



دانشگاه صنعتی شریف

دانشکده مهندسی کامپیوتر

گزارش پروژه‌ی پایانی درس پردازش علائم دیجیتال

عنوان:

Image Compression Algorithm Based On The DCT Transform

دکتر حسین صامتی

نام و نام خانوادگی:

مرضیه فرهادی (92204671)

سحر طاهری (92202527)

تینا خواجه (93210761)

زمستان ۱۳۹۳

چکیده

امروزه حجم اطلاعات برای ذخیره سازی و انتقال اطلاعات اهمیت ویژه‌ای داراست. لذا در پردازش علائم دیجیتال، فشرده سازی اطلاعات ورودی مورد توجه بسیار واقع شده است و روش‌های فراوانی در این رابطه ارائه شده است. در این پژوهش، فشرده سازی تصاویر که جزو مباحث پر اهمیت پردازش تصاویر است، مورد توجه قرار گرفته است. برای ذخیره‌سازی تصاویر باید حجم اطلاعات را تا جایی که ممکن است کاهش داد. اساس تمام روش‌های فشرده‌سازی کنار گذاردن بخش‌هایی از اطلاعات و داده‌ها است. قصد داریم به پیاده سازی روش ارائه شده در مقاله‌ی [1] بپردازیم. در این مقاله روشی برای فشرده سازی اتلافی تصاویر رنگی ارائه شده است که در آن بر روی تصاویر رنگی پس از مرحله‌ی پیش پردازش که شامل حذف میانگین از سه مؤلفه ی R، G و B و تبدیل فضای رنگ RGB به YCbCr است، با اعمال تبدیل DCT و الگوریتم bisection بر روی بلوک های تصویر و استفاده از روش adaptive scanning، تصویر فشرده شده بدست می‌آید.

کلیدواژه: فشرده سازی تصاویر رنگی، تبدیل DCT، adaptive scanning، پردازش تصویر

فهرست

۱- مقدمه	۱
۱-۱- تبدیل کسینوسی گسسته	۲
۲-۱- معیارهای عملکرد رایج در فشرده سازی تصویر	۳
۲- روش پیشنهاد شده	۴
۲-۱- تبدیل فضای رنگ از RGB به YCbCr	۴
۲-۲- تبدیل بلوکی DCT	۶
۲-۳- مرحله‌ی Quantization و Thresholding	۶
۲-۴- رمزگذار بدون اتلاف	۸
۲-۴-۱- انواع روش های scanning	۸
۲-۴-۲- فرمت جدید برای نمایش بردارها	۱۱
۲-۴-۳- روش Two-Role Encoder	۱۲
۳- آزمایشات انجام شده و نتایج بدست آمده در مقاله	۱۵
۴- آزمایشات و نتایج حاصل از پیاده سازی	۱۸
۴-۱- آزمایش اول نمایش کارایی الگوریتم:	۱۸
۴-۳- آزمایش سوم اندازه گیری معیارهای کیفیت:	۲۸
منابع	۳۰

۱- مقدمه

فشرده سازی تصویر به مسئله ای گفته می شود که طی آن میزان داده لازم برای نمایش تصویر را کاهش می دهیم. اساس این کاهش داده ها از بین بردن اطلاعات تکراری است. از دید ریاضیاتی این کاهش حجم را می توان به دید یک نگاشت تصویر دو بعدی به یک مجموعه ای از داده های کم ارتباط تر دید.

در این گزارش به بررسی شاخه ای از پردازش تصویر که خود زیرشاخه ای از پردازش سیگنال می باشد می پردازیم. می دانیم که در پردازش تصاویر، ورودی می تواند عکس یا تصویر ویدئویی باشد و خروجی متناظر نیز عکس، مجموعه ای از ویژگی ها و یا پارامترهای مرتبط با تصویر می باشد. پردازش تصاویر با اهداف مختلفی صورت می گیرد که معمولاً تصویر به صورت یک سیگنال دو بعدی در نظر گرفته می شود و تکنیک های مختلفی برای داشتن خروجی مناسب بر روی آن اعمال می گردد.

یکی از اهدافی که یک پروسه ی پردازش تصویر می تواند به دنبال داشته باشد فشرده سازی تصویر است، که طی آن حجم تصویر ورودی کاهش می یابد به گونه ای که کیفیت عکس خروجی، همچنان در حد قابل قبولی باشد. هدف فرایند فشرده سازی تصویر کاهش اطلاعات غیر ملزوم، از بین بردن اطلاعات اضافه و تکراری در عکس است به گونه ای که بتوان داده های آن را در فرم بهینه ای ذخیره و منتقل نمود.

در حالت کلی برای فشرده سازی تصاویر دو متد مختلف وجود دارد :

۱- فشرده سازی به همراه از دست دادن اطلاعات، با تلفات^۱.

۲- فشرده سازی بدون از دست دادن اطلاعات، بدون تلفات^۲.

در مورد اول پس از فشرده سازی تصویر، قسمتی از اطلاعات را از دست داده ایم به بیان دیگر امکان بدست آوردن تصویر اولیه از تصویر فشرده سازی شده تقریباً غیر ممکن است. در مورد دوم پس از فشرده سازی و

¹ lossy

² lossless

کاهش حجم عکس می توانیم دوباره عکس عملیات را انجام دهیم و به همه اطلاعات قبل از فشرده سازی دست یابیم، به بیانی دیگر طی این عملیات، اطلاعاتی از بین نمی رود و هر قسمتی که با هدف کاهش حجم از بین رفته است، قابل بازیابی هست.

۱-۱ تبدیل کسینوسی گسسته^۳

یکی از تبدیل هایی که برای فشرده سازی تصاویر مورد استفاده قرار می گیرد تبدیل سینوسی گسسته است. تبدیل کسینوسی گسسته دنباله ای از نقاط داده ها را به صورت حاصل جمع توابع \cos ای مشخص می نماید. تبدیل کسینوسی گسسته در زمینه های مختلف از علم و مهندسی اهمیت دارد و در مواردی همچون فشرده سازی همراه با اتلاف فایل های صوتی، تصاویر و ویدیو و غیره کاربرد دارد. استفاده از \cos به جای \sin در این حالت بسیار اهمیت دارد به این دلیل که باعث می شود در این موارد تعداد توابع \cos کمتری برای تخمین مورد احتیاج باشد. این تبدیل مشابه تبدیل DFT است اما تنها از اعداد حقیقی استفاده می نماید. به طور تقریبی می توان گفت که تبدیل کسینوسی گسسته معادل DFT با طول داده های دو برابر و برای داده های حقیقی هست. در کل ۸ نوع مختلف DCT معرفی شده است که ۴ نوع آن رایج هست. معمول ترین نوع DCT نیز نوع دوم آن است که به طور ساده به آن DCT می گویند. رابطه ای آن در زیر آمده است:

$$X_k = \sum_{n=0}^{N-1} x_n \cos \left[\frac{\pi}{N} \left(n + \frac{1}{2} \right) k \right] \quad k = 0, \dots, N-1. \quad (1)$$

³ Discrete cosine transform

۲-۱ معیارهای عملکرد رایج در فشرده سازی تصویر

عملکرد روش فشرده سازی با استفاده از دو معیار اساسی زیر برآورد می‌شود:

نسبت تراکم (CR) و اندازه گیری کیفیت تصویر بازسازی شده ($PSNR^4$).

در ادامه به تعریف این دو معیار همراه با معیارهای رایج دیگر برای تصاویر رنگی پرداخته می‌شود:

- **نسبت تراکم (CR):** نسبت بین اندازه تصویر اصلی و اندازه تصویر فشرده شده است که مقاله از رابطه‌ی زیر برای محاسبه استفاده می‌نماید.

$$CR = \frac{\text{Original RGB color image size in bits}}{\text{Compressed image in bits}} \quad (۲)$$

- **میانگین مربع خطا (MSE):** این معیار در اصل برای اندازه گیری میزان اعوجاج در تصویر بازسازی شده است و با رابطه زیر محاسبه می‌شود:

$$MSE = \frac{1}{N \times M} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} (x_{ij} - \hat{x}_{ij})^2 \quad (۳)$$

- **PSNR:** به عنوان یک معیار پذیرفته شده برای اندازه گیری کیفیت به طور گسترده در زمینه‌ی فشرده سازی تصاویر مورد استفاده واقع شده است.

