

## فصل سوم

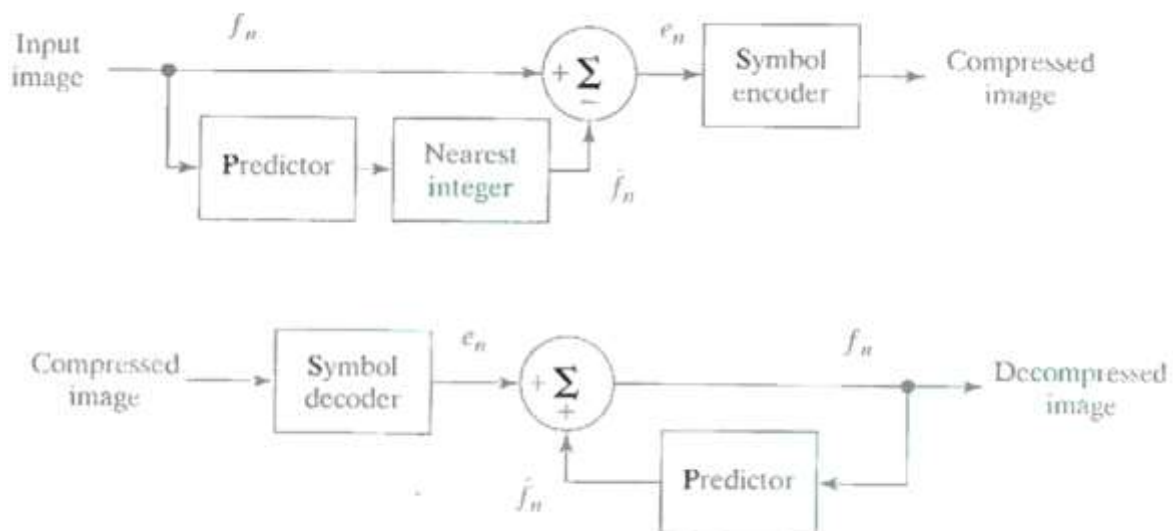
# انواع روش های فشرده سازی تصویر

تکنیک‌های فشرده‌سازی تصویر متعددی در طول سال‌ها پیشنهاد شده‌اند. فشرده‌سازی تصویر یکی از روشهای کاهش حجم آن می‌باشد و برای ذخیره‌سازی عکس‌ها از این روش استفاده می‌شود تا حجم اطلاعات و حافظه مصرفی تا جای ممکن کاهش پیدا کند. برای انجام این کار دو روش با اتلاف و بدون اتلاف کیفیت تصویر وجود دارد که به وسیله الگوریتم‌های مختلف رایانه‌ای انجام می‌شود. در ادامه برخی از روشهای فشرده‌سازی تصاویر را ذکر خواهیم کرد.

### ۳-۲ رمزنگاری پیشگویانه بدون اتلاف<sup>۱</sup>

اطلاعاتی که توسط یک پیکسل در تصویر حمل می‌شود به نسبت بسیار اندک است. درواقع می‌توان گفت بیشتر اطلاعات بصری در تصویر اضافه هستند و اکثر آنها با توجه به مقادیر همسایگانشان قابل حدس می‌باشند. در راستای کاهش افزونگی بین پیکسلی می‌توان از تفاوت بین پیکسل‌های برای نمایش تصویر استفاده نمود. که از این عمل تحت عنوان نگاشت اطلاق می‌گردد. در صورتی که اجزای تصویر اصلی با توجه به اطلاعات تبدیل یافته قابل بازسازی باشند به این تبدیل، نگاشت برگشت پذیر اطلاق می‌شود. (گونزالس، ۱۳۹۱)

یک روال ساده برای نگاشت در شکل زیر نشان داده شده است. از این روش تحت عنوان رمزنگاری پیشگویانه بدون اتلاف یاد می‌شود. این روش منجر به حذف افزونگی بین پیکسلی از طریق استخراج و کد کردن اطلاعات جدید در هر پیکسل می‌گردد. اطلاعات جدید مربوط به هر پیکسل با توجه به تفاوت بین مقادیر واقعی و پیش بینی شده هر پیکسل تعریف می‌گردد. این سیستم از دو قسمت رمزگشا و رمزنگار تشکیل میشود که هر دو شامل بخش پیشگوی مشابه می‌باشند. به ازای ورود هر یک از مقادیر پیکسل‌های ورودی که تحت عنوان  $f_n$  نشان داده شده‌اند. از آن برای محاسبه تفاوت یا خطای پیش بینی استفاده می‌شود.



شکل ۹-۰ الگوریتم پیشگویانه بدون اتلاف

<sup>۱</sup> -Lossless Predictive Coding

از روش های مختلف محلی ، سراسری یا انطباقی می توان برای تولید  $\hat{f}_n$  استفاده نمود . در اکثر این موارد عملیات پیش بینی با توجه به ترکیب خطی  $m$  پیکسل قبل به صورت زیر انجام می شود.

$$\hat{f}(x,y) = \text{round} \left[ \sum_{i=1}^m \alpha_i f(x,y-i) \right]$$

که از  $m$  تحت عنوان درجه پیش بینی کننده خطی یاد می شود. Round عدد حاصل را به نزدیک ترین عدد صحیح گرد می کند و  $\alpha_i$  ها ضرایب پیش بینی کننده نامیده می شوند.  
اگر  $m=1$  باشد از این پیش بینی کننده تحت عنوان پیش بینی کننده پیکسل قبل و از روال رمزنگاری انجام شده تحت آن تحت عنوان رمزنگاری تفاضلی یاد می شود.

### جدول ۱-۰ فشرده سازی با LPC

Original image

Image compressed with Lossless Predictive Coding (m=1)



این نوع کد گذاری تصویر درواقع پراکندگی مقادیر مختلف شدت پیکسل را کم میکند و درنتیجه اگر بخواهیم مقادیر پیکسل های تصویر را با الگوریتم هایی مانند هافمن رمز گذاری کنیم تعداد سمبل ها کاهش میابد. این موضوع را با مشاهده نمودار زیر می توان به خوبی درک کرد.

### ۳-۳ تبدیل فوریه

برای اولین بار ژوزف فوریه در سالهای ۱۸۰۶-۱۸۰۲ میلادی نظریه نمایش یک تابع برحسب حاصل جمع بی نهایت تابع سینوسی و کسینوسی را مطرح کرد. براساس این نظریه، تبدیل فوریه، یک سیگنال را به مجموعه نامتناهی از  $n$  تابع نمایی مختلط افراز می کند که هرکدام از آنها دارای فرکانس های مختلفی است. از آن جایی که تصاویر دیجیتال به صورت سیگنال گسسته هستند برای تبدیل تصویر از حوزه مکان به حوزه فرکانس از تبدیل فوریه گسسته استفاده می شود. اگر  $f(x,y)$  یک تصویر دیجیتال  $m*n$  باشد تبدیل فوریه گسسته دوبعدی آن به شکل زیر است :

$$F(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi(ux/M + vy/N)}$$

$$u = 0.1.2. \dots M-1$$

$$v = 0.1.2. \dots M-1$$

تبدیل فوریه گسسته معکوس نیز به صورت زیر تعریف می گردد.

$$F(x, y) = 1/MN \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} f(u, v) e^{-j2\pi(ux/M + vy/N)}$$

#### تبدیل فوریه سریع<sup>۲</sup>

تبدیل فوریه سریع یکی از مهم ترین الگوریتم های مورد استفاده در پردازش سیگنال و آنالیز داده است. در واقع تبدیل فوریه سریع یک الگوریتم است که برای محاسبه تبدیل فوریه گسسته و نیز معکوس آن مورد استفاده قرار می گیرد. تبدیل فوریه سریع از طریق تجزیه ماتریس تبدیل فوریه گسسته به حاصل ضرب ماتریس های تنک<sup>۳</sup> که در آنها اکثر داریه های ماتریس صفر هستند، محاسبات را تسریع می بخشد. تعداد ضرایب تبدیل فوریه سریع گسسته برابر با توانی از ۲ است و با فرمول زیر محاسبه می شود.

$$\leq 2^{\log_2 N} \text{ سریع فوریه تبدیل ضرایب}$$

$N$  : تعداد نمونه های سیگنال

با توجه به توضیحات فوق هنگامی که یک تصویر را به دامنه فرکانس تبدیل می شود، یک عدد مختلط به ازای هر پیکسل بدست می آید که می توان از آن برای دریافت اطلاعات در مورد فرکانس هایی که در ساخت تصویر استفاده شده اطلاع کسب کرد.

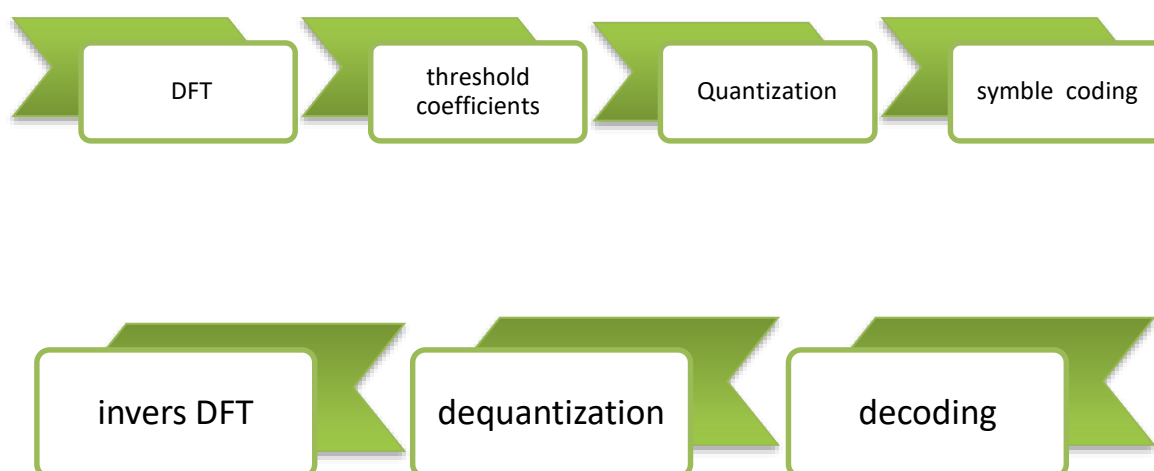
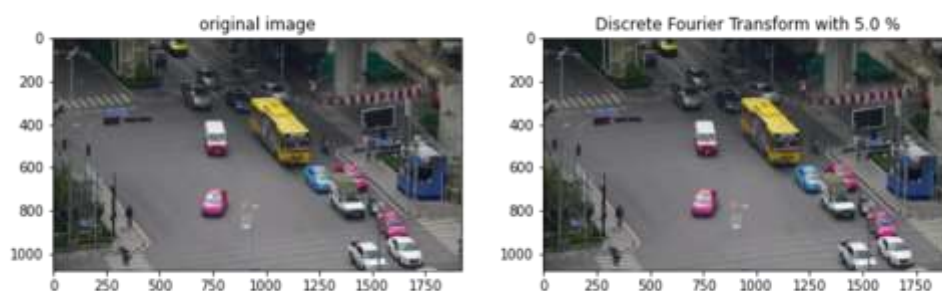
یک راه سریع و آسان برای فشرده سازی تصویر، تبدیل تصویر به فضای فرکانس، یافتن کمترین فرکانس های دامنه و دور انداختن آنها - به معنای واقعی کلمه صفر کردن عدد مختلط است. اگر مقدار کافی از آنها را دور ریخته شود، برای توصیف محتوای فرکانس یک تصویر، داده کمتری نسبت به پیکسل های تصویر می گیرد و تصویر فشرده شده است. هرچه در دور انداختن فرکانس ها تهاجمی تر باشد، کیفیت تصویر بیشتر کاهش می یابد. این روش یک روش فشرده سازی «باتلاف» است .

<sup>1</sup> -dft

<sup>2</sup> -fast fourier transfer (fft)

<sup>3</sup> -sparse

## جدول ۲-۰ فشرده سازی با تبدیل فوریه



## ۳-۴ تبدیل کسینوسی گسسته

تبدیل کسینوسی گسسته دنباله‌ای محدود از اعداد (داده‌ها) را به صورت مجموع توابع کسینوسی با فرکانس‌های متفاوت نمایش می‌دهد. این روش که توسط احمد ناتاراجان و راثو در سال ۱۹۷۴ میلادی توسعه یافته، فقط روی بخش‌های حقیقی یک سیگنال مختلط کار می‌کند زیرا بیشتر سیگنال‌های دنیای واقعی سیگنال‌های حقیقی هستند که اجزای موهومی ندارند. با تعمیم این تبدیل از یک بعد به دو بعد می‌توان از آن در پردازش و فشرده‌سازی تصویر استفاده نمود.

$$D(i, j) = \frac{1}{\sqrt{MN}} c(i) c(j) \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \text{pixel}(x, y) \cos \left[ \frac{(2x+1)i\pi}{2M} \right] \cos \left[ \frac{(2y+1)j\pi}{2N} \right]$$

$$c(u) = \begin{cases} 1/\sqrt{2} & \text{if } u = 0 \\ 1 & \text{if } u > 0 \end{cases}$$

در فرمول بالا  $pixel(x,y)$  مقدار روشنایی تصویر در پیکسلی به مختصات  $(x,y)$  است و  $N$  سایز بلوکی است که تبدیل کسینوسی گسسته با آن اجرا می شود.

