pp. 417-424.
- [3] Rane, Shantanu D., Guillermo Sapiro, and Marcelo Bertalmio. "Structure and texture filling-in of missing image blocks in wireless transmission and compression applications." *IEEE Transactions on Image Processing*, vol. 12, no. 3, pp: 296-303, March 2003.
- [4] Criminisi, Antonio, Patrick Pérez, and Kentaro Toyama. "Region filling and object removal by exemplar-based image inpainting." *IEEE Transactions on Image Processing*, vol. 13, no. 9, pp: 1200-1212, September 2004.
- [5] Xu, Zongben, and Jian Sun. "Image inpainting by patch propagation using patch sparsity." *IEEE Transactions on Image Processing*, vol. 19, no. 5, pp: 1153-1165, May 2010.
- [6] Ružić, T., and A. Pižurica. "Context-aware patch-based image inpainting using Markov random field modeling." *IEEE transactions on image processing*, vol. 24, no. 1, pp: 444-456, January 2015.
- [7] Weickert, Joachim. "From Optimized Inpainting with Linear PDEs Towards Competitive Image Compression Codecs." In *Image and Video Technology: 7th Pacific-Rim Symposium, PSIVT 2015, Auckland, New Zealand, November 25-27, Revised Selected Papers*, vol. 9431, Springer, 2016, pp: 63-68
- [8] Peter, Pascal, and Joachim Weickert. "Compressing images with diffusion-and exemplar-based inpainting." In *Scale Space and Variational Methods in Computer Vision*, Springer International Publishing, pp. 154-165, April 2015.
- [9] Liu, Dong, Xiaoyan Sun, Feng Wu, Shipeng Li, and Ya-Qin Zhang. "Image compression with edge-based inpainting." *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 10 pp: 1273-1287, January 2007.
- [10] Xiong, Zhiwei, Xiaoyan Sun, and Feng Wu. "Block-based image compression with parameter-assistant inpainting." *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp: 1651-1657, June 2010.
- [11] Zhao, Chen, Jian Zhang, Siwei Ma, and Wen Gao. "Wavelet inpainting driven image compression via collaborative sparsity at low bit rates." In *Image Processing (ICIP), 2013 20th IEEE International Conference on, IEEE, 2013*, pp. 1685-1689.
- [12] Bastani, Vahid, Mohammad Sadegh Helfroush, and Keyvan Kasiri. "Image compression based

حاضر ایشان دانشیار گروه مخابرات و الکترونیک دانشگاه شیراز هستند که از سال ۱۳۸۳ تا کنون مشغول تدریس هستند. ایشان همچنین به مدت ۹ ماه دوره فرصت مطالعاتی را در آزمایشگاه پردازش تصویر در دانشگاه لاروشل در شهر لاروشل کشور فرانسه در سال ۱۳۹۴ گذرانده‌اند. زمینه مورد علاقه ایشان پردازش تصویر و ویدیو، شنا سایی الگو، آنالیز تصاویر پزشکی، ماشین بینایی و سنجش از دور است.

نشانی رایانامه ایشان عبارت است از:

yazdi@shirazu.ac.ir



مریم‌السادات منافی مدرک

کارشناسی را در رشته مهندسی برق از گروه مهندسی برق و الکترونیک دانشگاه شیراز در سال ۱۳۸۶ و مدرک کارشناسی ارشد را در رشته مخابرات سیستم از دانشکده

مهندسی برق و کامپیوتر دانشگاه شیراز در سال ۱۳۹۰ دریافت کرده است. زمینه مورد علاقه ایشان پردازش تصویر است.

نشانی رایانامه ایشان عبارت است از:

m_manafi_2010@yahoo.com

"Image Inpainting under Single Image." In Congress on Image and Signal Processing, IEEE 2008. CISP'08. vol. 1, 2008, pp. 636-640.

[23] Gonzalez, Rafael C., and Richard E. Woods. Digital Image Processing. Prentice-Hall, New Jersey, 3rd Edition, 2007.

[24] Varghese, Sikha Mary, Alphonsa Johny, and Jubilant Job. "A survey on joint data-hiding and compression techniques based on SMVQ and image inpainting." In International Conference on Soft-Computing and Networks Security (ICSNS), IEEE, 2015, pp. 1-4.

[25] Di, Wu, Ren Li, and Wu Shuang. "Inpainting intergrate with decomposition for image compression." In Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), 2015 IEEE, 2015, pp.35-38.



علی جمشیدی مدرک کارشناسی را

در رشته الکترونیک از گروه مهندسی برق و الکترونیک دانشگاه شیراز در سال ۱۳۷۶ و مدارک کارشناسی ارشد و دکترا را در رشته مخابرات سیستم از دانشکده مهندسی برق دانشگاه

شریف به ترتیب در سال‌های ۱۳۷۸ و ۱۳۸۵ دریافت کرد. در حال حاضر ایشان دانشیار بخش مخابرات و الکترونیک دانشکده مهندسی برق و کامپیوتر دانشگاه شیراز هستند که از سال ۱۳۸۶ تاکنون مشغول تدریس هستند. زمینه مورد علاقه ایشان پردازش سیگنال، مخابرات بی‌سیم و پردازش تصویر است.

نشانی رایانامه ایشان عبارت است از:

jamshidi@shirazu.ac.ir



مهران یزدی مدرک کارشناسی را

در رشته مخابرات از گروه مهندسی برق و الکترونیک دانشگاه شیراز در سال ۱۳۶۹ و مدارک کارشناسی ارشد و دکترا را در رشته مخابرات سیستم از گروه مهندسی برق

دانشگاه لاوال شهر کبک کشور کانادا به ترتیب در سال‌های ۱۳۷۴ و ۱۳۸۱ دریافت کرد. ایشان در ادامه دوره فوق دکترا را در گرایش مهندسی پزشکی به مدت دو سال در دانشگاه کبک در شهر کبک کشور کانادا به اتمام رسانید. در حال