برای تصاویر غیر رنگی برابر زیر هست:

$$PSNR = 10 \times \log_{10} \frac{255^2}{MSE} \quad (۴)$$

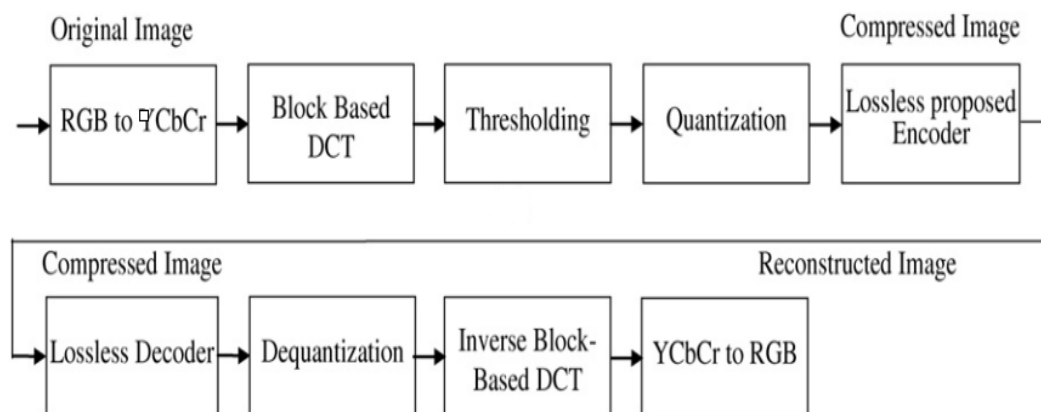
برای تصاویر رنگی برابر زیر هست:

$$PSNR = 10 \times \log_{10} \left(\frac{255^2 \times 3}{MSE(R) + MSE(G) + MSE(B)} \right) \quad (۵)$$

⁴ popular peak signal to noise ratio

۲- روش پیشنهاد شده

در شکل زیر دیاگرام روش فشرده‌سازی ارائه شده نشان داده شده است که در ادامه به ترتیب مراحل آن توضیح داده می‌شود:



شکل ۱ - بلوک دیاگرام روش پیشنهادی مقاله‌ی [1].

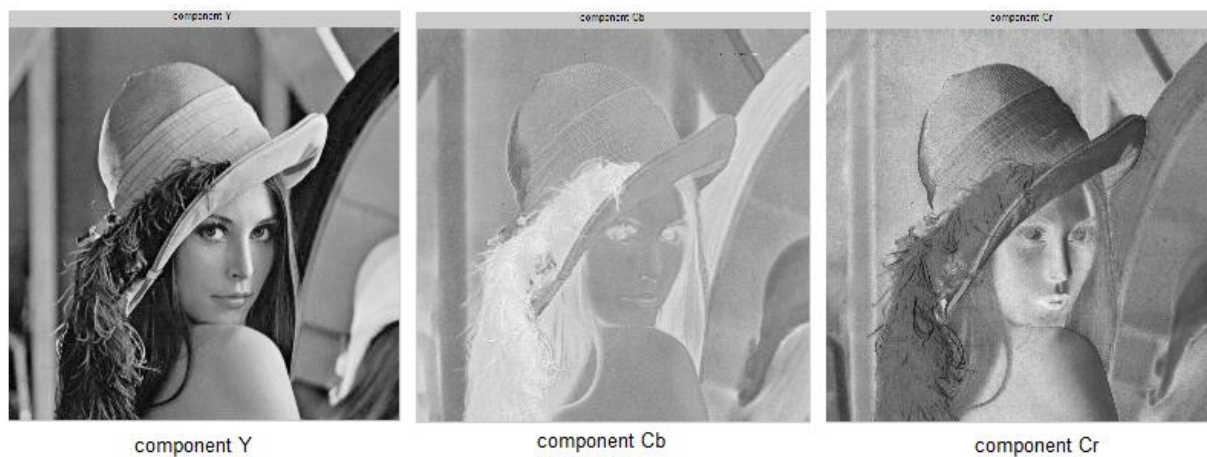
۲-۱- تبدیل فضای رنگ از RGB به YCbCr

این مرحله‌ی پیش پردازش در عمل برای کاهش همبستگی میان مولفه‌های رنگ هست. مزیتی که فضای رنگی YCbCr نسبت به فضای رنگی RGB دارد این است که در فضای رنگی RGB اطلاعات رنگی تصویر در هر سه کانال آن به طور تقریباً مساوی قرار دارد که در شکل ۲ قابل مشاهده است. اما در فضای رنگی YCbCr این همبستگی کاهش می‌یابد در فضای رنگ YCbCr بیشتر اطلاعات در کانال Y تصویر وجود دارد. بنابراین می‌توان با داشتن تنها یک کانال اطلاعات زیادی از تصویر را ذخیره کرد.



شکل ۲: همان طور که در شکل مشاهده می کنیم همبستگی میان سه مؤلفه ی رنگ R, G و B بسیار زیاد است

چون در فضای YCbCr توزیع انرژی در مؤلفه ی Y بیشتر است بنابراین دو مؤلفه ی دیگر می توانند بیشتر فشرده شوند. در شکل ۳ این خصوصیت در این فضای رنگ قابل مشاهده است.



شکل ۳: همان طور که در شکل زیر مشاهده می کنیم همبستگی میان سه مؤلفه کاهش یافته است بیشتر اطلاعات در کانال Y تصویر وجود دارد.

معیار PSNR برای کنترل کیفیت استفاده می شود و می تواند تضمین کند که کیفیت تصویر بعد از فشرده سازی در حد قابل قبولی باقی مانده است. ماتریس های تبدیل این دو فضای رنگ به یکدیگر در زیر آمده است.

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.144 \\ -0.16875 & -0.33126 & 0.5 \\ 0.5 & -0.41869 & -0.08131 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (۶)$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1.402 \\ 1 & -0.34413 & -0.71414 \\ 1 & 1.772 & 0 \end{bmatrix} \begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} \quad (۷)$$

۲-۲- تبدیل بلوک DCT

بعد از مرحله‌ی حذف میانگین و تبدیل فضای رنگ به YCbCr، هر یک از مؤلفه های Y، Cb، Cr به بلوک هایی با سایز ۸*۸، ۱۶*۱۶، ۳۲*۳۲ و ۶۴*۶۴ تقسیم شده و به طور مجزا بر روی آنها تبدیل DCT انجام می شود. در تبدیل DCT مانند تبدیل Wavelets انرژی در تعداد محدودی از ضرایب تبدیل ذخیره می-شوند.

۳-۲- مرحله‌ی Quantization و Thresholding

همانطور که در بلوک دیاگرام داده شده در شکل ۱ دیده می‌شود در طی فرایند فشرده‌سازی تصویر، یک مرحله به نام thresholding وجود دارد که مقادیر کوچکتر از یک عدد خاص را حذف می‌کند که چون در فرایند بازسازی تصویر در شکل ۳ عکس این عمل صورت نگرفته است، باعث از بین رفتن اطلاعات می‌شود. بنابراین تلفات ناشی از فشرده‌سازی در طی دو مرحله‌ی thresholding و quantization صورت می‌گیرد به گونه‌ای که با معیارهای کنترل کیفیت همانند PSNR و MSE که در بخش‌های گذشته اشاره شده‌است، این میزان فشرده‌گی کنترل می شود.

فرض کنید $GPSNR^5$ ، PSNR هدف باشد که بعد از thresholding و quantization بخواهیم به آن دست یابیم؛ فرآیند فشرده سازی به وسیله ی حل معادله ۸ قابل کنترل است :

$$PSNR(TH) - GPSNR = 0 \quad (۸)$$

معادله ی ۸ را می توان در یک محدوده قابل حل تعریف کرد که تخمین بهینه ترین TH^* را ممکن می سازد:

$$|PSNR(TH^*) - GPSNR| \leq \varepsilon \quad (۹)$$

در این مقاله $\varepsilon = 0.5\%$ انتخاب شده است.

Quantization در این مقاله بر اساس quantizer خطی انجام می شود که تجربه نشان داده که یک quantizer ۷بیتی، کیفیت تصاویر رنگی را در حد قابل قبول حفظ می کند. برای این مرحله داریم:

$$QNZDCT = \left\lfloor 1 + \left(\frac{NZDCT - NZDCT_{\min}}{NZDCT_{\max} - NZDCT_{\min}} \right) (2^{nbits} - 2) \right\rfloor \quad (۱۰)$$

که NZDCT ضرایب غیرصفر تبدیل بعد از thresholding است و $NZDCT_{\min}$ و $NZDCT_{\max}$ به ترتیب کمترین و بیشترین مقدار این ضرایب هستند. مقادیر NZDCT بعد از quantization (QNZDCT) در بازه ی $[1, (2^{nbit} - 1)]$ قرار دارند که nbit همان تعداد بیت های مورد نیاز برای کد کردن ضرایب است.