معکوس تبدیل کسینوسی گسسته به صورت زیر است :

$$pixel(x,y) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} c(i)c(j)D(i,j) \cos\left[\frac{(2x+1)i\pi}{2M}\right] \cos\left[\frac{(2y+1)j\pi}{2N}\right]$$

تبدیل فوری گسسته و تبدیل کسینوس گسسته عمل مشابهی را انجام می دهند: هر دو یک بردار زمان گسسته با طول محدود را به مجموع توابع پایه مقیاس شده و تغییر یافته تجزیه می کنند. تفاوت بین این دو، نوع تابع پایه است که توسط هر تبدیل استفاده می شود. تبدیل فوری گسسته از مجموعه ای از توابع نمایی مختلط مرتبط با هارمونیک استفاده می کند، در حالی که تبدیل کسینوس گسسته فقط از توابع کسینوس (فقط قسمت حقیقی) استفاده می کند. تبدیل فوری گسسته به طور گسترده ای برای کاربردهای تحلیل طیفی عمومی وسیعی از زمینه ها پیدا استفاده می شود. همچنین به عنوان زیربنای روش هایی استفاده می شود که از ویژگی های نمایش دامنه فرکانس سیگنال ها، مانند الگوریتم های کانولوشن سریع همپوشانی-ذخیره و همپوشانی-افزودن استفاده می کنند. خاصیتی از تبدیل کسینوس گسسته که آن را برای فشرده سازی کاملاً مناسب می کند، درجه بالای "تراکم طیفی" آن است. نمایش تبدیل کسینوس گسسته سیگنال در مقایسه با تبدیل های دیگر مانند تبدیل کسینوس گسسته، انرژی بیشتری را در تعداد کمی از ضرایب متمرکز میکند. این ویژگی برای یک الگوریتم فشرده سازی بسیار مطلوب است زیرا اگر بتوان سیگنال اصلی (حوزه زمانی یا مکانی) را با استفاده از مجموعه نسبتاً کوچکی از ضرایب تبدیل کسینوس گسسته نشان داد، می توان تنها با ذخیره خروجی های تبدیل کسینوس گسسته که حاوی مقادیر قابل توجهی انرژی هستند، میزان حجم لازم برای ذخیره سازی داده ها را کاهش داد. (gent) فرآیند :

اگر تصویر چند کاناله داریم، باید الگوریتم را به صورت جداگانه برای هر کانال اعمال کرد. قبل از اینکه بتوان پردازش تبدیل کسینوس گسسته داده را انجام داد، باید تصویر RGB را به فرمت YCbCr معادل تبدیل کرد. مرحله مهم دیگر در اینجا تغییر محدوده مقادیر پیکسل از -۱۲۸ به ۱۲۷ به جای ۰ تا ۲۵۵ است که محدوده مقادیر استاندارد برای تصاویر ۸ بیتی است.

تبدیل کسینوس گسسته داده ها (پیکسل های تصویر) را به مجموعه ای از فرکانس ها تبدیل می کند. اولین فرکانس در مجموعه با معنی ترین آنها است و به همین ترتیب تا آخرین فرکانس موجود در مجموعه. کم معنی ترین فرکانس ها را می توان بر اساس نرخ موردنظر برای فشرده سازی حذف کرد. (صفر شوند) فشرده سازی تصویر مبتنی بر تبدیل کسینوس گسسته به دو تکنیک برای کاهش داده های مورد نیاز برای نمایش تصویر متکی است. اولین مورد، کوانتیزه کردن ضرایب تبدیل کسینوس گسسته تصویر است. دومی کدگذاری آنتروپی ضرایب کوانتیزه شده است.

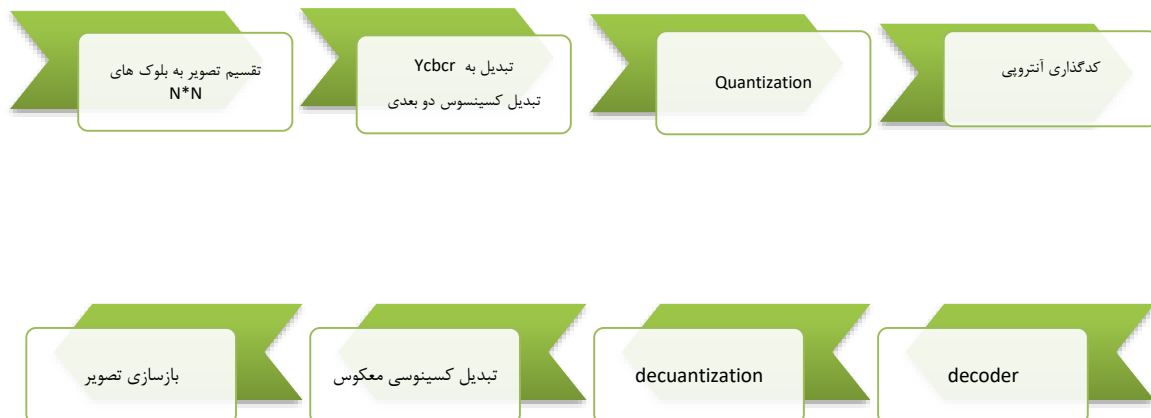
کوانتیزاسیون فرآیند کاهش تعداد مقادیر ممکن یک کمیت، در نتیجه کاهش تعداد بیت های مورد نیاز برای نشان دادن آن است. کوانتیزاسیون یک فرآیند با اتلاف است و این به معنای کاهش اطلاعات رنگ مرتبط با هر پیکسل در تصویر است.

<sup>1</sup> - overlap-save fast convolution

<sup>2</sup> - overlap-add fast convolution

کدگذاری آنتروپی یک تکنیک است برای نمایش ضرایب کوانتیزه شده به صورت هرچه کامل تر که ممکن است. تبدیل کسینوس گسسته اغلب در برنامه های فشرده سازی داده های با اتلاف استفاده می شود.

### جدول ۳-۰ فشرده سازی با DCT



## ۵-۳. Jpeg

JPEG که مخفف Joint Photographic Experts Group می باشد یک گروه متخصص ISO/IEC است که استانداردهایی را برای مجموعه ای از الگوریتم های فشرده سازی برای فایل های تصویری دیجیتالی ایجاد و حفظ می کنند. این عبارت همچنین اصطلاحی برای هر فایل تصویری گرافیکی است که با استفاده از استاندارد JPEG تولید می شود. JPEG یک روش متداول برای فشرده سازی با اتلاف برای تصاویر دیجیتال، به ویژه برای آن دسته از تصاویر تولید شده توسط عکاسی دیجیتال است. این روش یک مصالحه بین حجم ذخیره سازی و کیفیت تصویر ایجاد می کند. JPEG یکی از فرمت های فایل تصویری است که در شبکه جهانی وب پشتیبانی می شود و از آن جایی که کاهش حجم داده در اینترنت بسیار مهم است این روش فشرده سازی به یکی رایج ترین فرمت های تصویر در دنیای وب تبدیل شده است . JPEG معمولاً فشرده سازی ۱۰:۱ را با کاهش قابل توجه کمی در کیفیت تصویر انجام می دهد. با وجود همه مزایای این روش، JPEG برای نقاشی های خطی و سایر گرافیک های متنی یا نمادین مناسب نیست، زیرا تضادهای شدید بین پیکسل های مجاور می تواند باعث ایجاد مصنوعات قابل توجهی شود. به علاوه با توجه به این که این روش فشرده سازی جزو روش های فشرده سازی با اتلاف محسوب میشود و با این روش نمیتوان تصویر اصلی را بازتولید کرد بنابراین jpeg برای ذخیره سازی تصاویر پزشکی و تصویر بردار های علمی مناسب نیست. JPEG همچنین برای فایل هایی که چندین ویرایش می شوند مناسب نیست، زیرا با هر بار فشرده سازی مجدد تصویر، مقداری کیفیت تصویر از بین می رود، به ویژه اگر تصویر برش داده شود یا جابجا شود، یا اگر پارامترهای رمزگذاری تغییر کرده باشند. فرآیند :

JPEG از یک فرم فشرده سازی با اتلاف بر اساس تبدیل کسینوس گسسته استفاده می کند. این عملیات ریاضی، هر فریم/فیلد منبع ویدیویی را از حوزه فضایی (۲ بعدی) به حوزه فرکانس (معروف به دامنه تبدیل) تبدیل می کند. نمایش رنگ ها در تصویر از RGB به YCBCR تبدیل می شود.

وضوح داده های کروما معمولاً با ضریب ۲ یا ۳ کاهش می یابد. این بسیار مفید است زیرا چشم انسان در دیدن تغییرات روشنایی با فرکانس بالا در یک منطقه کوچک بسیار بد است، بنابراین ما می توانیم اساساً میزان فرکانس را کاهش دهیم و چشم انسان قادر به تشخیص تفاوت نخواهد بود.

JPEG یک تصویر را به تکه هایی از بلوک های  $8 \times 8$  پیکسل (به نام حداقل واحد کدگذاری) تبدیل می کند، محدوده مقادیر پیکسل ها را طوری تغییر می دهد که روی ۰ متمرکز شوند و سپس تبدیل کسینوس گسسته را برای هر بلوک اعمال می کند تبدیل کسینوسی گسسته برای هر یک از اجزای یک پیکسل به طور جداگانه اعمال می شود. خروجی اعمال تبدیل کسینوسی گسسته یک ماتریس ضریب  $8 \times 8$  است که به ما می گوید هر تابع کسینوس (از مجموع ۶۴ تابع) چقدر در ماتریس ورودی  $8 \times 8$  مشارکت دارد. ماتریس ضرایب یک تبدیل کسینوسی گسسته عموماً حاوی مقادیر بزرگتر در گوشه سمت چپ بالای ماتریس ضریب و مقادیر کوچکتر در گوشه پایین سمت راست است. گوشه سمت چپ بالا نشان دهنده تابع کسینوس کمترین فرکانس و پایین سمت راست نشان دهنده تابع کسینوس بالاترین فرکانس است. این بیانگر این است که اکثر تصاویر حاوی مقدار زیادی اطلاعات با فرکانس پایین و مقدار کمی اطلاعات با فرکانس بالا هستند. اگر مولفه های پایین سمت راست هر ماتریس تبدیل کسینوسی گسسته را به ۰ تبدیل کنیم، تصویر حاصل همچنان یکسان به نظر می رسد زیرا همانطور که اشاره شد، انسان ها در مشاهده تغییرات فرکانس بالا بد هستند.

بر روی بلوک حاصل از تبدیل کسینوسی گسسته باید کوانتیزسیون انجام شود. بسته به سطح فشرده سازی مورد نیاز، برخی از ماتریس های کوانتیزسیون رایج استفاده می شود. عنصر ماتریس ضریب تبدیل کسینوسی گسسته را با ماتریس کوانتیزسیون تقسیم می کنیم، نتیجه را به یک عدد صحیح گرد می کنیم و ماتریس کوانتیزه شده را بدست می آوریم. در این ماتریس کوانتیزه، اولین مقدار را مقدار دی سی و بقیه مقادیر را مقادیر ای سی می نامند. اگر بخواهیم مقادیر دی سی

1 - MCU



را از همه ماتریس‌های کوانتیزه‌شده بگیریم و یک تصویر جدید تولید کنیم، اساساً با یک تصویر کوچک با وضوح ۸/۱ از تصویر اصلی مواجه می‌شویم.

یک ماتریس کوانتیزسیون رایج به شکل زیر است :

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

پس از کوانتیزه کردن، JPEG از رمزگذاری زیگ زاگ برای تبدیل ماتریس به آرایه یک بعدی استفاده می‌کند. این رمزگذاری ترجیح داده می‌شود زیرا بیشتر اطلاعات فرکانس پایین (مهمترین) در ابتدای ماتریس پس از کوانتیزه کردن ذخیره می‌شود و رمزگذاری زیگزاگ همه آن‌ها را در ابتدای ماتریس ۱ بعدی ذخیره می‌کند. این برای فشرده سازی که در مرحله بعد اتفاق می‌افتد مفید است.

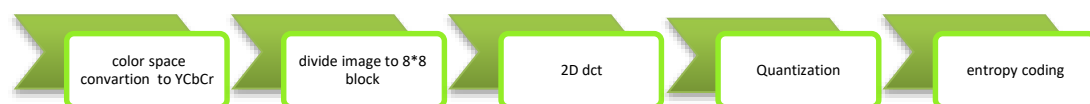
رمزگذاری طول اجرا برای فشرده سازی داده های تکراری استفاده می‌شود. با توجه به اینکه در پایان رمزگذاری زیگ زاگ، اکثر آرایه های ۱ بعدی رمزگذاری شده زیگ زاگ دارای ۰های زیادی هستند. رمزگذاری طول اجرا به ما این امکان را می‌دهد که تمام فضای تلف شده را بازیابی کنیم و از بایت های کمتری برای نشان دادن همه آن ۰ها استفاده کنیم. در یک JPEG اطلاعات تبدیل کسینوس گسسته با استفاده از رمزگذاری هافمن ذخیره می‌شود.

مراحل رمزگشایی jpeg به شکل زیر است:

جداول هافمن را استخراج کرده و بیت ها را رمزگشایی می‌شود.  
ضرایب تبدیل کسینوسی گسسته را با لغو رمزگذاری های طول اجرا و دلتا می‌شود.  
از ضرایب تبدیل کسینوسی گسسته برای ترکیب امواج کسینوس و بازسازی مقادیر پیکسل برای هر بلوک ۸×۸ استفاده می‌شود.

YCbCr را برای هر پیکسل به RGB تبدیل می‌شود.

<sup>1</sup> - Run-length encoding



جدول ۴-۰ فشرده سازی با روش jpeg

Original image



Jpeg compressiom with 10% qulity



موجک یکی از قوی ترین و پرکاربرد ترین تبدیل های ریاضی در حوزه پردازش سیگنال و تصویر است. این تبدیل به دلیل ویژگی انعطاف پذیری در نمایش سیگنال های تصویری نایستا (ایستا بودن طیف فرکانسی به این معنی است که فرکانسهای ظاهر شده در یک سیگنال، وابسته به زمان نباشند. به عبارت دیگر، اگر یک سیگنال شامل فرکانس  $X$  هرتز باشد، این فرکانس باید به صورت برابر در تمام طول سیگنال وجود داشته باشد.) و همچنین قدرت تطابق با ویژگی های بینایی انسان در زمینه پردازش تصویر به طور قابل ملاحظه ای مورد توجه قرار گرفته است. با استفاده از این تبدیل میتوان یک سیگنال تصویری را به بخش هایی که هریک متناظر با باند فرکانسی مختلف است تقسیم کرد.

یکی از مشکلات اصلی تبدیل های معمولی (مانند تبدیل فوریه) این است که آنها حول مجموعه ای از توابع پایه با طول بینهایت مانند سینوسی یا نمایی مختلط ساخته شده اند این بدان معنی است که هر گونه تغییر زمانی (یا مکانی) محلی یا ویژگی ها در کل حوزه فرکانس پخش می شوند. این یک مشکل خاص برای سیگنال های غیر ثابت مانند سیگنال های معمولی تصاویر و فیلم ها است. تبدیل فوریه معکوس پذیر است و این قابلیت باعث می شود تا فرآیند تبدیل سیگنال خام به سیگنال پردازش شده و بالعکس امکان پذیر باشد. اما در این میان (تبدیل به حوزه فرکانسی) اطلاعات حوزه زمان از بین می رود و درک زمان دقیق این رویداد امکان پذیر نیست. اگر سیگنال ایستا باشد این چندان اهمیتی ندارد زیرا مولفه های فرکانسی در تمام زمان ها وجود دارد اما در مورد سیگنال های نایستا تحلیل فوریه ابزار مناسبی نیست. یکی از راه هایی که برای غلبه بر این مشکل پیشنهاد شده تبدیل فوریه زمان کوتاه است. این ایده به این صورت است که یک سیگنال نایستا را به بخش های کوچکی که هر بخش را بتوان یک سیگنال ایستا در نظر گرفت تقسیم کرد. این فرآیند با استفاده از روش پنجره گذاری امکان پذیر است. نکته این است که در تبدیل فوریه زمان کوتاه طول تابع پنجره ثابت است.