⁵ Goal PSNR

۲-۴- رمزگذار بدون اتلاف

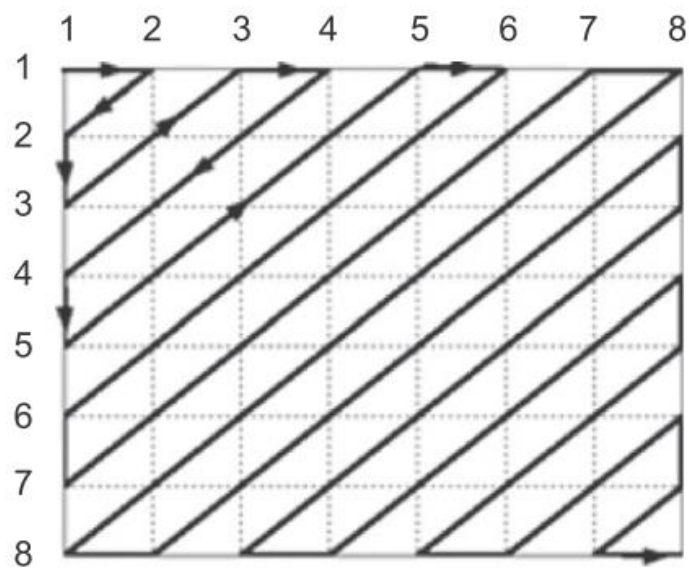
در این مقاله روشی برای کدگذاری بلوک‌های تبدیل threshold و quantize شده، ارائه شده است که بدون اتلاف باعث کاهش حجم بسیاری در فشرده سازی انجام شود. در این روش ابتدا بلوک های $n \times n$ به بردارهایی با طول n^2 تبدیل می شوند. روش هایی برای scanning بلوک‌ها و تبدیل آنها به بردار وجود دارد که در این مقاله ۴ نوع scanning بلوک ها مطرح شده و به صورت adaptive برای هر بلوک، بررسی می شود که کدام نوع scanning برای آن بهتر است. این بهتر بودن بدین معنی است که با scanning یک بلوک رشته هایی از ضرایب صفر در بردار ایجاد می شود که در کد کردن این رشته ها حذف شده و عددی به عنوان تعداد صفرها در کدگذاری قرار داده می شود. حال نوع scanning ای که بیشترین طول دنباله‌ی صفر در انتهای بردار را ایجاد کند به عنوان مناسبترین scanning برای آن بلوک به حساب میاید و در کدگذاری لحاظ می شود.

۲-۴-۱ انواع روش های scanning

در این مقاله ۴ نوع scanning در بلوک استفاده شده است. در ادامه با شکل به نحوه‌ی محاسبه آنها می-

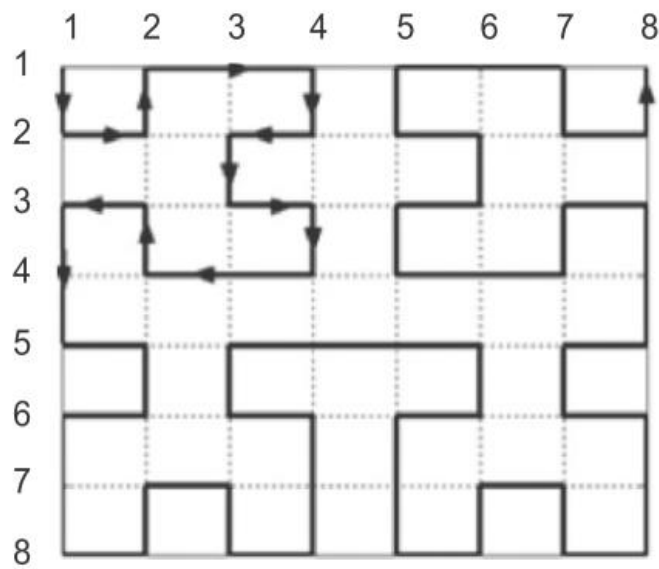
پردازیم:

• Zigzag scanning :



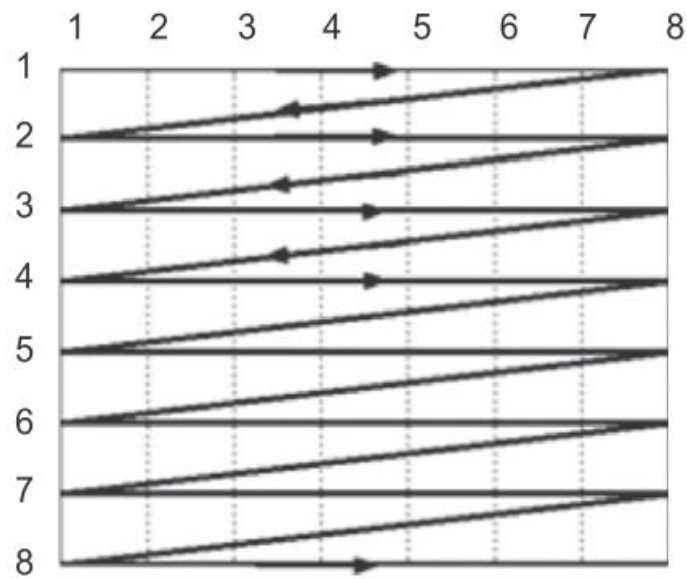
شکل ۴: روش ZigZag

• Hilbert scanning :



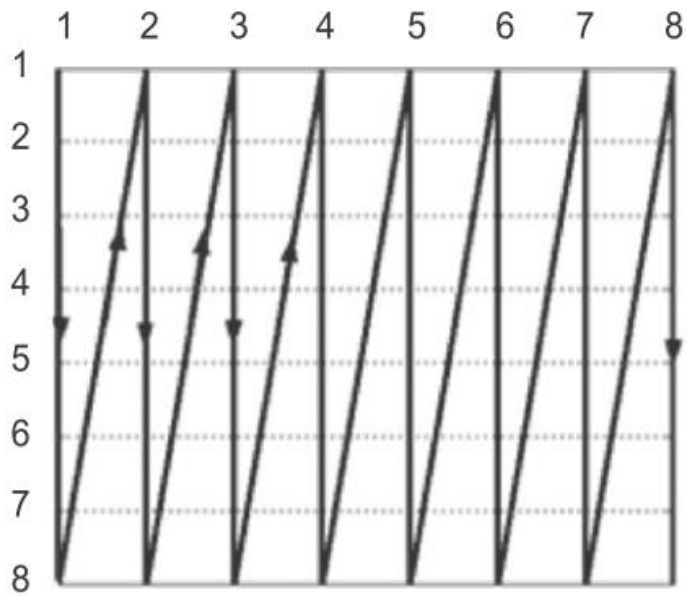
شکل ۵: روش Hilbert

• Horizontal scanning :



شکل ۶: روش Horizontal

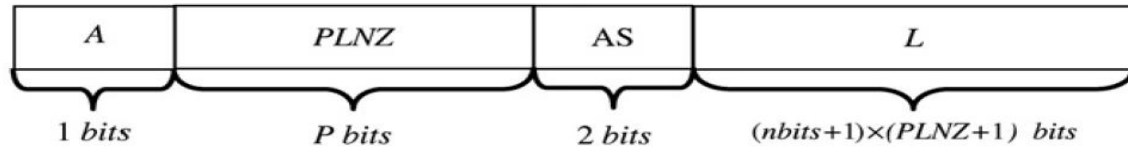
• Vertical scanning :



شکل ۷: روش Vertical

۲-۴-۲ فرمت جدید برای نمایش بردارها

تک تک بردارهایی که از scanning بلوک ها بدست می آیند به فرمت زیر در می آیند.



شکل ۸: فرمت جدید برای نمایش بردارها

که توضیح هر بخش بردار بالا، در زیر آمده است :

- (۱ بیت) *A* یا بیت فعالیت :

این بیت اگر ۰ باشد به معنی صفر بودن همه ی ضرایب DCT در بلوک است در غیر این صورت مقدار آن ۱ است.

- (*P* بیت) *PLNZ* :

این مجموعه بیت آدرس آخرین ضریب QNZDCT در بردار را نگه می دارد. مثلا اگر مکان آخرین ضریب غیر صفر در بردار حاصل از scanning یک بلوک، *k* باشد معادل باینری *k-1* در *PLNZ* قرار می گیرد.

P ماکزیمم تعداد بیت مورد نیاز برای کد کردن *PLNZ* است چون ماکزیمم مقدار *PLNZ* برابر $n \times n - 1$

است و $P = \log_2 n \times n$ است.