اما این روش مشکلات اضافی را به دلیل اثرات مرزی و اعوجاج لبه ایجاد می کند. اینها می توانند طیف زیرین را تحریف کنند و در زمینه فشرده سازی، بازسازی کاملی از سیگنال اصلی غیر ممکن سازد. همچنین عیب اصلی که در این روش وجود دارد این است که با یک محدودیت فیزیکی در تبدیل فوریه رو به رو خواهد شد که عدم قطعیت نام دارد. در این روش، هرچه اندازه پنجره ها را کوچکتر کنیم، قادر خواهیم بود به صورت دقیقتر تعیین کنیم که یک فرکانس در چه زمانی از سیگنال اصلی به وقوع پیوسته است، اما از طرف دیگر اطلاعات کمتری را راجع به مقدار فرکانس سیگنال اصلی به دست خواهیم آورد. به صورت مشابه، هر چه اندازه پنجره ها را بزرگتر انتخاب کنیم، اطلاعات بیشتری راجع به مقدار فرکانس و اطلاعات کمتری راجع به زمان وقوع فرکانس به دست خواهیم آورد.

تبدیل موجک شیوه پنجره گذاری با اندازه متغیر است. در تحلیل موجک امکان دسترسی به اطلاعات حوزه فرکانس و زمان به طور همزمان وجود دارد. تبدیل موجک سیگنال را به حوزه فرکانس تبدیل نمیکند بلکه به حوزه زمان-مقیاس تبدیل میکند. (Song, 2008)

رزولوشن تبدیل موجک به صورت زیر تغییر میکند:

برای مقادیر فرکانسهای کوچک، رزولوشن بالا در حوزه فرکانس و رزولوشن پایین در حوزه زمان دارد.

برای مقادیر فرکانسهای بالا، رزولوشن پایین در حوزه فرکانس و رزولوشن بالا در حوزه زمان دارد.

نحوه عملکرد تبدیل موجک

تبدیل فوریه برای آنالیز سیگنال از یک سری امواج سینوسی با فرکانسهای مختلف استفاده میکند. در این حالت، سیگنال به صورت ترکیبی خطی از سیگنالهای سینوسی نمایش داده میشود. اما تبدیل موجک از تعدادی توابع به نام موجک استفاده میکند که هر کدام مقیاس متفاوتی دارند. همان طور که میدانیم معنی واژه موجک، موج کوچک است و توابع موجک نیز دقیقاً به همین صورت کوچک هستند.

<sup>1</sup> - Short time Fourier transform

چون موجک در زمان واقع شده است، در نتیجه میتوان سیگنال اصلی را در لحظات مختلف از زمان در موجک ضرب کرد. در گام نخست، با نقاط ابتدایی سیگنال شروع میکنیم و به تدریج موجک را به سمت انتهای سیگنال حرکت میدهیم. این عمل را کانولوشن میگویند. بعد از این که کانولوشن را با سیگنال موجک اصلی (موجک مادر) انجام دادیم، میتوانیم آن را به نحوی مقیاسدهی کنیم که بزرگتر شود و دوباره فرایند را تکرار کنیم.

چون فقط دو شرط ریاضی وجود دارد که تابع موجک باید در آن صدق کند، در نتیجه تولید انواع مختلف موجک کار آسانی است. دو شرط ریاضی که موجک باید در آنها صدق کند، قیود متعامدسازی و نرمالسازی نام دارد. بر اساس این قیود، یک موجک باید اولاً انرژی محدود داشته باشد و دوماً میانگین آن برابر با صفر باشد.

تبدیل موجک پیوسته یک سیگنال یک بعدی مانند  $x(t)$  به شرح زیر است

$$CWT_x^\psi(a, b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} x(t) \psi^*\left(\frac{t-b}{a}\right) dt$$

در فرمول بالا  $\psi(t)$  موجک مادر پیوسته است که توسط فاکتور  $a$  مقیاس و توسط فاکتور  $b$  منتقل شده است. مقدار فاکتور مقیاس دهی و جابه جایی اعدادی حقیقی هستند بنابراین تعداد بیشمار موجک وجود دارد. می توان موجک مادر را توسط  $1/3$  یا  $1/113$  یا هر عدد دیگر مقیاس کرد.

اما زمانی که راجع به تبدیل موجک گسسته بحث میکنیم، تفاوت اصلی که وجود دارد این است که از مقادیر گسسته برای فاکتور مقیاسدهی و جابه جایی در این تبدیل استفاده میشود فاکتور مقیاس به صورت اعداد توان دو افزایش می یابد بنابراین  $a=1,2,4,\dots$  خواهد بود. فاکتور جابه جایی به صورت اعداد صحیح افزایش می یابند بنابراین  $b=1,2,3,\dots$  خواهد بود. نکته مهمی که وجود دارد این است که تبدیل موجک گسسته فقط در دو حوزه مقیاس و جابه جایی گسسته است و در حوزه زمان گسسته نیست.

تبدیل موجک گسسته<sup>۵</sup>

به منظور امکان پذیر بودن عملیات محاسباتی توسط یارانه نیازمند بدست آوردن تبدیل موجک گسسته هستیم. اصول تبدیل موجک گسسته بر مبنای روشی تحت عنوان کدینگ زیرباند که توسط *crochiere* و همکاران در سال ۱۹۷۶ میلادی ارائه شد.

در حالت گسسته فیلترهایی با فرکانس های قطع مختلف برای تحلیل سیگنال در مقیاس های متفاوت مورد استفاده قرار می گیرند. با عبور سیگنال از فیلترهای بالاگذر و پایین گذر، فرکانس های مختلف آن تحلیل می شود. در حالت گسسته رزولوشن سیگنال توسط عملکرد فیلترها کنترل می شود. مقیاس نیز از طریق نمونه افزایشی یا نمونه کاهی تغییر می کند. روند پردازش تبدیل موجک گسسته چنین است که ابتدا سیگنال از فیلتر پایین گذر نیم باند با پاسخ ضربه  $h[n]$  عبور میکند. خروجی فیلتر، حاصل اعمال پیچش ورودی و پاسخ ضربه فیلتر است. در نتیجه این عمل فیلترینگ، تمام مولفه

<sup>1</sup> - Convolution

<sup>2</sup> - Orthogonalization

<sup>3</sup> - Normalization

<sup>4</sup> - continuous wavelet transform (cwt)

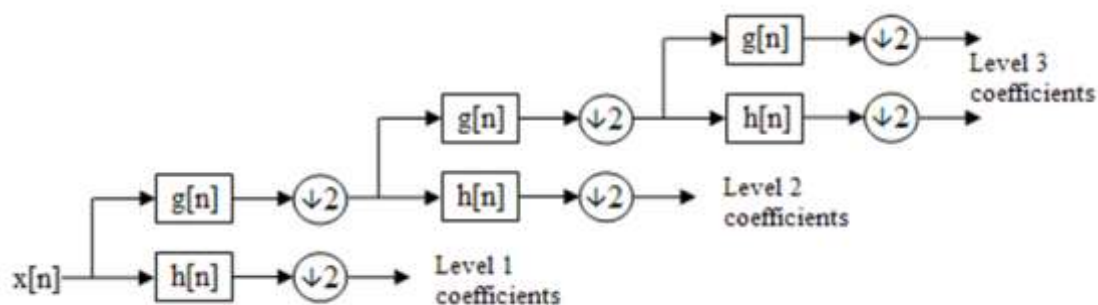
<sup>5</sup> - discrete wavelet transform (DWT)

های فرکانسی که از نصف بزرگترین فرکانس موجود در سیگنال ، بزرگتر باشد حذف می شوند. بنابراین نیمی از نمونه ها قابل حذف هستند. لذا با حذف یک در میان نمونه ها بدون اینکه اطلاعاتی را از دست داده باشیم طول سیگنال نصف خواهد شد. روند مشابهی نیز با استفاده از یک فیلتر بالاگذر نیم باند با پاسخ ضربه  $g[n]$  انجام می پذیرد. سیگنال فرکانس پایین و بالا حاصل از تجزیه سیگنال اصلی را می توان از طریق معادلات زیر بدست آورد :

$$y_{high}[n] = \sum_n x[n] \cdot g[2k - n]$$

$$y_{low}[n] = \sum_n x[n] \cdot h[2k - n]$$

لذا رزولوشن زمانی نصف و در مقابل رزولوشن فرکانسی دو برابر می شود. این روند را می توان مجدداً بر روی سیگنال فرکانس پایین بدست آمده اعمال کرد و در هر مرحله با کاهش رزولوشن زمانی به میزان نصف مرحله قبل ، رزولوشن فرکانسی را دو برابر کرد. این ایده به روش بانک فیلتر<sup>۱</sup> مشهور است.



شکل ۱۰-۱۰ دیگرام نمونه برداری در موجک

در تبدیل موجک گسسته ، یک سیگنال تصویر از طریق تجزیه و تحلیل بانک فیلتری بدست می آید. بانک فیلتری در هر مرحله از فرآیند تجزیه از یکسری فیلتر های بالا گذر و پایین گذر تشکیل شده است. وقتی سیگنال تصویر از این تصویر ها عبور می کند به دو باند فرکانسی متمایز تقسیم می شود. فیلتر پایین گذر اطلاعات کلی سیگنال را استخراج می کند در حالی که فیلتر بالاگذر اطلاعات جزئی سیگنال را استخراج می کند. خروجی مرحله فیلترینگ با استفاده از دو تبدیل یک بعدی مجدداً خود به سیگنال تقسیم می شود. این فرآیند به این صورت ادامه می یابد که ابتدا سیگنال به صورت ردیفی فیلتر می شود و سپس به صورت ستونی هر سیگنال خود به دو سیگنال دیگر تجزیه می گردد. این عملیات تصویر را به چهار باند مختلف به نام های  $LL, LH, HL, HH$  تقسیم میکند. این فرآیند را تجزیه موجک<sup>۲</sup> می نامند.

انواع موجک ها :

خانواده های موجک براساس ویژگی های زیر به شش خانواده کلی تقسیم می شوند:

- پشتیبانی موجک در حوزه زمان و فرکانس

- تقارن یا عدم تقارن موجک

- یکنواختی موجک (هرچه موجک یکنواخت تر باشد رزولوشن فرکانسی بیشتر است)

- وجود تابع مقیاس  $\varphi$

<sup>1</sup> -filter bank

<sup>2</sup>

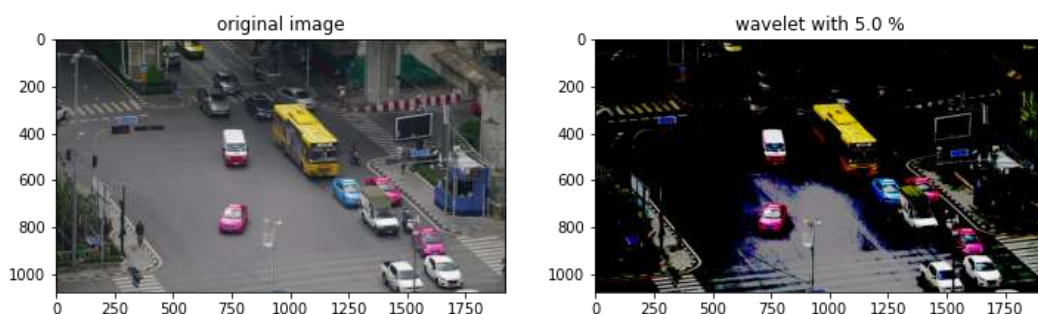
خانواده موجک	موجک های مربوطه
Daubechies	'db1' or 'haar' 'db2'.....'db10'...'db45'
Coiflets	'cofi1',.....'cofi45'
Symlets	'sym2',.....'sym10',.....'sym45'
Discrete Meyer	'dmey'
Biorthogonal	'bior2.4', 'bior2.2', 'bior1.5', 'bior1.3', 'bior1.1', 'bior2.6'.....,
Reverse Biorthogonal	'rbior1.1', 'rbior1.3', 'rbior1.5', 'rbior2.2', 'rbior2.4', 'rbior2.6', 'rbior2.8', 'rbior3.1', 'rbior3.3',....

یک موجک خاص ممکن است نمایش پراکنده تری از سیگنال نسبت به دیگری ایجاد کند، بنابراین انواع مختلف موجکها باید بررسی شوند تا ببینیم کدام یک برای فشرده سازی تصویر مناسب تر است. فرآیند:

ابتدا یک تصویر را به ضرایبی به نام باندهای فرعی تجزیه می شود و سپس ضرایب به دست آمده را با یک آستانه مقایسه خواهد شد. ضرایب زیر آستانه صفر تعیین می گردد. در نهایت، ضرایب بالاتر از مقدار آستانه با تکنیک فشرده سازی با اتلاف کدگذاری می شوند.



## جدول ۵-۰ فشرده سازی با روش موجک



## ۳-۷ هار

موجک هار سری خاصی از توابع است که اکنون به عنوان اولین موجک شناخته می‌شود. این سری اولین بار توسط آلفرد هار، ریاضیدان مجاری در سال ۱۹۰۹ پیشنهاد شد. موجک هار ساده‌ترین موجک ممکن می‌باشد. مشکل این موجک این (Patterson) است که پیوسته نیست و در نتیجه مشتق‌پذیر نمی‌باشد. موجک مادر هار به شکل زیر تعریف می‌شود.