- (۲ بیت) *AS* :

این دو بیت برای نشان دادن ۴ نوع scanning استفاده می‌شود که بهترین نوع scanning که دنباله ای با بیشترین طول در انتهای بردار می‌دهد را مشخص می‌کند. به طور مثال 00 برای کد کردن روش Zigzag scanning، 01 برای کد کردن روش Horizontal scanning، 10 برای کد کردن روش Vertical scanning و 11 برای کد کردن روش Hilbert scanning استفاده می‌شود.

$$\bullet (PLNZ + 1) \times (nbits + 1) \text{ بیت } L :$$

همه‌ی ضرایب تبدیل quantize شده صفر یا غیر صفر با nbits+1 بیت کد می‌شوند. به تعداد PLNZ+1 ضریب در برداری که صفرهای انتهای آن حذف شده وجود دارد. بنابراین به تعداد L بیت برای کدگذاری بردار نیاز است.

بخش L در فرمت جدید نمایش بردار، خود نیز توسط Two-Role Encoder کدگذاری می‌شود که این کدگذار نیز بدون اتلاف، بخش L را فشرده می‌کند.

۲-۴-۳ روش Two-Role Encoder

در بردارهایی که با ۴ روش scanning بدست می‌آیند با حذف صفرهای انتهایی طول بردارها تغییر می‌کنند و فیلد AS با کد کردن روشی که طول بردار را بیشتر کاهش می‌دهد پر می‌شود. سپس فیلد L به وسیله‌ی two-role encoder کد می‌شود. الگوریتم این کدگذار در مقاله‌ی [2] ارائه شده است. در ادامه روش ارائه شده را بیان می‌کنیم.

Two-Role encoder تغییر یافته‌ی روش RLE (Run-Length Encoder) است و مطابق با توضیحات زیر کار می‌کند :

- همانطور که گفته شد ضرایب غیر صفر و quantize شده تبدیل DCT با $nbits+1$ بیت کد می شوند که $nbits$ دقت quantizer استفاده شده است. بنابراین ماکزیمم ضریب غیر صفر به $2^{nbits}-1$ و کمینه مقدار غیر صفر به ۱ کد می شوند.

- دنباله‌هایی از صفرهای پشت سرهم با طول‌های مختلف در بردارها وجود دارد که می توان آن‌ها را برداشت و تعداد آن‌ها را با $nbits$ کد کرد که با این تعداد بیت ماکزیمم طول $2^{nbits}-1$ صفر پشت سرهم را می توان کد کرد.

چون quantizer، دقت $nbits$ دارد برای کد کردن هر ضریب به $nbits+1$ بیت نیاز است. ۱ بیت اضافه در ابتدای هر کد قرار می‌گیرد که ۱ بودن آن، نشان دهنده‌ی ضرایب غیر صفر است و صفر بودن آن یعنی اگر بیت اضافی در ابتدای کد صفر باشد نشان دهنده‌ی تعداد صفر در یک دنباله ی صفر پشت سرهم در آن مکان است.

بنابراین در کدگشایی اگر $TRE \geq 2^{nbits}$ باشد یک QNZDCT است که با حذف بیت اول و تبدیل باینری به دسیمال $nbits$ بعد می‌توان آنرا کدگشایی کرد.

و اگر $TRE < 2^{nbits}$ باشد یک دنباله‌ی صفر کدگشایی می‌شود که تعداد آن، با حذف بیت اول و تبدیل باینری به دسیمال $nbits$ بعد بدست می آید.

برای مثال بردار زیر را فرض کنید از Zigzag scanning یک بلوک تصویری زیر بدست آمده است:

$$x = \begin{bmatrix} 163 & 157 & 159 & 160 & 159 & 154 & 156 & 157 \\ 161 & 155 & 155 & 157 & 157 & 157 & 157 & 152 \\ 159 & 159 & 157 & 157 & 159 & 154 & 158 & 154 \\ 160 & 159 & 158 & 158 & 156 & 155 & 157 & 158 \\ 161 & 158 & 160 & 157 & 156 & 153 & 156 & 171 \\ 159 & 155 & 154 & 158 & 156 & 155 & 154 & 160 \\ 157 & 156 & 151 & 160 & 157 & 156 & 156 & 155 \\ 158 & 158 & 158 & 159 & 155 & 156 & 154 & 171 \end{bmatrix}$$

بعد از طی مراحل خروجی زیر را داریم و جواب Zigzag scanning در زیر آمده است:

$$QXDCT = \begin{bmatrix} 97 & 0 & 27 & 0 & 28 & 0 & 0 & 0 \\ 0 & 27 & 27 & 27 & 27 & 27 & 0 & 27 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 27 & 0 & 0 & 0 \\ 27 & 0 & 27 & 26 & 0 & 0 & 27 & 0 \\ 0 & 0 & 27 & 27 & 0 & 0 & 27 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Zigzag scan (AS = 00)

97 0 0 0 27 27 0 27 0 0 27 0 0 27 28 0 27 0 0 0 0 0 27 0
0 27 0 0 0 0 27 26 27 0 0 0 0 27 0 0 0 27 0 0 0 0 0 0 0
27 0 0 27.

TRE بردار نتایج بدست آمده، برابر زیر است :

After TRE:

225 3 155 155 1 155 2 155 2 155 156 1 155 6 155 2 155 4
155 154 155 4 155 3 155 9 155 2 155.

سپس معادل باینری این اعداد را با دقت ۸ بیت (در این مثال quantizer ۷بیتی بوده است) بدست آورده و در کنار هم قرار می دهیم و ۱ بیت A و ۲ بیت AS و ۶ بیت PLNZ ($\log_2 8 \times 8 = 6$) را به رشته‌ی بدست آمده اضافه می کنیم. سپس برای تمام بلوک ها به طور مشابه رشته ها را بدست آورده و در کنار هم قرار می دهیم. اکنون رشته ای از بیت های صفر و یک داریم که نمایانگر یک تصویر است و قطعاً دارای تعداد کمتری بیت نسبت به حجم اولیه‌ی تصویر است.

۳- آزمایشات انجام شده و نتایج بدست آمده در مقاله

برای بررسی این روش و نشان دادن مزیت‌های آن، چند تصویر مهم که در پردازش تصویر کاربرد دارند مورد استفاده قرار گرفته است از جمله تصاویر lena و airplane و peppers در ابعاد 512×512 و چند تصویر در ابعاد 256×256 .

یکی از آزمایشاتی که انجام شده، مقایسه‌ی کارایی روش فشرده سازی adaptive scanning پدر فضای RGB با YCbCr است. در این آزمایشات پارامترهایی چون دقت quantizer که با بیت مشخص شده و ابعاد بلوک بندی تصاویر برای تبدیل DCT تغییر داده شده و نتایج بدست آمده در جدولی قرار داده شده است که بخشی از آن را در جدول زیر می بینید:

Performances in the RGB space for the different DCT block sizes and the quantizer width.

RGB nbits	7			8		
	PSNR	bpp	CR	PSNR	bpp	CR
Image						
Bloc 8 × 8						
Airplane	30.32	1.2241	19.606	30.349	1.3374	17.945
Peppers	30.216	1.2457	19.266	30.244	1.3571	17.685
Lena	32.057	1.5417	15.567	31.793	1.6002	14.998
Bloc 16 × 16						
Airplane	30.372	0.8735	27.475	30.456	0.9717	24.698
Peppers	30.167	0.8916	26.919	30.055	0.9417	25.485
Lena	31.959	1.1272	21.292	32.071	1.266	18.958

از مقایسه دو جدول بالا می توان دریافت که کارایی الگوریتم در فضای رنگ YCbCr بیشتر بوده و با PSNR قابل قبول به نرخ فشرده سازی بهتری میتوان دست یافت. هم چنین در یک PSNR تقریباً ثابت، با quantizer ۷ بیت هم می توان کیفیت را ثابت و فشرده گی را افزایش داد.

با بلوک های ۱۶*۱۶ نیز نتیجه ی بهتری نسبت به بلوک های ۸*۸ می توان گرفت.

CR(نرخ فشرده سازی) و bpp از روابط زیر محاسبه می شوند:

$$CR = \frac{\text{Original RGB color image size in bits}}{\text{Compressed image in bits}}$$

and

$$bpp = \frac{24 \text{ bits}}{CR}$$

هم چنین در جدولی استاندارد JPEG با روش ارائه داده شده مقایسه شده که در شکل زیر آن را می بینیم:

Performances comparison between the proposed method and the standard DCT JPEG algorithm.

Image	Proposed method YCbCr (16×16)		JPEG [4]	
	<i>PSNR</i>	<i>bpp</i>	<i>PSNR</i>	<i>bpp</i>
Airplane	31.403	0.7208	31.46	0.9
Peppers	30.3330	0.8772	30.47	1.47
Lena	32.7729	1.0073	32.76	1.03

با توجه به جدول بالا می بینیم که در PSNR تقریباً مشابه *bpp* روش ارائه شده نسبت به *bpp* استاندارد JPEG کمتر و در نتیجه نرخ فشرده سازی بیشتر است.