The Haar Wavelet Transform: Compression and Reconstruction, 2006)

$$\psi(x) = \begin{cases} 1 & \text{if } 0 < x < 1 \cdot 2 \\ 0 & \text{if } 1 \cdot 2 \leq x < 1 \\ -1 & \text{otherwise} \end{cases}$$

این تابع را می‌توان در پردازش تصویر به ویژه در فرآیند فشرده سازی استفاده کرد. فشرده سازی موجک هار روشی کارآمد برای انجام فشرده سازی تصویر بدون اتلاف و هم با اتلاف است. استفاده از موجک هار در فرآیند فشرده سازی، به دست آوردن نتیجه در سطوح مختلف (سطح ۱، سطح ۲، ... و غیره) امکان‌پذیر است. برای هر سطح، نتیجه به چهار قسمت تقسیم می‌شود: یکی نشان دهنده تصویر فشرده و بقیه حاوی جزئیات (افقی، عمودی و مورب). یک ماتریس پراکنده<sup>۱</sup> ماتریسی است که بخش بزرگی از ورودی‌های آن ۰ است. فشرده سازی موجک هار برای ایجاد یک ماتریس پراکنده از روش زیر استفاده می‌کند.

فرایند:

<sup>۱</sup> -spars matrix

ابتدا تصویر به بلوک های  $8 \times 8$  تقسیم میشود.

برای هر بلوک، ابتدا میانگین پیکسل ها با هم، به صورت زوجی، محاسبه می شود تا تصویر جدید با وضوح پایین تر با مقادیر پیکسل به دست آید. واضح است که برخی از اطلاعات در این فرآیند میانگین گیری از بین می رود. برای بازیابی مقادیر چهار پیکسل اصلی از دو مقدار متوسط، باید ضرایب جزئیات را ذخیره کنیم.

اگر  $r_1 = [88.88.89.90.92.94.96.97]$  سطر اول از ماتریس نمونه  $A_{8 \times 8}$  باشد با میانگین گیری جفت جفت

آرایه ها ۴ ستون اول را پر می کنیم و ۴ ستون بعدی با اختلاف هر آرایه با میانگین پر می شود پس داریم :

$$r_1 h_1 = [88 \quad 89 \cdot 5 \quad 93 \quad 95 \cdot 5 \quad 0 \quad -0 \cdot 5 \quad -1 \quad -0 \cdot 5]$$

۴ ورودی اول را ضرایب تقریبی<sup>۱</sup> و ۴ ورودی آخر را ضرایب جزئیات<sup>۲</sup> می نامند.

برای ۴ ورودی اول باز عمل میانگین گیری رو اختلاف را تکرار می کنیم خواهیم داشت :

$$r_1 h_1 h_2 = [88 \cdot 75 \quad 94 \cdot 75 \quad -0 \cdot 75 \quad -1 \cdot 75 \quad 0 \quad -0 \cdot 5 \quad -1 \quad -0 \cdot 5]$$

برای دو آرایه اول باز تکرار می کنیم :

$$r_1 h_1 h_2 h_3 = [91 \cdot 74 \quad -3 \quad -0 \cdot 75 \quad -1 \cdot 75 \quad 0 \quad -0 \cdot 5 \quad -1 \quad -0 \cdot 5]$$

این عملیات را برای باقی سطرهای  $A_{8 \times 8}$  تکرار می کنیم. سپس برای ستون های این ماتریس این عملیات را تکرار کرده

، نتیجه ماتریسی است که بیشتر مقادیر آن صفر یا نزدیک به صفر هستند. با آستانه گذاری این ماتریس می توان آن را

فشرده کرد.

با استفاده از ماتریس تبدیل موجک ها انجام این عملیات ها آسان تر خواهد بود. این ماتریس به صورت زیر است

$$H = \begin{bmatrix} \frac{1}{8} & \frac{1}{8} & \frac{1}{4} & 0 & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{8} & \frac{1}{8} & \frac{1}{4} & 0 & \frac{-1}{2} & 0 & 0 & 0 \\ \frac{1}{8} & \frac{1}{8} & \frac{-1}{4} & 0 & 0 & \frac{1}{2} & 0 & 0 \\ \frac{1}{8} & \frac{1}{8} & \frac{-1}{4} & 0 & 0 & \frac{-1}{2} & 0 & 0 \\ \frac{1}{8} & \frac{-1}{8} & 0 & \frac{1}{4} & 0 & 0 & \frac{1}{2} & 0 \\ \frac{1}{8} & \frac{-1}{8} & 0 & \frac{1}{4} & 0 & 0 & \frac{-1}{2} & 0 \\ \frac{1}{8} & \frac{-1}{8} & 0 & \frac{-1}{4} & 0 & 0 & 0 & \frac{1}{2} \\ \frac{1}{8} & \frac{-1}{8} & 0 & \frac{-1}{4} & 0 & 0 & 0 & \frac{-1}{2} \end{bmatrix}$$

$$B = H^T A H$$

<sup>1</sup> - Approximate coefficients

<sup>2</sup> - Detail coefficients



B تصویر فشرده A را نشان می دهد .

مزایای هار:

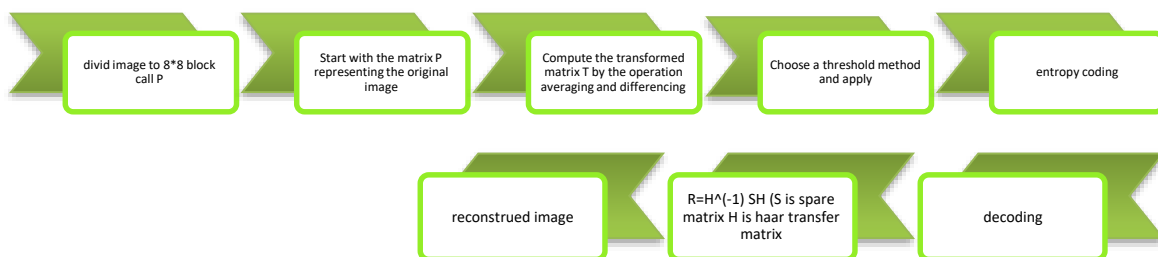
□ از نظر مفهومی ساده و سریع است

□ موجب صرفه جویی در مموری است ، زیرا می توان آن را بدون آرایه موقت در محاسبه کرد

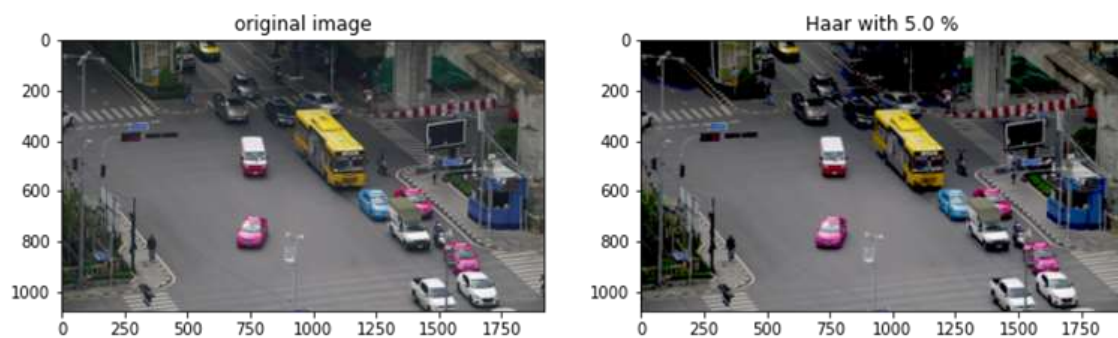
□ کاملاً قابل معکوس شدن است بدون اثر لبه ها که یک مشکل با سایر تبدیل های موجک است.

□ نسبت فشرده سازی بالا و PSNR بالا را فراهم می کند(نسبت پیک سیگنال به نویز).

□ جزئیات را به صورت بازگشتی افزایش می دهد.



جدول ۶-۰ فشرده سازی با روش هار



## ۸-۳ آنالیز مولفه اساسی

تحلیل مولفه اساسی به بیان ساده، روشی برای استخراج متغیرهای مهم (به شکل مولفه) از مجموعه بزرگی متغیرهای موجود در یک مجموعه داده است. (huffman, 2021)

در آنالیز مولفه اساسی، هدف یافتن الگوهای مختلف در داده‌های مورد نظر است؛ به طور دقیق‌تر، در این آنالیز سعی بر آن است که همبستگی میان داده‌ها به دست آید. اگر در دو یا چند بعد مشخص، بین داده‌های ما، همبستگی شدیدی وجود داشته باشد، می‌توان آن‌ها را به یک بعد تبدیل نمود؛ به این ترتیب و با کاهش بعد، پیچیدگی و حجم داده‌ها به شدت کاهش می‌یابد. اگر بخواهیم هدف غایی آنالیز مولفه اساسی را بیان کنیم، باید بگوییم: هدف پیدا کردن جهت (بعد) با بیش‌ترین واریانس داده‌ها و کاهش بعد است؛ به صورتی که کم‌ترین میزان داده‌ی بااهمیت از دست رود. مراحل آنالیز مولفه اساسی به صورت زیر تشریح می‌شود:

### ○ استاندارد سازی

هدف از این مرحله استانداردسازی دامنه متغیرهای اولیه است به طوری که هر یک از آنها به طور مساوی در تجزیه و تحلیل مشارکت داشته باشند. به طور خاص، به این دلیل انجام استانداردسازی قبل آنالیز مولفه اساسی بسیار مهم است، که داده‌ها در مراحل بعد نسبت به واریانس متغیرهای اولیه کاملاً حساس هستند. به این معنا که اگر بین دامنه متغیرهای اولیه تفاوت زیادی وجود داشته باشد، آن دسته از متغیرهایی که دامنه بزرگتر دارند بر متغیرهایی با دامنه کوچک تسلط خواهند داشت (مثلاً متغیری که بین ۰ تا ۱۰۰ است بر متغیری که بین ۰ تا ۱ است تسلط خواهد داشت). ( که منجر به نتایج مغرضانه خواهد شد. بنابراین، تبدیل داده‌ها به مقیاس‌های قابل مقایسه می‌تواند از این مشکل جلوگیری کند. از نظر ریاضی، این کار را می‌توان با کم کردن میانگین و تقسیم بر انحراف استاندارد برای هر مقدار از هر متغیر انجام داد.

$$z = \frac{value - mean}{standard deviation}$$

### ○ محاسبه ماتریس کوواریانس

هدف این مرحله این است که بفهمیم متغیرهای مجموعه داده ورودی چگونه از میانگین نسبت به یکدیگر متفاوت هستند یا به عبارت دیگر، ببینیم آیا رابطه‌ای بین آنها وجود دارد یا خیر. زیرا گاهی اوقات متغیرها به نحوی

همبستگی بالایی دارند که حاوی اطلاعات اضافی هستند. بنابراین، برای شناسایی این همبستگی ها، ماتریس کوواریانس را محاسبه می کنیم. ماتریس کوواریانس یک ماتریس متقارن  $p \times p$  است (که در آن  $p$  تعداد ابعاد است) که به عنوان ورودی، کوواریانس های مرتبط با همه جفت های ممکن متغیرهای اولیه را دارد.

از آنجایی که کوواریانس یک متغیر با خودش واریانس آن است  $Cov(a,a)=Var(a)$ ، در قطراصلی در واقع واریانس های هر متغیر اولیه را داریم. و از آنجایی که کوواریانس جابجایی پذیر است است  $Cov(a,b)=Cov(b,a)$ ، ورودی های ماتریس کوواریانس نسبت به قطر اصلی متقارن هستند.

این در واقع علامت کوواریانس است که اهمیت دارد:

اگر مثبت باشد، این دو متغیر با هم افزایش یا کاهش می یابند (همبسته)

اگر منفی باشد: یکی افزایش می یابد زمانی که دیگری کاهش می یابد (همبستگی معکوس)

○ محاسبه بردار و مقادیر ویژه ماتریس کواریانس

بردارهای ویژه و مقادیر ویژه مفاهیم جبر خطی هستند که برای تعیین اجزای اصلی داده ها باید از ماتریس کوواریانس محاسبه کنیم. اجزای اصلی متغیرهای جدیدی هستند که به صورت ترکیب خطی یا مخلوطی از متغیرهای اولیه ساخته می شوند. این ترکیبها به گونه ای انجام می شوند که متغیرهای جدید (یعنی اجزای اصلی) همبستگی ندارند و بیشتر اطلاعات درون متغیرهای اولیه فشرده می شوند. نکته مهمی که در اینجا باید ذکر شود این است که اجزای اصلی کمتر قابل تفسیر هستند و هیچ معنای واقعی ندارند زیرا به صورت ترکیب خطی از متغیرهای اولیه ساخته شده اند. از نظر هندسی، مؤلفه های اصلی نشان دهنده جهت های داده ها هستند که حداکثر مقدار واریانس را توضیح می دهند. رابطه بین واریانس و اطلاعات در اینجا به این صورت است که هر چه واریانس حمل شده توسط یک خط بزرگتر باشد، پراکندگی نقاط داده در امتداد آن بیشتر است و هر چه پراکندگی در امتداد یک خط بزرگتر باشد، اطلاعات بیشتری دارد.

بردارهای ویژه ماتریس کوواریانس در واقع جهت محورهایی هستند که در آن بیشترین واریانس (بیشترین اطلاعات) وجود دارد و آنها را مولفه های اصلی می نامند. و مقادیر ویژه صرفاً ضرایبی هستند که به بردارهای ویژه متصل می شوند، که مقدار واریانس حمل شده در هر جزء اصلی را نشان می دهد. با رتبه بندی بردارهای ویژه به ترتیب مقادیر ویژه، از بالاترین به پایین ترین، مولفه های اصلی را به ترتیب اهمیت به دست می آید.

○ بردار ویژگی

حاسبه بردارهای ویژه و مرتب کردن آنها بر اساس مقادیر ویژه آنها به ترتیب نزولی، به ما امکان می دهد تا اجزای اصلی را به ترتیب اهمیت پیدا کنیم. در این مرحله، کاری که انجام می دهیم این است که انتخاب کنیم آیا همه این مؤلفه ها را حفظ کنیم یا مؤلفه هایی را که اهمیت کمتری دارند (مقادیر ویژه کم) را کنار بگذاریم و با بقیه ماتریسی از

بردارها تشکیل دهیم که آن را بردار ویژگی می‌نامیم. بنابراین، بردار ویژگی صرفاً یک ماتریس است که بردارهای ویژه اجزایی را که ما تصمیم به حفظ آنها داریم به عنوان ستون دارد. این باعث می‌شود که اولین گام به سمت کاهش ابعاد باشد، زیرا اگر ما انتخاب کنیم که فقط  $p$  مقدار ویژه را از  $n$  تا نگه داریم، مجموعه داده‌های نهایی فقط ابعاد  $p$  خواهد داشت.

○ بازنویسی داده‌ها را در امتداد محورهای اجزای اصلی

در این مرحله، که آخرین مرحله است، هدف استفاده از بردار ویژگی است که با استفاده از بردارهای ویژه ماتریس کوواریانس تشکیل شده است تا داده‌ها را از محورهای اصلی به محورهایی که توسط مؤلفه‌های اصلی نشان داده می‌شوند. این را می‌توان با ضرب انتقال مجموعه داده‌های اصلی در انتقال بردار ویژگی انجام داد.

$$Final\ Data = Feature\ Vector^T * StandardizedOriginalData^T$$

### آنالیز مؤلفه اصلی در فشرده سازی تصاویر

ما از آنالیز مؤلفه اساسی برای کاهش اندازه تصویر با انتخاب تعداد معینی از مؤلفه‌های اصلی برای استفاده می‌کنیم تا فقط پیکسل‌های مهم را برای حفظ واریانس تصویر اصلی ذخیره شوند.

اگر تصویر رنگی داشته باشیم تصویر اصلی ما از سه کانال رنگی تشکیل شده است: قرمز، سبز و آبی. برای هر کانال رنگی، پیکسل‌ها را به عنوان یک ماتریس دو بعدی (ارتفاع) و (عرض) در نظر می‌گیریم. آنالیز مؤلفه اساسی بر روی هر کانال رنگی انجام می‌شود، که منجر به یک طرح مؤلفه اساسی و اجزای اصلی (محور) می‌شود که هر دو به شکل یک ماتریس با شکل طول \* عرض تصویر اصلی خواهند بود.

تعداد مؤلفه‌های اساسی که انتخاب می‌شوند رزولوشن تصویر فشرده شده را مشخص می‌کند.

با استفاده از فرمول بازسازی ماتریس مربوط به هر کانال بازسازی و باهم ترکیب می‌شوند و یک ماتریس ۳ بعدی تشکیل داده می‌شود نتیجه تصویر رنگی فشرده شده خواهد بود.

$$Reconstruction = projection \cdot EigenVectort^T$$

برای فشرده سازی موثر تصاویر به چند جزء اصلی نیاز است؟

تصاویر مختلف را می‌توان با سطوح مصنوع مختلف با استفاده از اجزای اصلی یکسان فشرده کرد. همه چیز به این بستگی دارد که چقدر از کل واریانس تصویر را می‌توانیم در اجزای اصلی خود در نظر بگیریم.

$$explained\ variance\ of\ PC_k = \frac{eigenvalue\ of\ PC_k}{\sum_{i=1}^p eigenvalue\ of\ PC_i}$$

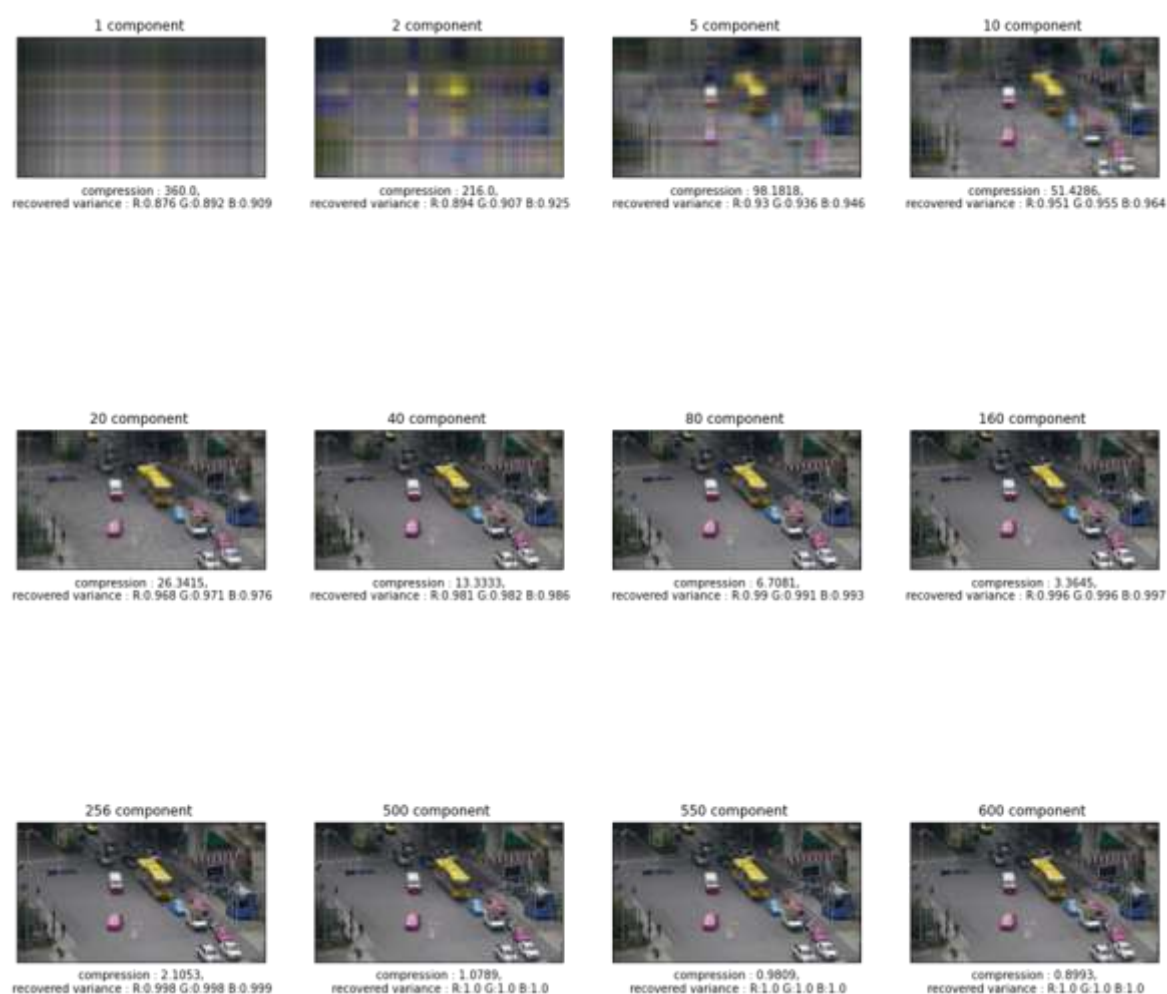
---

<sup>1</sup> - projection

بدیهی است که ۸۰ درصد واریانس جزئیات کافی برای تولید تصاویر با کیفیت نیست. با این حال، هرتصویری بعد از ۹۰٪ خوب به نظر می رسد و بعد از ۹۵٪ کیفیت واقعاً خوب است.

این دیدگاهی را در مورد میزان واریانسی که باید برای تولید یک تصویر با کیفیت خوب با فشرده سازی آنالیز مولفه اصلی بگیرید را به ارائه می دهد.

#### جدول ۷-۰ مقایسه فشرده سازی با کامپوننت های مختلف در روش pca



### ۳-۹ کدگذاری هافمن

در علوم کامپیوتر و تئوری اطلاعات، کدگذاری هافمن نوع مشخصی از کد پیشوندی<sup>۱</sup> بهینه است که کاربردی فراوان در فشرده سازی بی اتلاف اطلاعات دارد. فرایند پیدا کردن یا استفاده از این کد، با بهره گیری از الگوریتمی انجام می شود که توسط «دیوید هافمن» توسعه داده شده است و برای اولین بار در سال ۱۹۵۲ در مقاله ای با عنوان «روشی برای تولید کدی با کمترین تکرار زوائد» منتشر شد. کدهای پیشوندی نوعی از کدها (توالی بیت ها) هستند که در آن ها کد اختصاص داده

<sup>1</sup> Prefix code

شده به یک کاراکتر پیشوند کد تخصیص داده شده به هیچ کاراکتر دیگری نیست. این، روشی است که کدگذاری هافمن با استفاده از آن اطمینان حاصل می‌کند که هیچ ابهامی هنگام رمزگشایی توالی بیت‌های (جریان بیت) تولید شده وجود نخواهد داشت.

الگوریتم کدگذاری هافمن به شرح زیر است:

۱- چگالی هر کاراکتر را محاسبه می‌کنیم (تعداد دفعات حضور کاراکتر در متن مورد نظر).

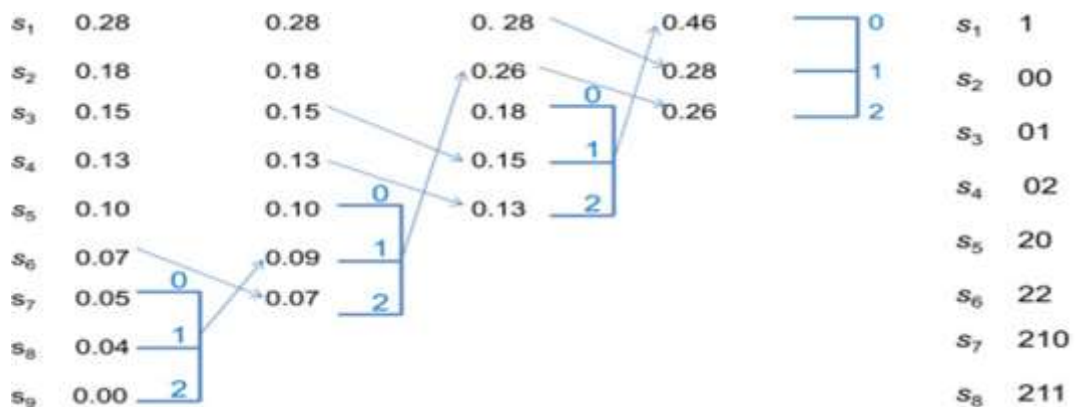
۲- دو کاراکتر با کمترین میزان تکرار (چگالی) را انتخاب می‌کنیم.

۳- کاراکترهای مرحله ۲ را با کاراکتر جدیدی که دارای چگالی برابر با مجموع چگالی دو کاراکتر فوق است جایگزین می‌کنیم.

۴- تا زمانی که فقط یک کاراکتر باقی مانده باشد، به مرحله ۲ می‌رویم.

۵- از عملیات فوق یک درخت حاصل می‌شود، بر روی این درخت هر مسیر به سمت چپ با ۰ و هر مسیر به سمت راست با ۱ وزن دهی می‌شود.

۶- کد هر کاراکتر با کنار هم گذاشتن وزن‌ها از ریشه تا آن کاراکتر به دست می‌آید.



شکل ۳-۱۱: مثالی از کدگذاری هافمن

در سیستم‌های فشرده‌سازی تصویر، کدگذاری هافمن بر روی نمادهای کوانتیزه انجام می‌شود. از آن جایی که یک تصویر یک ماتریس  $2 \times 2$  یا سه بعدی (اگر رنگی باشد) است، داده‌هایی که باید رمزگذاری کنیم، ابعاد تصویر و شدت هر پیکسل است. اگر  $I$  ردیف‌ها،  $C$  ستون‌ها و  $d$  کانال‌های برای تصویر وجود داشته باشد، سپس باید  $C * d * r$  را رمزگذاری کنیم.

اگر این کدگذاری را برای تصویررنگی زیر با ابعاد  $(1920 * 1080)$  پیاده‌سازی کنیم حجم لازم برای ذخیره‌سازی تصویر به  $43350$  بیت می‌رسد. این کاهش حجم ذخیره‌سازی به خاطر تولید کدهایی با طول متغیر برای نمایش هر شدت رنگ در تصویر می‌باشد بنابراین میانگین طول کدهای تولید شده با الگوریتم هافمن از میانگین طول کدهای تصویر اصلی (در تصویر اصلی هر شدت رنگ با کدی با طول یکسان نمایش داده شده است) کمتر خواهد بود. جدول زیر نمونه از تفاوت کدگذاری هافمن را با کدگذاری طول ثابت را نشان می‌دهد.

رمزگذاری هافمن	رمزگذاری طول ثابت ۸ بیتی	عدد
۰۰۰۰۰۰۰۰	۰۰۱۰۰۰۱۱	۳۵
۱۱۱۱۰	۱۰۰۰۰۱۰۰	۱۳۲
⋮	⋮	⋮
⋮	⋮	⋮
۰۱۱۰۱۱۱۰۰	۰۰۰۰۰۰۱۰	۲

### ۱۰-۳ کوانتیزاسیون برداری<sup>۱</sup>

کوانتیزاسیون برداری یک تکنیک موثر برای انجام فشرده سازی داده ها است. از نظر تئوری، کوانتیزاسیون برداری همیشه بهتر از کوانتیزه کردن اسکالر است، زیرا به طور کامل از همبستگی بین اجزای درون بردار بهره برداری می کند. عملکرد بهینه کدگذاری زمانی به دست می آید که بعد بردار به بی نهایت نزدیک شود و سپس از همبستگی بین همه اجزا برای فشرده سازی استفاده شود.

یکی دیگر از ویژگی های بسیار جذاب کوانتیزاسیون برداری تصویر این است که روش رمزگشایی آن بسیار ساده است زیرا فقط از جدول جستجو تشکیل شده است. با این حال، دو مشکل عمده در تکنیک های کوانتیزاسیون برداری تصویر وجود دارد. اولی این است که پیچیدگی کوانتیزاسیون برداری به طور تصاعدی با افزایش ابعاد افزایش می یابد بنابراین، برای کوانتیزاسیون برداری مهم است که مشکل چگونگی طراحی یک سیستم کدگذاری عملی را به نحوی حل کنیم که بتواند عملکرد معقولی را تحت پیچیدگی معین ارائه دهد.

دومین مشکل عمده کوانتیزاسیون برداری تصویر نیاز به کتابچه کد<sup>۲</sup> است که باعث ایجاد مشکلاتی در موارد عملی مانند تولید یک کتابچه کد جهانی برای تعداد زیادی تصویر، مقیاس بندی کتابچه کد برای مطابقت با نرخ بیت مورد نیاز و .. می شود.

یک کوانتایزر  $N$  سطحی،  $Q$ ، یک نگاشت از مجموعه بردار  $k$  بعدی  $\{v\}$  به یک کتابچه کد محدود  $w = \{w_1 w_2 w_n\}$  است.

$$Q: v \rightarrow w$$

به عبارت دیگر، یک بردار ورودی،  $v$ ، را به یک بردار نماینده (کلمه رمز)،  $w$  از کتابچه کد،  $W$ ، کوانتایزر برداری،  $Q$ ، به طور کامل توسط کتابچه کد، همراه با پارتیشن جدا،  $R = \{r_1. r_2. \dots r_n\}$  توضیح داده شده است.