• مقایسه‌ی کیفیت روش پیشنهادی و روش CBTC-PF و JPEG

Performances comparison between the proposed method and the CBTC-PF algorithm.

Image	Proposed method YCbCr (16×16)		CBTC-PF [4]	
	<i>PSNR</i>	<i>bpp</i>	<i>PSNR</i>	<i>bpp</i>
Airplane	30.381	0.5917	30.36	1.04
Peppers	30.059	0.809	30.15	1.5
Lena	31.97	0.8101	31.93	1.17
Girl	35.002	0.4574	35.13	0.6
Couple	32.287	0.9252	32.44	1
House	31.726	0.8232	31.79	1.2
Zelda	31.333	0.8705	31.31	1.12
Average	31.8226	0.7553	31.8729	1.09

Performances comparison between the proposed method and the standard DCT JPEG algorithm.

Image	Proposed method YCbCr (16×16)		JPEG [4]	
	<i>PSNR</i>	<i>bpp</i>	<i>PSNR</i>	<i>bpp</i>
Airplane	31.403	0.7208	31.46	0.9
Peppers	30.3330	0.8772	30.47	1.47
Lena	32.7729	1.0073	32.76	1.03
Girl	36.9587	0.6920	36.85	0.62
Couple	33.0700	1.1276	33.02	0.94
House	31.3231	0.7546	31.34	1.24
Zelda	32.0521	1.0890	32.06	1
Average	32.5590	0.8955	32.5657	1.0286

۴- آزمایشات و نتایج حاصل از پیاده سازی

۴-۱ آزمایش اول نمایش کارایی الگوریتم:

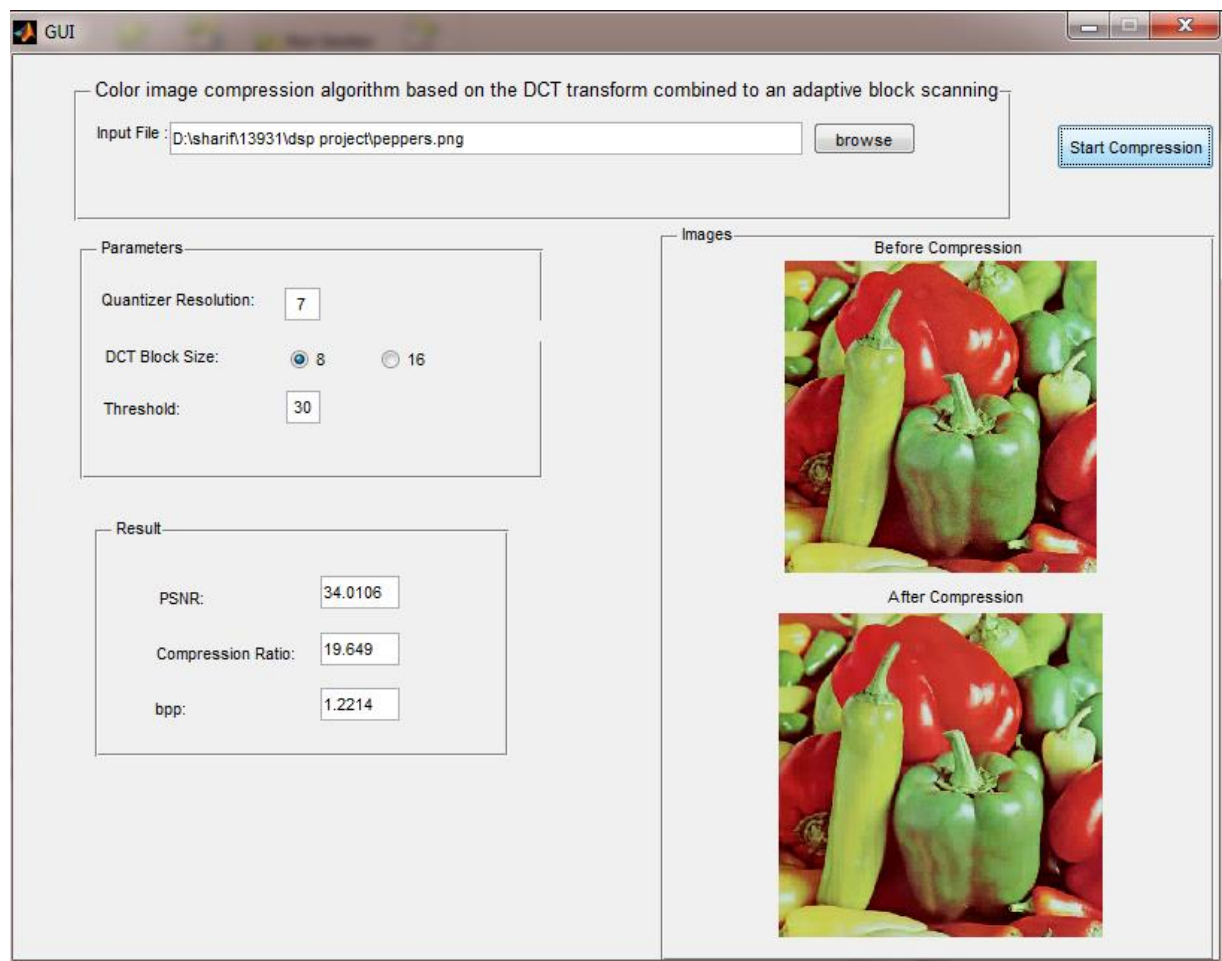
ابتدا برای نشان دادن کارایی الگوریتم، تصاویر متعددی را فشرده کرده و نرخ فشرده سازی و کیفیت تصاویر بعد از اجرای الگوریتم را نشان می دهیم. سعی شده پارامترها به گونه ای انتخاب شوند که نتایج تا حدودی به نتایج مقاله نزدیک باشد. همان طور که در نتایج بدست آمده مشاهده می کنیم در مقابل بدست آوردن نرخ فشرده سازی خوب، کیفیت تصویر نیز تقریباً خوب باقی مانده و این یکی از محاسن این الگوریتم فشرده سازی است. پارامترهایی که در آزمایشات تغییر کرده دقت کوانتایزر یا همان quantizer resolution است که واحد آن بیت بوده. همچنین سایز بلوک بندی تصویر در تبدیل DCT نیز از ۸ به ۱۶ قابل تغییر است و مشاهده می کنیم که سایز ۱۶ نتایج بهتری حاصل می کند. هم چنین threshold پارامتری است که افزایش آن نرخ فشرده سازی را افزایش داده و تا حدودی کیفیت یا همان PSNR تصویر را کاهش می دهد.