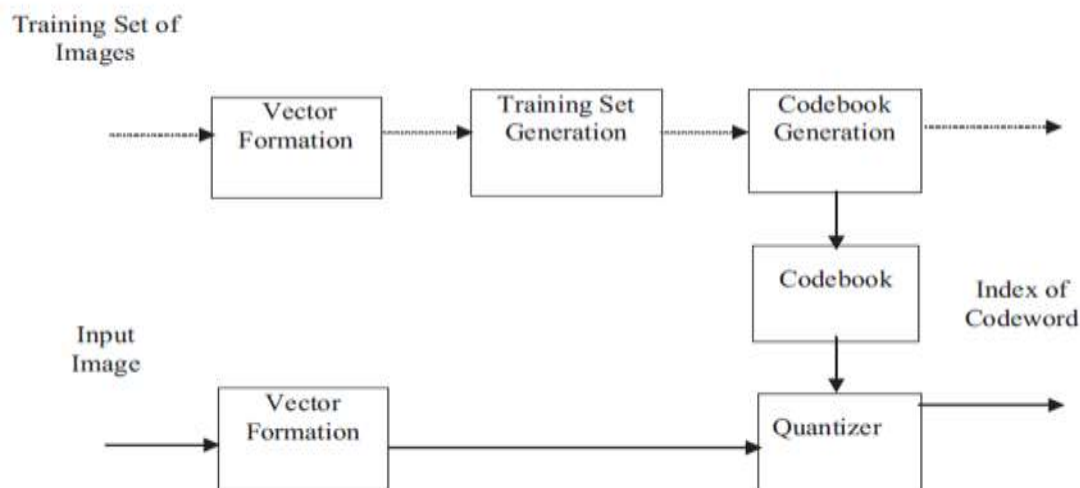
$$r_i = \{v: Q(v) = w_i\}$$

و  $w$  و  $v$  بردارهای  $K$  بعدی هستند. دیاگرام زیر مراحل مختلف کوانتیزاسیون برداری را نشان می دهد.

<sup>۱</sup> - vector quantization

<sup>۲</sup> - look-up

<sup>۳</sup> -code book



شکل ۱۱-۰ دیاگرام کوانتیزاسیون برداری

اولین مرحله از کوانتیزاسیون برداری تصویر، تشکیل<sup>۱</sup> تصویر است. داده های تصویر ابتدا به مجموعه ای از بردارها تقسیم می شوند. سپس تعداد زیادی بردار از تصاویر مختلف برای تشکیل یک مجموعه آموزشی استفاده می شود. مجموعه آموزشی برای تولید یک کتابچه کد، معمولاً با استفاده از یک الگوریتم خوشه بندی تکراری<sup>۲</sup> استفاده می شود. مرحله کمی سازی و کدگذاری شامل جستجو برای هر بردار ورودی، نزدیکترین کلمه رمز در کتابچه کد است. سپس شاخص متناظر کلمه رمز انتخاب شده کدگذاری شده و به رمزگشا ارسال می شود. در رمزگشا، ایندکس رمزگشایی شده و با همان دفترچه رمزی که در رمزگذار در جدول جستجو وجود دارد، به بردار مربوطه تبدیل می شود.

#### شکل گیری برداری

اولین مرحله کوانتیزاسیون برداری، تشکیل برداری است، یعنی تجزیه تصاویر به یک مجموعه از بردارها. اساساً رویکردهای تشکیل بردار را می توان به دو دسته تقسیم کرد: مستقیم زمانی یا مکانی و استخراج ویژگی. مستقیم مکانی یا زمانی یک رویکرد ساده برای تشکیل بردارها از مقادیر شدت یک بلوک به هم پیوسته مکانی یا زمانی از پیکسل ها در یک تصویر یا یک دنباله تصویر است. نوع دیگر روش استخراج ویژگی است. یک ویژگی تصویر یک ویژگی اولیه متمایز است. برخی از ویژگی ها طبیعی هستند به این معنا که چنین ویژگی هایی با ظاهر بصری یک تصویر تعریف می شوند، در حالی که سایر ویژگی های مصنوعی از دستکاری ها یا اندازه گیری های خاص تصاویر یا توالی های تصویر ناشی می شوند. در شکل گیری برداری، رایج است که داده های تصویر در حوزه فضایی را می توان به حوزه متفاوتی تبدیل کرد تا کوانتیزاسیون بعدی و رمزگذاری آنتروپی مشترک کارآمدتر باشد. برای این منظور می توان برخی از ویژگی های داده های تصویری مانند ضرایب تبدیل شده، میانگین بلوک ها را استخراج و بردار کوانتیزه کرد. اهمیت عملی استخراج ویژگی این است که می تواند منجر به کاهش اندازه برداری و در نتیجه کاهش پیچیدگی روند کدگذاری شود.

#### تولید مجموعه آموزشی

یک کوانتایزر برداری بهینه باید به طور ایده آل با آمار منبع بردار ورودی مطابقت داشته باشد. با این حال، اگر آمار یک منبع بردار ورودی ناشناخته باشد، می توان از مجموعه آموزشی نماینده منبع بردار ورودی مورد انتظار برای طراحی کوانتایزر برداری استفاده کرد. اگر منبع بردار مورد انتظار دارای واریانس زیادی باشد، به مجموعه آموزشی بزرگی نیاز است. برای کاهش پیچیدگی پیاده سازی ناشی از یک مجموعه آموزشی بزرگ، منبع بردار ورودی را می توان به زیر مجموعه ها تقسیم کرد. کتابچه کدهای جداگانه به دست آمده از هر زیرمجموعه بهم وصل می گردد و به یک کتابچه کد نهایی بدست

<sup>۱</sup> -formation

<sup>۲</sup> - iterative clustering algorithm



می آید. در روش‌های دیگر، منابع ورودی محلی کوچک مربوط به بخش‌هایی از تصویر به‌عنوان مجموعه‌های آموزشی استفاده می‌شوند، بنابراین کتابچه کد بهتر می‌تواند با آمارهای محلی مطابقت داشته باشد. با این حال کتابچه کد برای ردیابی تغییرات آمار محلی منابع ورودی باید به روز شود. که این ممکن است پیچیدگی را افزایش داده و کارایی کدگذاری را کاهش دهد. عملاً در اکثر سیستم‌های کدگذاری مجموعه‌ای از تصاویر معمولی به عنوان مجموعه آموزشی انتخاب شده و برای تولید کتابچه کد استفاده می‌شود.

### تولید کتابچه کد

مرحله کلیدی کوانتیزاسیون برداری، توسعه یک کتابچه کد خوب است. کتابچه کد بهینه، با استفاده از معیار میانگین مربعات خطا<sup>۱</sup> باید دو شرط لازم را برآورده کند. ابتدا منبع بردار ورودی به تعدادی از مناطق از پیش تعیین شده با قانون حداقل فاصله تقسیم می‌شود. تعداد مناطق بر اساس نیاز نرخ بیت یا نسبت فشرده سازی و عملکرد کدگذاری تعیین می‌شود. دوم، کلمه رمز برای بردار نماینده این منطقه، مقدار میانگین یا مرکز آماری (میان) بردارهای داخل منطقه است. تحت این دو شرط، یک الگوریتم خوشه بندی ال بی جی<sup>۲</sup> تعمیم یافته به طور گسترده برای تولید کتابچه کد استفاده شده است. الگوریتم خوشه بندی یک فرآیند تکراری است که شاخص عملکرد محاسبه شده از فاصله بین بردارهای نمونه و مراکز خوشه آنها را به حداقل می‌رساند. الگوریتم خوشه بندی ال بی جی فقط می‌تواند یک کتابچه کد با یک بهینه محلی تولید کند که به دانه های خوشه اولیه بستگی دارد. دو روش اساسی برای به دست آوردن کتابچه کد اولیه یا دانه های خوشه ای استفاده شده است. در رویکرد اول، نقطه شروع شامل یافتن یک کتابچه کد کوچک با تنها دو کلمه رمز و سپس تقسیم مجدد کتاب رمز تا زمانی که تعداد مورد نیاز کلمه رمز به دست آید، می‌باشد. به این روش تقسیم باینری می‌گویند. دومی با دانه های اولیه برای تعداد مورد نیاز کلمه رمز شروع می‌شود، این دانه ها با پردازش اولیه مجموعه های آموزشی تولید می‌شوند. برای پرداختن به مشکل بهینه محلی، (Equitz (1989 یک الگوریتم خوشه بندی جدید، الگوریتم نزدیکترین همسایه زوجی<sup>۳</sup> را پیشنهاد کرد. الگوریتم نزدیکترین همسایه زوجی با یک خوشه جداگانه برای هر بردار در مجموعه آموزشی شروع می‌شود و دو خوشه را در یک زمان با هم ادغام می‌کند تا اندازه کتابچه کد مورد نظر به دست آید. در ابتدای فرآیند خوشه بندی، هر خوشه فقط شامل یک بردار است. در فرآیند زیر، دو بردار نزدیک در مجموعه آموزشی با مقدار میانگین آماری خود ادغام می‌شوند، به گونه ای که خطای ایجاد شده با جایگزینی این دو بردار با یک کلمه رمز به حداقل می‌رسد. الگوریتم، نزدیکترین همسایه زوجی به طور قابل توجهی پیچیدگی محاسبات را بدون به خطر انداختن عملکرد کاهش می‌دهد. این الگوریتم همچنین می‌تواند به عنوان تولید کتابچه کد اولیه برای الگوریتم ال.بی.جی استفاده شود.

### کوانتیزاسیون

کوانتیزاسیون در زمینه یک کوانتیزاسیون برداری شامل انتخاب یک کلمه رمز در کتابچه کد برای هر بردار ورودی است. کوانتیزه کردن فول سرچ<sup>۴</sup> فرآیند جستجوی جامع در کل کتابچه کد برای یافتن نزدیکترین کلمه رمز است. با این روش جستجو حجم محاسبات زیاد است. یک رویکرد جایگزین، کوانتیزاسیون درختی<sup>۵</sup> است، جایی که جستجو بر اساس یک پارتیشن سلسله مراتبی انجام می‌شود. جستجوی درخت بسیار سریعتر از جستجوی کامل است، اما واضح است که جستجوی درختی برای کتابچه کد داده شده کمتر از حد بهینه است و به حافظه بیشتری برای کتابچه کد نیاز دارد.

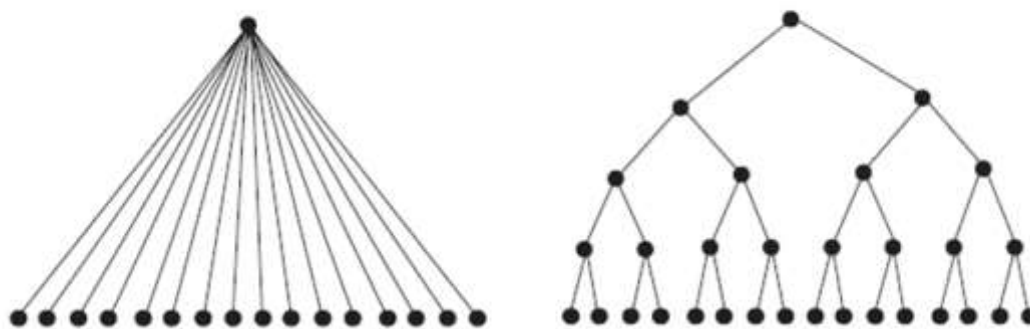
<sup>1</sup> - MSE

<sup>2</sup> - LBG algorithm(proposed by Linde, Buzo, and Gray in 1980)

<sup>3</sup> - pairwise nearest neighbor (PNN)

<sup>4</sup> - full-search

<sup>5</sup> - tree-search



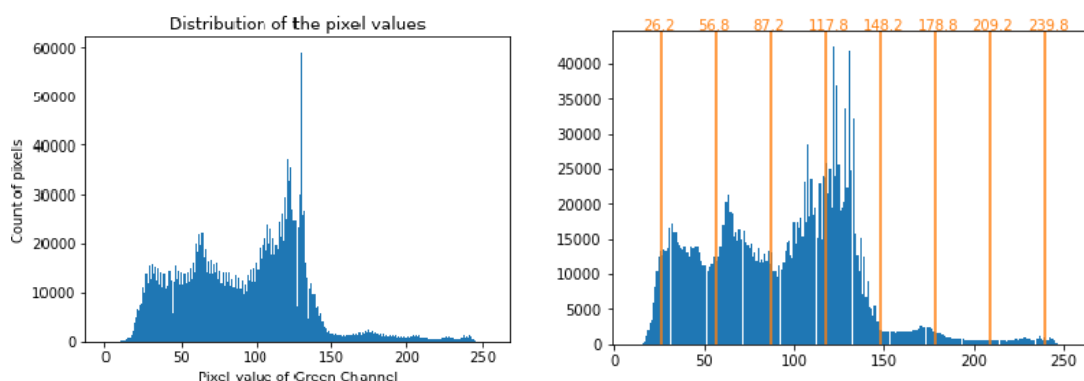
شکل ۱۲-۰ مفایسه جستجوی باینری و جستجو جامع

### فرآیند عملی انجام کوتنتزاسیون برداری:

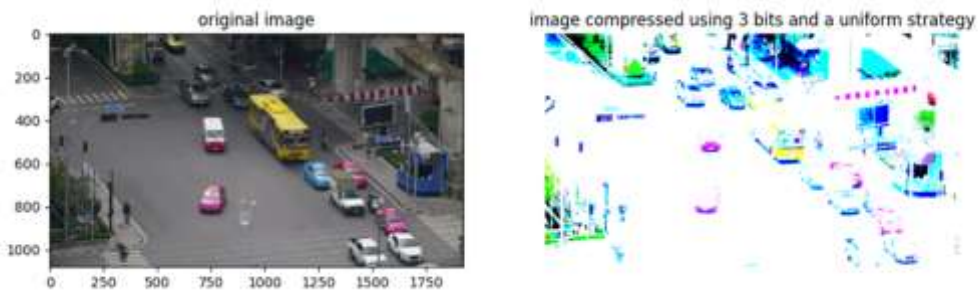
با کاهش تعداد سطوح خاکستری برای نمایش یک تصویر فشرده سازی از طریق کوانتیزاسیون برداری را انجام می دهیم. به عنوان مثال، ما می توان به جای ۲۵۶ مقدار از ۸ مقدار برای نمایش شدت هر پیکسل استفاده کرد. بنابراین، به این معنی است که ما می توان به طور موثر از ۳ بیت به جای ۸ بیت برای رمزگذاری یک پیکسل استفاده کرد و بنابراین مصرف حافظه را تقریباً ۲,۵ کاهش دهیم. برای فشرده سازی تصاویر رنگی با این روش ابتدا تصویر را با کانال های آبی، سبز و قرمز تقسیم می کنیم و عملیات فشرده سازی برای هر کانال به طور جداگانه انجام می شود سپس نتیجه حاصل باهم ادغام گشته و تصویر نهایی فشرده شده را به وجود می آورد. برای تعریف ۸ مقدار خاکستری برای نمونه باید یک استراتژی انتخاب نمود. ساده ترین استراتژی این است که آنها را با فواصل مساوی تعریف کرد. نتیجه این استراتژی را می توان در هیستوگرام زیر مشاهده نمود.

نتیجه اعمال استراتژی فواصل مساوی برای کوانتیزه کردن تصویر در شکل قابل مشاهده است.

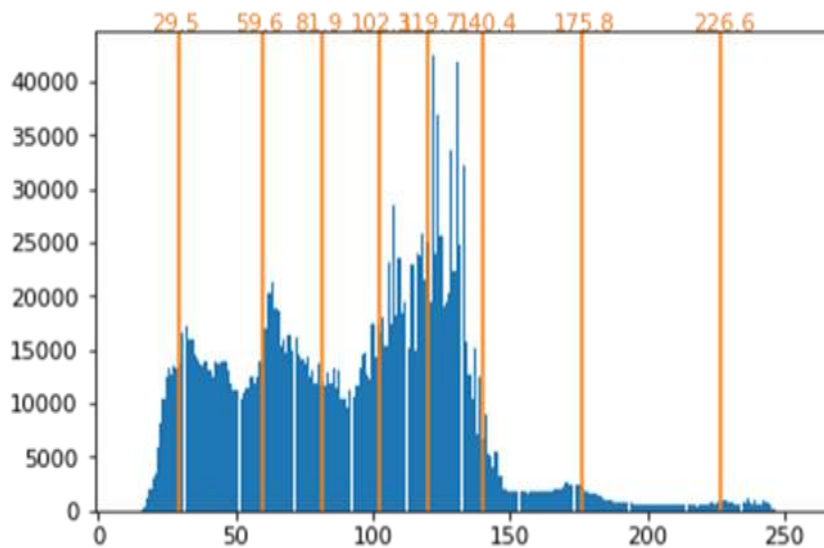
### جدول ۸-۰ تفسیم بندی یکسان شدت پیکسل ها



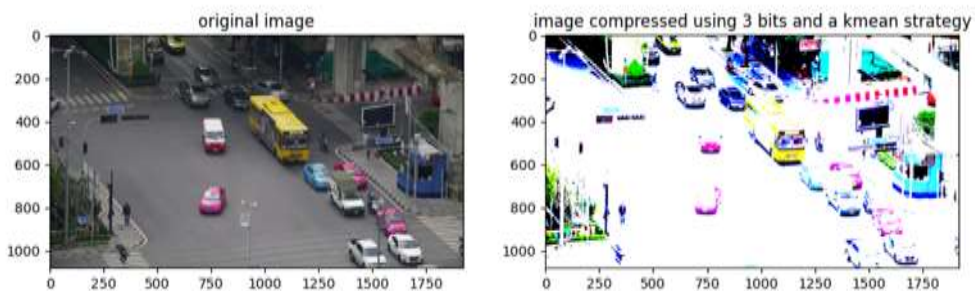
## جدول ۹-۰ فشرده سازی با روش vq



اما استراتژی نمونه گیری یکنواخت بهینه نیست. زیرا همان طور که در مثال بالا مشخص است پراکندگی شدت پیکسل ها در همه مقادیر یکسان نیست. برای بهبود بخشیدن به این فرآیند از یک الگوریتم خوشه بندی مثل استراتژی خوشه بندی مانند k-means برای یافتن یک نقشه بهینه تر استفاده کرد. نتیجه اعمال این الگوریتم را در هیستوگرام زیر می توان مشاهده نمود.



شکل ۱۳-۰ پراکندگی با الگوریتم kmean



## جدول ۱۰-۰ فشرده سازی با روش vq و kmean

## ۱۱-۳ روش ترکیبی تبدیل فوریه و تحلیل مولفه های اساسی

طرح ایده

در مطالعات قبلی به نقش روش تحلیل مولفه های اساسی برای فشرده سازی تصویر پرداخته شد اما نتایج حاصل از آن بهیچیک مورد نظر را به وجود نیاورد بنابراین این ایده در این قسمت مورد بررسی قرار میگیرد که اگر روش تجزیه و تحلیل مولفه های اساسی بر روی دامنه فرکانسی تصویر اعمال گردد آیا می توان نتیجه گرفت که تجزیه و تحلیل مولفه های اساسی توانسته فرکانس های مهم و اصلی تصویر را استخراج کند و آیا می توان از این فرکانس های اساسی استخراج شده توسط این روش جهت فشرده سازی تصویر استفاده نمود یا خیر.

روش کار

روش انجام کار به این صورت خواهد بود که داده ها از حوزه زمانی/مکانی به حوزه فرکانس با استفاده از تکنیک هایی مانند تبدیل فوریه، منتقل می شوند و سپس اعمال تجزیه و تحلیل مولفه های اساسی به نمایش دامنه فرکانس حاصل از داده ها اتفاق می افتد. در نهایت ضرایب فرکانس حاصل را می توان بر اساس بزرگی آنها مرتب کرد. تنها با حفظ مهم ترین ضرایب و کنار گذاشتن بقیه، می توانیم به فشرده سازی با حداقل کاهش کیفیت تصویر دست یابیم. ضرایب حفظ شده را می توان برای کاهش مقدار داده های مورد نیاز برای نمایش تصویر، کمی بیشتر کرد.

هنگامی که تحلیل مولفه های اساسی در حوزه فرکانس انجام میگردد اولین مولفه های اساسی بیانگر این هستند که کدام فرکانس ها بیشترین تأثیر را در واریانس مجموعه داده ها دارند.

تفاوت بین تجزیه و تحلیل مولفه های اساسی در حوزه زمان و دامنه فرکانس را می توان از نظر نحوه محاسبه مقادیر ویژه درک کرد. در حوزه زمانی از ماتریس همبستگی استفاده می شود. در حوزه فرکانس، تبدیل فوریه ماتریس همبستگی یا ماتریس چگالی طیفی برای محاسبه مقادیر ویژه استفاده می شود.

اعمال تجزیه مولفه های اساسی بر ماتریس فرکانس یک رویه سیستماتیک برای استخراج ویژگی های مهم در مجموعه داده های تصویر در اختیار ما قرار می دهد. علاوه بر این، این روش به آسانی با سیگنال های غیر ثابت، یعنی سیگنال هایی که ویژگی های طیفی آنها در طول زمان تکامل می یابد، سازگار است.

### جدول ۱۱-۰ مقایسه فشرده سازی تبدیل فوریه و تبدیل فوریه در حوزه فرکانس



فشرده سازی با روش تبدیل فوریه در حوزه فرکانس

41.9534:MSE  
31.9031:PNSR



فشرده سازی با تبدیل فوریه

51.8613:MSE  
30.9823:PNSR

مشاهده میشود که در درصد ثابت فشرده سازی (از لحاظ حجم فایل تصویری) روش پیشنهادی میانگین مربعات کمتری نسبت به روش فشرده سازی با تبدیل فوریه دارد که این یک مزیت محسوب می شود اما از لحاظ کیفیت بصری انسان این روش نسبت به روش تبدیل فوریه افت دارد .



تجزیه و تحلیل مولفه های اساسی در حوزه فرکانس  
41.9534:MSE  
31.9031:PNSR



تجزیه و تحلیل مولفه های اساسی در حوزه مکان  
:MSE  
22.3026:PNSR

از طرفی به علت دو مرحله بودن این روش و نیاز به محاسبات تبدیل فوریه و تحلیل مولفه های اساسی در این کار زمان و هزینه پیاده سازی افزایش یافته که این نکته در کاربرد کنترل ترافیک اهمیت ویژه ای دارد. با توجه به نتایج حاصل شده بنظر می رسد این روش اگرچه برای استخراج ویژگی تصویر روش کاربردی است اما در هدف فشرده سازی تصویر بهینه عمل نمی کند.

## فصل ۴

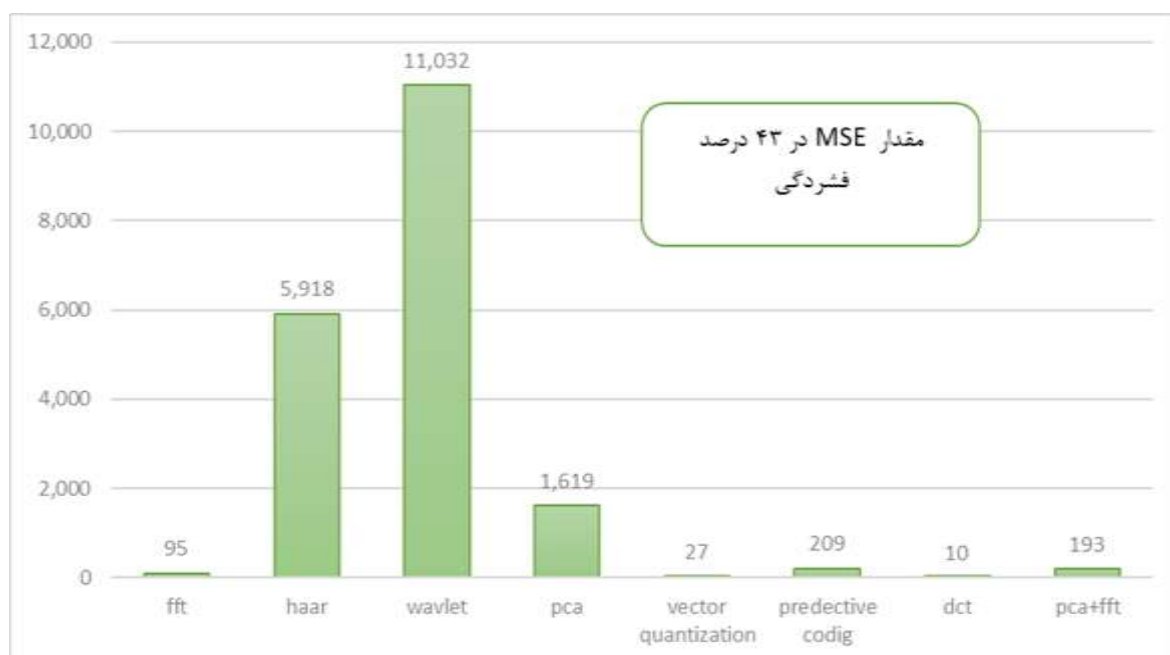
### نتیجه گیری و پیشنهادات

#### 4-1 مقایسه و نتیجه گیری

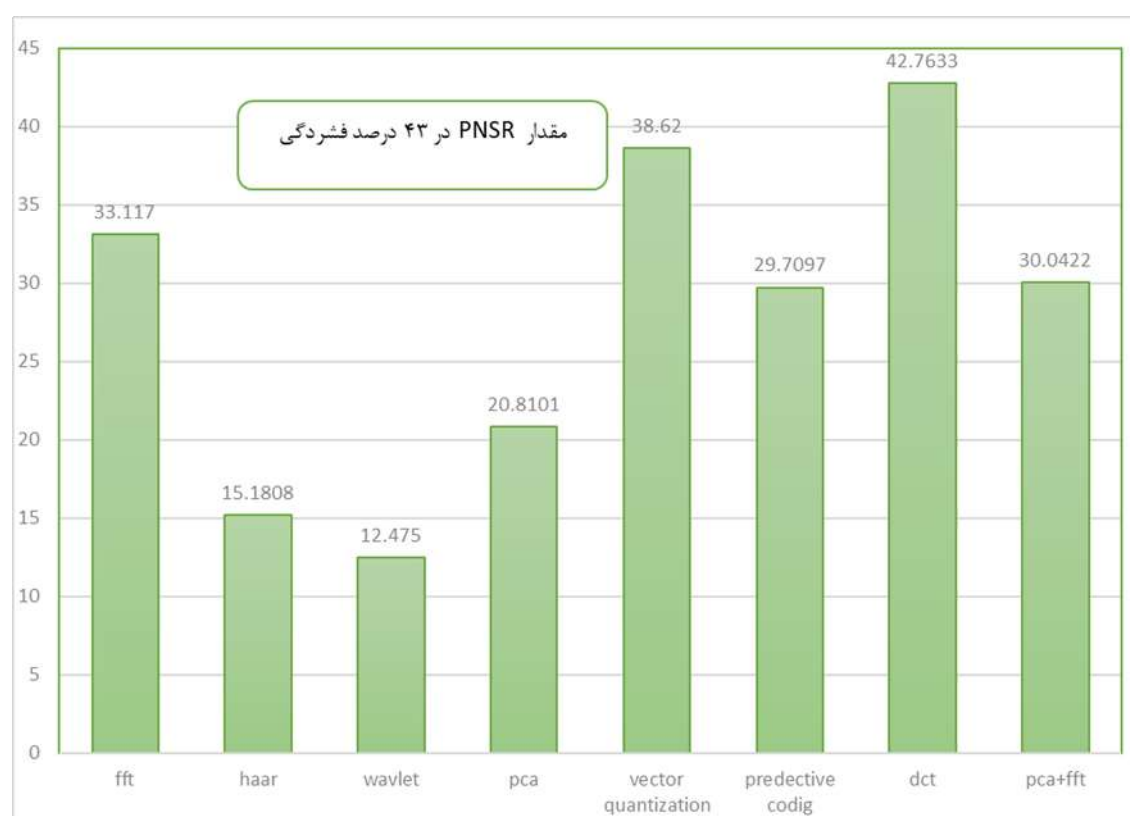
در فصل قبل به طور مفصل درمورد الگوریتم یا تکنیک‌های اصلی فشرده‌سازی تصویر و استانداردهای فعلی که توسط آن تکنیک استفاده می‌شود توضیح داده شد. برای انتخاب بهترین تکنیک فشرده سازی از آنجایی که دو نوع تکنیک فشرده‌سازی وجود دارد که به طور گسترده مورد استفاده قرار می‌گیرند، تشخیص اینکه کدام تکنیک فشرده‌سازی باید استفاده شود دشوار است. کاربردهای حیاتی داده مانند تصویربرداری پزشکی که از دست دادن هیچ داده ای قابل مورد قبول نیست ، می توان از تکنیک های فشرده سازی بدون اتلاف استفاده کرد اما در مواردی که از دست دادن اندک داده هایی که محسوس و قابل ملاحظه نیستند، تکنیک فشرده سازی با اتلاف مورد قبول است. در تکنیک های فشرده سازی با تلفات، تبدیل‌هایی مانند تبدیل کسینوس گسسته و تبدیل موجک برای تغییر پیکسل‌های تصویر اصلی به ضرایب حوزه فرکانس استفاده می‌شوند. این ضرایب دارای چندین ویژگی مطلوب هستند، مانند خاصیت تراکم انرژی که منجر به تمرکز بیشتر انرژی داده های اصلی تنها در تعداد کمی از ضرایب تبدیل قابل توجه می شود. ضرایبی که مقادیر کوچکی دارند کنار گذاشته می شوند و ضرایب باقی مانده با مقادیر قابل توجه انتخاب می شوند. ضرایب انتخاب شده برای کمی سازی و رمزگذاری بیشتر در نظر گرفته می شوند.