تصویری pepper

Quantizer resolution: 7 bit

Threshold: 30

Dct block size: 8



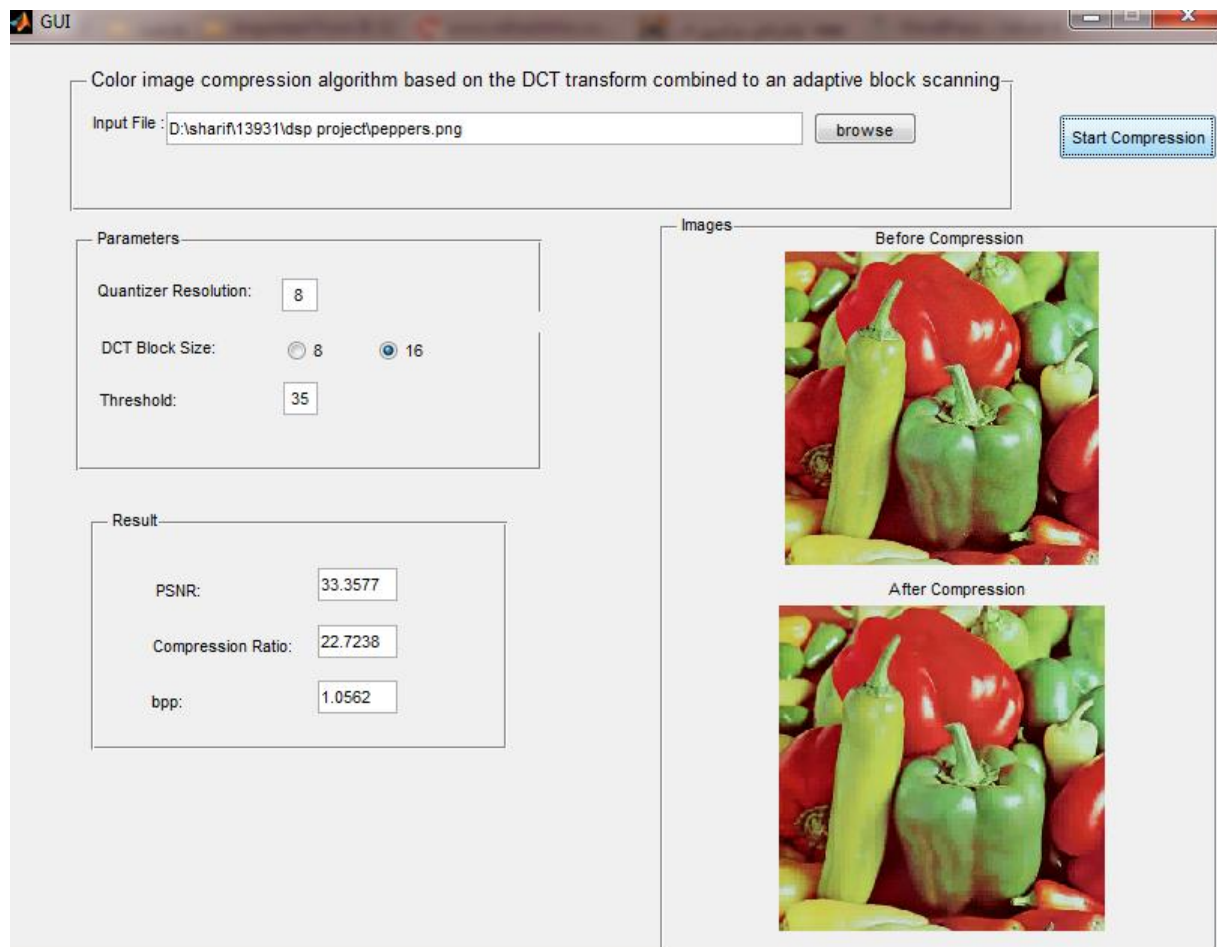
شکل ۳

پارامترها:

Quantizer resolution: 8bit

Threshold: 35

Dct block size: 16

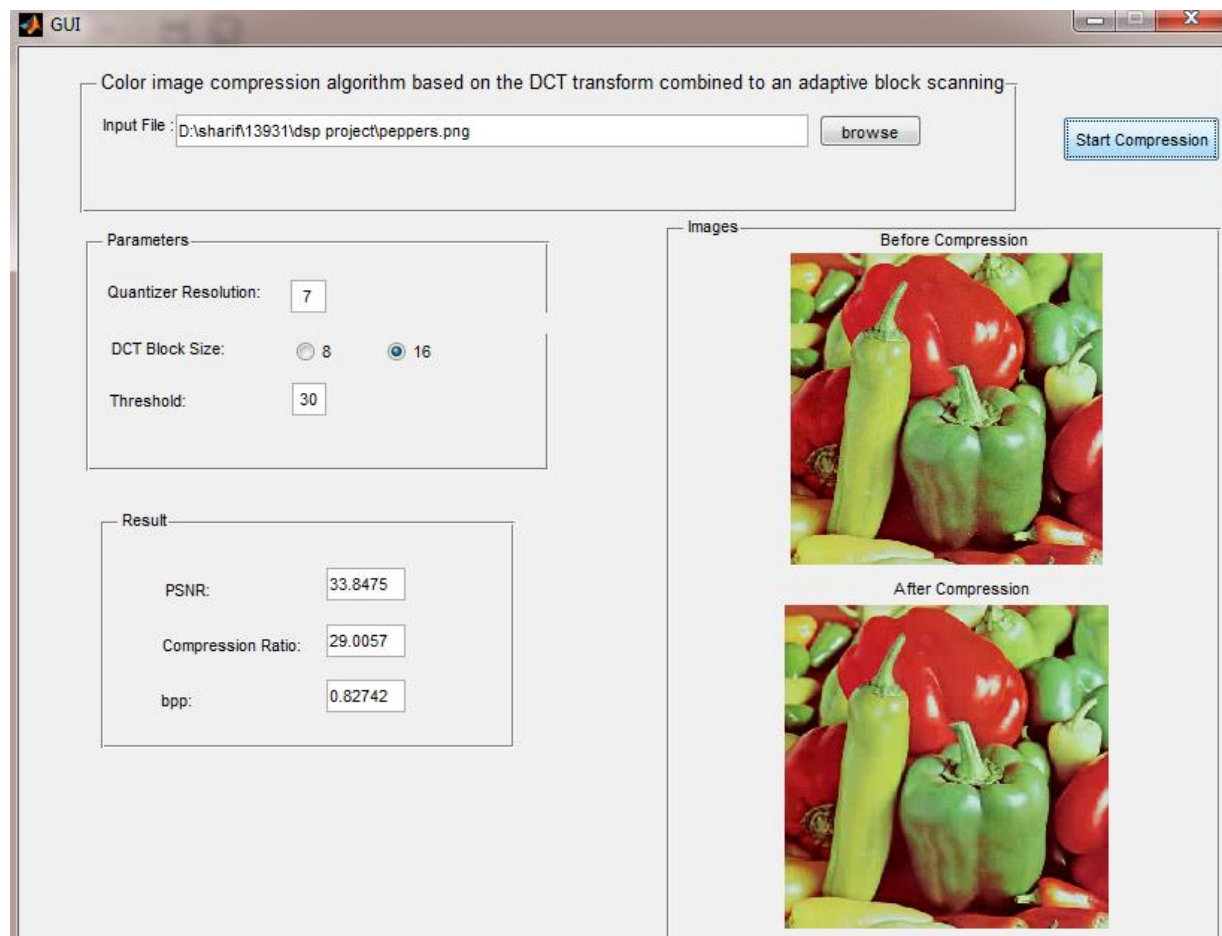


شکل ۱۵

Quantizer resolution: 7 bit

Threshold: 30

Dct block size: 16

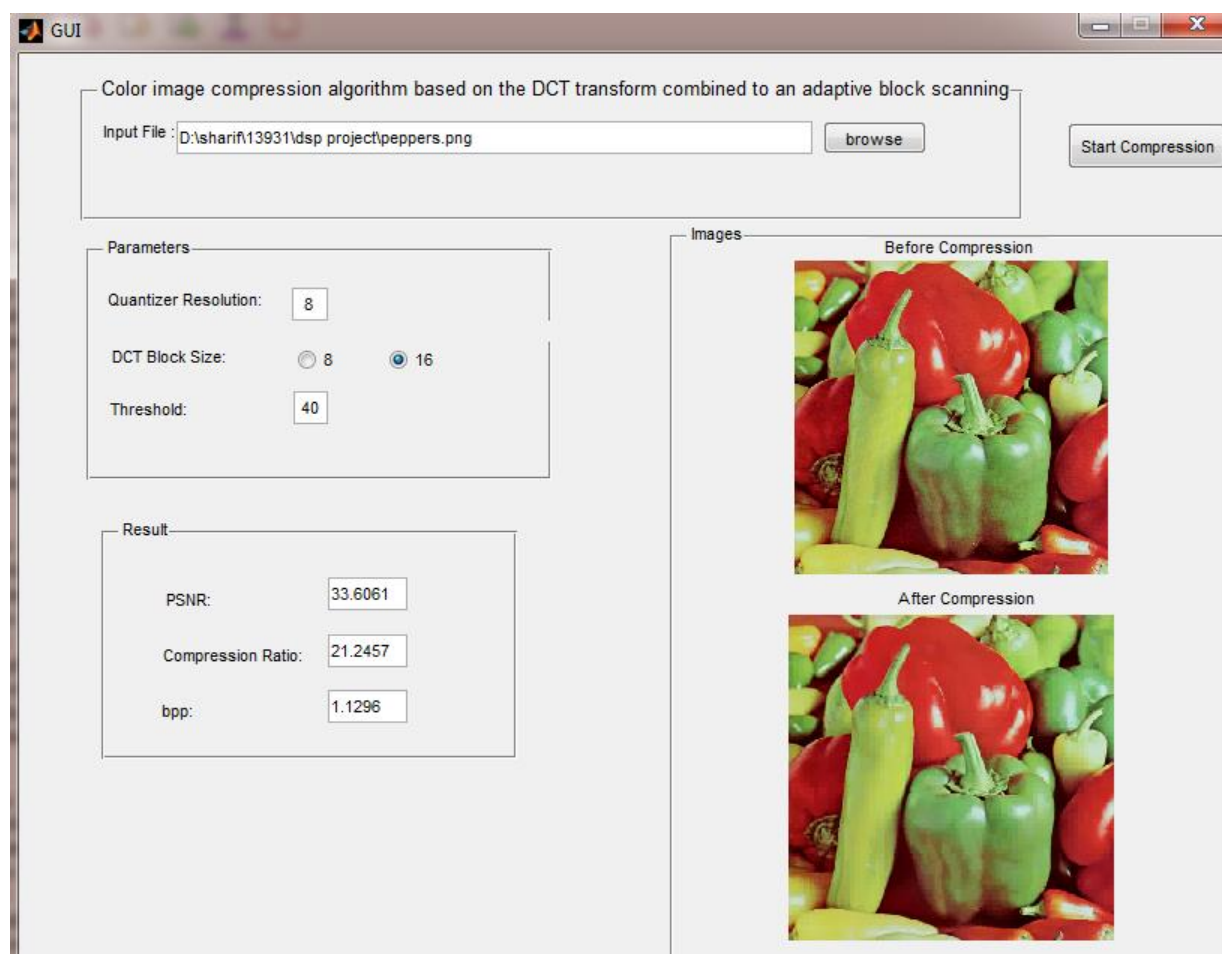


شكل ٤

Quantizer resolution: 8 bit

Threshold: 40

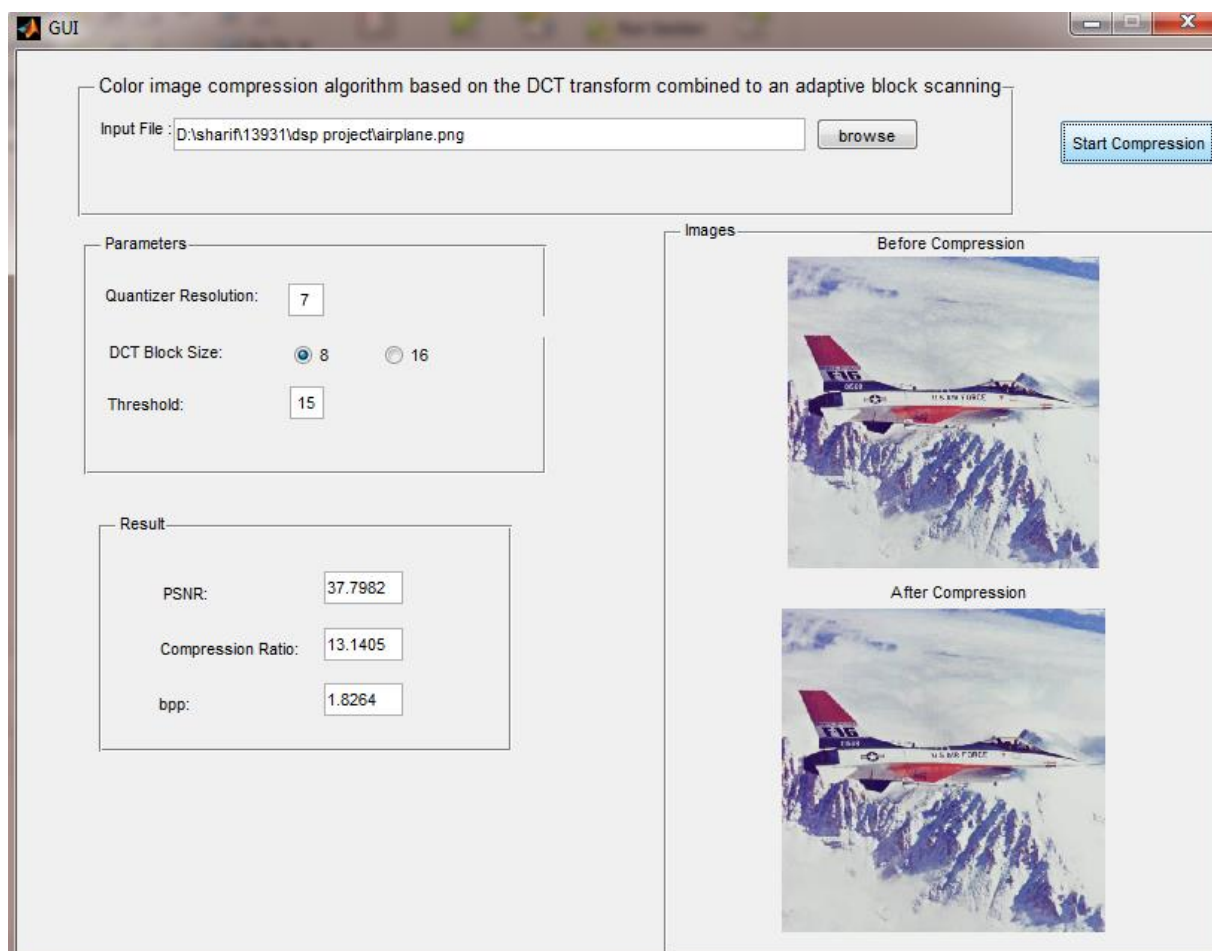
Dct block size: 16



شکل ۵

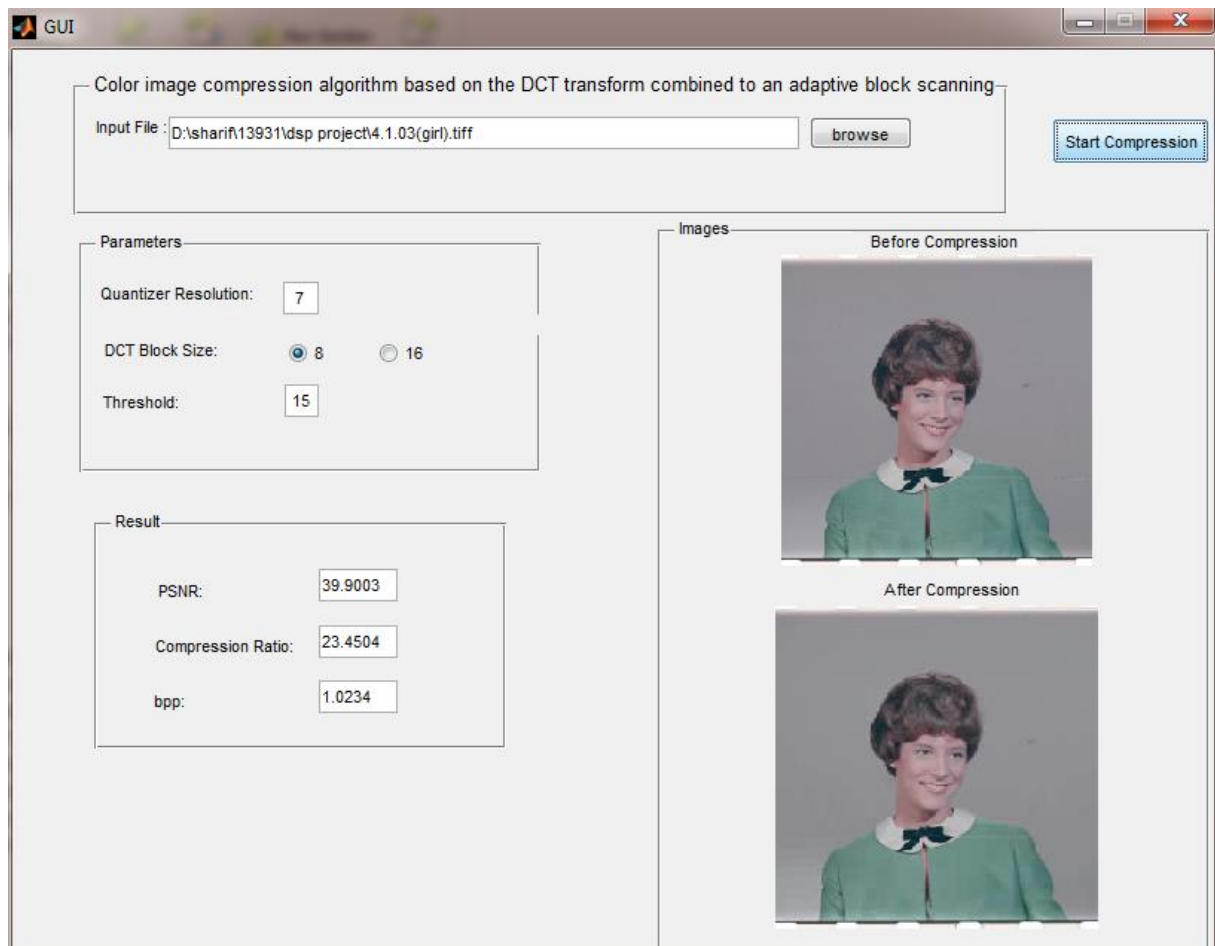
برای دو تصویر airplane و girl نیز آزمایشات بالا انجام گرفت ولی تنها برای مشاهده کیفیت تصویر بعد از فشرده سازی برای هر تصویر نتیجه یک آزمایش قرار داده شده است.