برای مقایسه الگوریتم های فشرده سازی تصویر در هر الگوریتم تصویر را تا ۴۳ درصد حجم اصلی خود فشرده نموده ایم و معیار های میانگین مجموع مربعات خطا و پیک نویز به سیگنال و مدت زمان اجرا کد الگوریتم فشرده سازی با زبان برنامه نویسی پایتون را برای هر کدام از الگوریتم ها در این حالت اندازه گیری نموده ایم. نتایج حاصله به شکل زیر هستند.

روش فشرده سازی	MSE	PSNR (db)	Execution time(second)
تبدیل فوریه گسسته	95.1687	33.1170	1.2273
تبدیل موجک	11031.7031	12.4750	0.9574
تبدیل هار	5918.2443	15.1808	0.9938
تبدیل کسینوسی	10.3246	42.7633	4.4434
تحلیل مولفه اساسی	1618.7633	20.8101	8.1389
کوانتیزاسیون برداری	26.7525	38.6200	120
رمزگذاری پیش گوینه بدون اتلاف	208.5564	29.7097	0.7038
تحلیل مولفه اساسی در حوزه فرکانس	193.1860	30.0422	13.2589

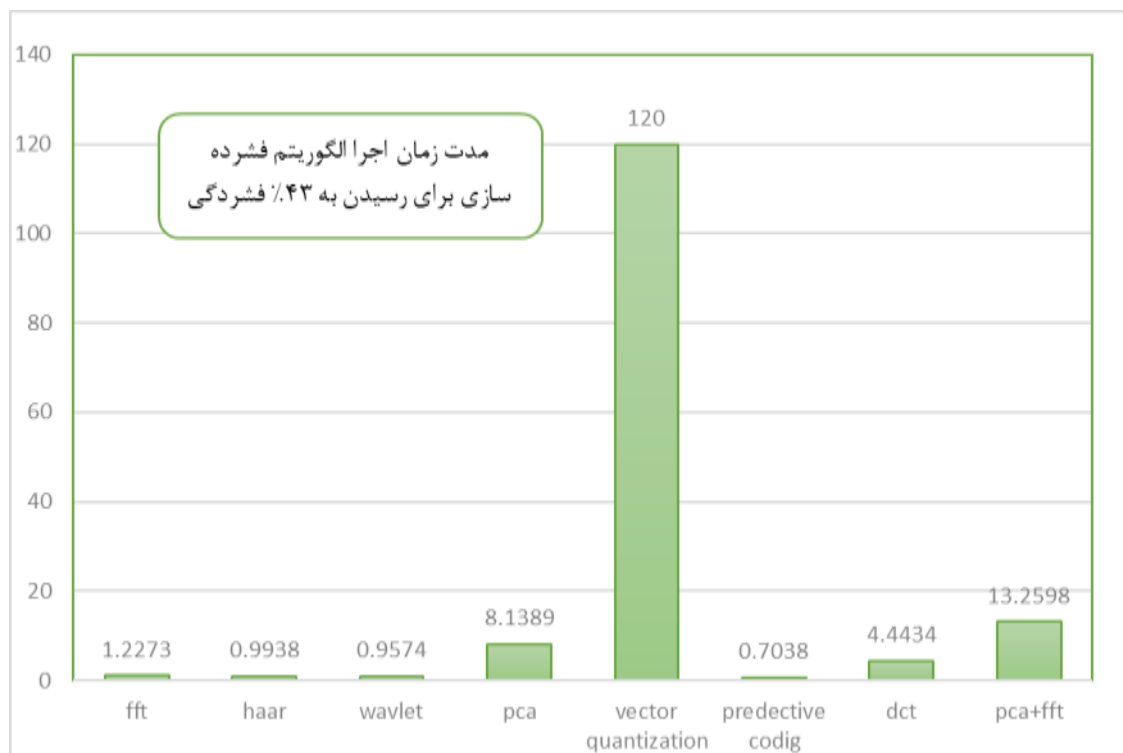


شکل ۱۴-۰ مقایسه میانگین مربعات



شکل ۱۵-۰ مقایسه پیک سیگنال به نویز





شکل ۱۶-۰ مقایسه مدت زمان اجرا

با توجه به نتایج فوق روش تبدیل کوسینوسی گسسته توانسته است علاوه بر حفظ کیفیت تصویر در زمان قابل قبولی اجرا گردد.

نکته قابل ذکر در این جا این می باشد که این الگوریتم ها همه با زبان پایتون پیاده سازی شده اند که یک زبان برنامه نویسی سطح بالا محسوب می گردد و برای کاربرد های پیاده سازی الگوریتم های پایه مناسب نمی باشد پس زمان اجرای همه ی این الگوریتم ها پس از پیاده سازی با یک زبان سطح پایین و نزدیک به حافظه مسلما کمتر خواهد بود اما این نسبت ها فارغ از زبان برنامه نویسی که برای پیاده سازی الگوریتم های فشرده سازی انتخاب شده همچنان باقی خواهد بود.

از آنجا که در این مطالعه، فشرده سازی تصویر در کاربرد سیستم کنترل ترافیک مورد بحث است در انتخاب روش مناسب برای فشرده سازی این موضوع اهمیت دارد.

در سیستم های کنترل ترافیک امروزی هم انسان (شخص اپراتوری که تصاویر دوربین ها را مشاهده و در اتاق فرمان کنترل ترافیک را به عهده دارد) هم رایانه برای هدف کنترل ترافیک نقش دارند پس این امر اهمیت دارد که تصویر فشرده شده هم برای چشم انسان کیفیت کافی را داشته باشد و هم کیفیت کافی را برای انجام پردازش های تصویر توسط رایانه را داشته باشد.

در زیر به معرفی معیاری که به نوعی بیان کننده کیفیت بصری تصویر برای چشم انسان است می پردازیم.

#### میانگین امتیاز نظرات<sup>۱</sup>

میانگین امتیاز نظر معیاری است که در حوزه کیفیت تجربه و مهندسی مخابرات استفاده میشود و کیفیت کلی یک محرک یا سیستم را نشان میدهد. به طور کلی، میانگین امتیاز عقیده را می توان در هر جایی که تجربه و نظر ذهنی انسان مفید

<sup>1</sup> -Mean opinion score

باشد، به کار گرفت. در عمل، اغلب برای قضاوت در مورد تقریب های دیجیتالی پدیده های جهان استفاده می شود. دامنه هایی که معمولاً در آنها میانگین امتیاز نظر اعمال میشود شامل فشرده سازی تصویر ثابت (مانند jpg) کدک های صوتی و ویدیویی است. راجح ترین مقیاس رتبه بندی، مقیاس رتبه بندی مطلق (ACR) است که از ۱ تا ۵ متغیر است. سطوح رتبه بندی دسته بندی مطلق عبارتند از:

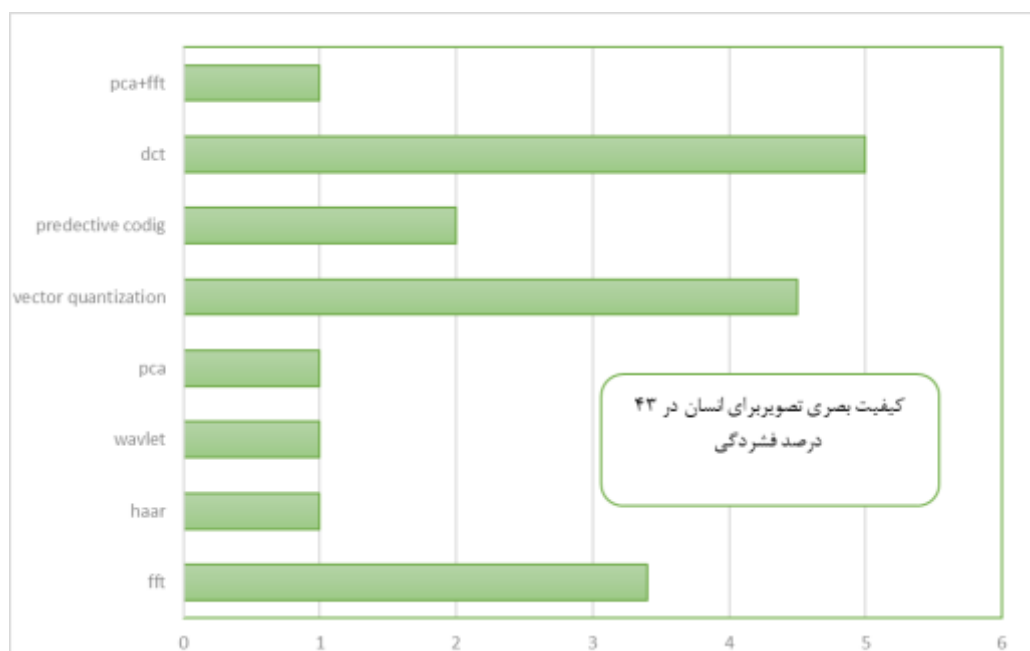
- 5 عالی
- 4 خوب
- 3 منصفانه
- 2 ضعیف
- 1 بد

میانگین امتیاز نظرات با ارزیابی کیفیت تصاویر فشرده شده با استفاده از الگوریتم های مختلف و ارائه امتیاز توسط گروهی از افراد به دست می آید. امتیازها برای ارائه میانگین امتیازی است که نشان دهنده کیفیت کلی تصویر فشرده شده است، همانطور که توسط گروه افراد درک می شود.

برای استفاده از میانگین امتیاز نظرات برای فشرده سازی تصویر، گروهی از افراد باید انتخاب شوند تا کیفیت تصاویر فشرده شده با استفاده از الگوریتم های مختلف را ارزیابی کنند. افراد باید در زمینه پردازش تصویر و درک بصری تجربه و تخصص داشته باشند. کیفیت تصاویر فشرده شده را می توان با مقایسه تصویر فشرده شده با تصویر اصلی ارزیابی کرد. سپس امتیازهای اختصاص داده شده توسط هر فرد را می توان جمع آوری و میانگین گرفت تا میانگین امتیاز نظر را بدست آورد که نشان دهنده کیفیت کلی تصویر فشرده شده است.

میانگین امتیاز نظرات می تواند به عنوان یک معیار برای مقایسه الگوریتم های فشرده سازی مختلف و تعیین اثربخشی الگوریتم ها در حفظ کیفیت تصویر در طول فشرده سازی استفاده شود.

نتایج حاصل از تحقیق زیر را با استفاده از معیار میانگین امتیاز نظرات برای ۲۰ نفر مورد بررسی قرار دادیم و نتیجه به شکل زیر است.



شکل ۱۷-۰ مقایسه میانگین امتیاز نظرات

برای ارزیابی کیفیت الگوریتم های فشرده سازی در کاربرد پردازش های تصویر توسط رایانه به بررسی پردازش های تصویر رایج در حوزه کنترل ترافیک پرداختیم .

یکی از پردازش های انجام شده در سیستم کنترل ترافیک تشخیص وسایل نقلیه در جاده است که این تشخیص برای تخمین حجم ترافیک و شمارش خودرو ها و هم چنین تخمین سرعت خودرو ها انجام می شود . با استفاده از زبان برنامه نویسی پایتون یک الگوریتم تشخیص وسایل نقلیه پیاده سازی کردیم . ابتدا تصویر فشرده نشده را با این الگوریتم آزمایش کردیم و در این حالت مشاهده شد که الگوریتم ۴ وسیله نقلیه را در تصویر تشخیص می دهد. سپس تصاویر فشرده شده با الگوریتم های مورد بحث را که هر کدام تا ۴۳ درصد حجم اصلی تصویر فشرده شده اند را با این الگوریتم تست نمودیم نتیجه به شکل زیر است.



شکل ۱۸-۰ مقایسه تعداد وسایل نقلیه تشخیص داده شده

در نتیجه با یک مصالحه بین تمام معیار های ارائه شده می توان نتیجه گرفت روش تبدیل کسینوسی گسسته بهینه ترین روش برای فشرده سازی در کاربرد کنترل ترافیک می باشد زیرا که هم توانسته در حین حفظ کیفیت بصری برای چشم انسان و حفظ کیفیت جهت اعمال پردازش های تصویر رایانه ای در زمان نسبتا خوبی انجام گردد.

جدول ۱۲ مقایسه روش های فشرده سازی در ۴۳ درصد فشردگی



تبدیل فوریه گسسته



تب کسینوسی گسسته



تبدیل موجک هار



تبدیل موجک



کدگذاری پیشگویانه بدون اتلاف



کوانتیزاسیون برداری



تحلیل مولفه های اساسی در حوزه فرکانس



تحلیل مولفه های اساسی

## فهرست مراجع

- gent, K. c. (n.d.). Image compression and Discrete Cosine Transform. *collage of the Redwoods*.
- huffman, k. (2021). *Image Compression using Principal Component Analysis*. sect.
- Patterson, D. A. (2006). The Haar Wavelet Transform: Compression and Reconstruction.
- Patterson, D. A. (2006). The Haar Wavelet Transform: Compression and Reconstruction.
- Song, M.-S. (2008). Wavelet Image Compression.
- The Haar Wavelet Transform: Compression and. (n.d.).
- The Haar Wavelet Transform: Compression and . (n.d.).
- کلهر, گ. ب. (۱۳۹۲). بینایی ماشین و پردازش تصویر با *Open CV*. دانشگاهی کیان.
- گونزالس, ر. (۱۳۹۱). پردازش تصویر دیجیتال با زبان متلب. علوم رایانه.