تصویر airplane :



شکل ۶

تصویر girl :



شکل ۷

۵-۲ آزمایش دوم کاهش همبستگی در حوزه ی YCbCr:

آزمایش دوم در مقاله مقایسه ی نرخ فشرده سازی در فضای RGB و YCbCr است. با اجرای این آزمایش نتایج زیر بدست می آید که همان طور که مشاهده می کنیم تفاوت بسیاری بین نرخ فشرده سازی در فضای RGB و YCbCr مشاهده می شود و علت آن تفاوت در توزیع انرژی در مؤلفه های رنگ Cr و Cb، Y است که باعث می شود Cr و Cb بیشتر فشرده شوند.

تصویر couple

The screenshot shows a MATLAB GUI window titled 'test2'. At the top, there is a 'Start' button. Below it, a section titled 'Performances in the YCbCr space & in the RGB space' contains an 'Input File' text box with the path 'D:\sharif\13931\dsp project\4.1.02(couple).tiff' and a 'Browse' button. Below this is a 'Parameters' section with 'Quantizer resolution' set to 7, 'Block size' with radio buttons for 8 and 16 (16 is selected), and 'Threshold' set to 25. At the bottom, there are two orange boxes: 'RGB space results' and 'YCbCr space results'. The RGB results show PSNR: 36.2196, CR: 13.0502, and bpp: 1.8391. The YCbCr results show PSNR: 35.4128, CR: 28.122, and bpp: 0.85342.

RGB space results	
PSNR:	36.2196
CR:	13.0502
bpp:	1.8391

YCbCr space results	
PSNR:	35.4128
CR:	28.122
bpp:	0.85342

شکل ۸

تصویر house

The screenshot shows a MATLAB GUI window titled 'test2'. At the top, there is a 'Start' button. Below it, a section titled 'Performances in the YCbCr space & in the RGB space' contains an 'Input File' text box with the path 'D:\sharif\13931\dsp project\4.1.05(house).tiff' and a 'Browse' button. A 'Parameters' section below that includes 'Quantizer resolution' set to 8, 'Block size' with radio buttons for 8 and 16 (16 is selected), and 'Threshold' set to 25. At the bottom, two orange boxes display results. The 'RGB space results' box shows PSNR: 36.7385, CR: 12.4905, and bpp: 1.9215. The 'YCbCr space results' box shows PSNR: 35.5181, CR: 23.6628, and bpp: 1.0143.

RGB space results	
PSNR:	36.7385
CR:	12.4905
bpp:	1.9215

YCbCr space results	
PSNR:	35.5181
CR:	23.6628
bpp:	1.0143

شکل ۹

نتایج مقاله:

Table 1
Energy distribution in RGB and YCbCr spaces.

Image	RSE	GSE	BSE	YSE	CbSE	CrSE
Airplane	31.901	32.668	35.431	99.253	0.477	0.270
Peppers	49.109	38.080	12.811	86.007	5.614	8.380
Lena	58.373	21.095	20.533	89.530	1.503	8.967
Girl	32.361	32.926	34.713	99.587	0.052	0.361
Couple	46.763	28.888	24.349	93.029	2.310	4.661
House	33.354	30.901	35.744	96.507	1.044	2.449
Zelda	48.265	28.757	22.978	90.936	2.603	6.461
Average	42.875	30.474	26.651	93.55	1.943	4.507

نتایج پیاده سازی:

image	RSE	GSE	BSE	YSE	CbSE	CrSE
Airplane	31.9008	32.6685	35.4307	99.2536	0.4766	0.2697
Lena	58.3727	21.0946	20.5325	89.5268	1.5031	8.9699
Girl	32.3609	32.9264	34.7127	99.5872	0.0524	0.3604
Couple	46.7630	28.8883	24.3486	93.0272	2.3104	4.6624
House	33.3544	30.9012	35.7443	96.5075	1.0440	2.4484
Zelda	48.2645	28.7570	22.9784	90.9346	2.6030	6.4623

۵-۳ آزمایش سوم اندازه گیری معیارهای کیفیت:

• فضای RGB

نتایج مقاله:

Performances in the RGB space for the different DCT block sizes and the quantizer width.

RGB nbits	7			8			9		
	PSNR	bpp	CR	PSNR	bpp	CR	PSNR	bpp	CR
Bloc 8 × 8									
Airplane	30.32	1.2241	19.606	30.349	1.3374	17.945	30.356	1.4488	16.566
Peppers	30.216	1.2457	19.266	30.244	1.3571	17.685	30.25	1.4688	16.34
Lena	32.057	1.5417	15.567	31.793	1.6002	14.998	31.801	1.7349	13.834
Girl	35.13	1.0745	22.337	35.189	1.2003	19.995	35.203	1.3033	18.414
Bloc 16 × 16									
Airplane	30.372	0.8735	27.475	30.456	0.9717	24.698	30.477	1.0639	22.558
Peppers	30.167	0.8916	26.919	30.055	0.9417	25.485	30.074	1.0329	23.235
Lena	31.959	1.1272	21.292	32.071	1.266	18.958	31.79	1.2973	18.5
Girl	35.148	0.8907	26.944	35.233	1.0032	23.924	35.26	1.1034	21.751

نتایج پیاده سازی:

RGB						
nbit	7			8		
image	PSNR	bpp	CR	PSNR	bpp	CR
Block Size (8×8)						
Airplane	34.5701	1.2199	19.6736	34.6172	1.3197	18.1864
Peppers	34.0106	1.2214	19.649	33.8402	1.2144	19.7636
Girl	37.6489	1.1044	21.7318	37.7328	1.1897	20.1735
Block Size (16×16)						
Airplane	34.663	0.8692	27.6096	34.7698	24.953	0.9618
Peppers	33.8475	0.8274	29.0057	33.3577	1.0562	22.7238
Girl	37.2168	0.60608	39.5988	37.3726	0.66573	36.0509

• فضای YCbCr

Performances in the YCbCr space for the different DCT block sizes and the quantizer width.

YCbCr nbits	7			8			9		
Image	PSNR	bpp	CR	PSNR	bpp	CR	PSNR	bpp	CR
Bloc 8 × 8									
Airplane	30.415	0.8387	28.617	30.458	0.9188	26.122	30.47	0.9984	24.038
Peppers	30.282	1.1868	20.223	30.035	1.2291	19.526	30.047	1.3286	18.064
Lena	32.075	1.1973	20.046	31.947	1.263	19.002	31.961	1.3697	17.523
Girl	35.19	0.6129	39.157	34.982	0.6537	36.715	34.995	0.7103	33.787
Bloc 16 × 16									
Airplane	30.381	0.5917	40.559	30.251	0.6247	38.417	30.282	0.6848	35.044
Peppers	30.059	0.809	29.668	30.189	0.899	26.698	30.226	0.9902	24.237
Lena	31.97	0.8101	29.625	31.94	0.8625	27.826	31.8	0.9045	26.535
Girl	35.002	0.4574	52.471	35.147	0.5096	47.092	35.181	0.5618	42.722

Ycbcr

nbit	7			8			9		
image	PSNR	bpp	CR	PSNR	bpp	CR	PSNR	bpp	CR
Block Size (8×8)									
Airplane	34.9063	0.88522	27.1118	34.9861	0.96074	24.9809	34.9883	1.0362	23.1605
Lena	35.4368	1.2005	19.9922	35.6184	1.3024	18.4281	35.6254	1.4043	17.091
Girl	37.341	0.6134	39.126	37.3085	0.6643	36.1312	37.3185	0.7150	33.5623
Block Size (16×16)									
Airplane	34.6536	0.62263	38.5459	34.8158	0.68745	34.9116	34.8724	0.75227	31.9036
Lena	35.3997	30.0434	0.7988	35.5418	0.8828	27.1847	35.6415	0.9669	24.8211
Girl	38.1152	0.4710	50.9450	38.925	0.5180	38.7434	38.291	0.565	42.4777

1. Fouzi, D., Redha, B., Nabil, B., “Color image compression algorithm based on the DCT transform combined to an adaptive block scanning”, International Journal of Electronics and Communications, Volume 65, Issue 1, January 2011.
2. Benzid, R., Marir, F., Bouguechal, N. , “Electrocardiogram compression method based on the adaptive wavelet coefficients quantization combined to a modified two-role encoder,” IEEE Signal Processing Letters, 2007.
3. Netravali, A. N., Haskell, B. G., “Digital Pictures, Representation, Compression, and Standards”, 2 nd ed., Plenum Press, 1995.
4. Storer, James A., “Data Compression: Methods and Theory” Computer Science Press, Rockville, MD, 1988.