

کاهش حجم ذخیره‌سازی تصاویر با استفاده از تحلیل مؤلفه‌های اصلی دوبعدی تُنک

زهرا یک کلام^۱ و فرح ترکمنی‌آذر^۲

چکیده

تحلیل مؤلفه‌های اصلی یکی از روش‌هایی است که در پردازش اطلاعات و کاهش ابعاد مجموعه داده موفق عمل کرده است؛ اما در زمان اعمال این الگوریتم به تصاویر، بایستی آن‌ها را به شکل یک بردار درآورد که سبب از بین رفتن همبستگی مکانی پیکسل‌های مجاور می‌گردد. برای حل این مشکل، تحلیل مؤلفه‌های اصلی دوبعدی مطرح شده که فرآیند تبدیل تصویر به بردار را نیاز ندارد. نکته‌ی دیگر، تُنک نبودن بردارهای پایه‌ی تحلیل مؤلفه‌های اصلی و هم ارزش نبودن تمامی آنان است. اخیراً تحلیل مؤلفه‌های اصلی تُنک مطرح گردیده که با حفظ خواص تحلیل مؤلفه‌های اصلی استاندارد، سعی می‌کند تعداد زیادی از درایه‌های بردارهای پایه را صفر کند. در این مقاله تحلیل مؤلفه‌های اصلی دوبعدی تُنک جهت بهره‌گیری همزمان از دو الگوریتم فوق بررسی خواهد شد. الگوریتم *Least Angle Regression- Elastic Net* با استفاده از قید نُرم یک و نُرم دو، محاسبه‌ی مؤلفه‌های اصلی یک‌بعدی تُنک را محقق می‌سازد. در این مقاله با اندک تغییراتی در ورودی الگوریتم مذکور، حالت دوبعدی آن را تحقق می‌بخشیم. عملکرد تحلیل مؤلفه‌های اصلی دوبعدی تُنک جهت فشرده‌سازی یک تصویر که به بلوک‌های 8×8 تقسیم گردیده، ارزیابی شده و در مقایسه با عملکرد تحلیل مؤلفه‌های اصلی دوبعدی، نتایج مناسبی بدست آمده است. همچنین با استفاده از ماتریس کوواریانس بلوک‌های 8×8 از ۶۰ تصویر متفاوت، مؤلفه‌های اصلی دوبعدی تُنک به‌گونه‌ای محاسبه گردیده که امکان استفاده از آن‌ها برای هر تصویر آزمون دیگری میسر می‌باشد.

کلیدواژه‌ها

کاهش ابعاد تصاویر، تحلیل مؤلفه‌های اصلی دوبعدی، تحلیل مؤلفه‌های اصلی تُنک، تحلیل مؤلفه‌های اصلی دوبعدی تُنک.

۱ - مقدمه

مباحث آماری کاربردهای زیادی دارند؛ در زمینه‌های پردازش سیگنال و تصویر نیز به‌طور گسترده مورد استفاده قرار گرفته‌اند. در حقیقت یکی از اصلی‌ترین حوزه‌هایی که از پیشرفت‌های اخیر نمایش تُنک داده‌ها سود می‌برد؛ پردازش تصویر است [۱-۲]. تحلیل مؤلفه‌های اصلی نیز از این قضیه مستثنی نبوده و محاسبات تُنک به آن راه‌یافته‌اند. تحلیل مؤلفه‌های اصلی تُنک^۱ برای بیش از یک دهه، موضوع فعال پژوهشی بوده است. در این مقاله هدف ما بررسی نسخه‌ی دوبعدی تحلیل مؤلفه‌های اصلی تُنک است. برای رسیدن به این موضوع، بررسی خود را از تحلیل مؤلفه‌های اصلی استاندارد آغاز می‌نماییم. از این پس،

یکی از روش‌های ذخیره‌سازی داده‌های با ابعاد بزرگ، استفاده از روش‌های تجزیه به مؤلفه‌های اصلی یا *PCA* است. در مورد مجموعه تصاویر با توجه به ابعاد بالای آنان همواره حجم محاسباتی مشکل‌ساز بوده است. اخیراً مسائل تقریب تُنک با اقبال زیادی مواجه شده‌اند و همان‌طور که در

این مقاله در خردادماه ۱۳۹۵ دریافت، در آبانماه بازنگری و در آذرماه همان سال پذیرفته شد.

^۱ دانشجوی کارشناسی ارشد رشته‌ی مهندسی برق گرایش مخابرات، دانشگاه شهید بهشتی

رایانامه: zahra.yekkalam@yahoo.com

^۲ دانشکده مهندسی کامپیوتر، دانشگاه شهید بهشتی

رایانامه: pourmahyabadi@uk.ac.ir

^۱Sparse Principal Component Analysis (SPCA)

یک تصویر، ابتدا باید آن را به یک بردار تبدیل نمود. این کار با قرار دادن سطرها و یا ستون‌های متوالی ماتریس تصویر به دنبال یکدیگر انجام می‌گیرد که در وهله‌ی اول موجب از بین رفتن همبستگی مکانی پیکسل‌ها نسبت به یکدیگر می‌شود؛ در واقع همبستگی بین پیکسل‌های نزدیک به هم از بین می‌رود. علاوه بر این اگر قرار باشد تحلیل مؤلفه‌های اصلی به مجموعه‌ای از تصاویر، مثلاً M تصویر با ابعاد $r \times c$ اعمال گردد؛ هرکدام از آن‌ها به صورت یک بردار با ابعاد $(r \times c) \times I$ درآمده و در یک ستون ماتریس داده که با \mathbf{X} نشان می‌دهیم، قرار می‌گیرد. به این ترتیب ابعاد ماتریس مذکور $(r \times c) \times M$ خواهد شد که منجر به ماتریس کواریانس با ابعاد $(r \times c) \times (r \times c)$ می‌گردد. به عنوان مثال برای تصاویر دارای 256 سطر و ستون، ابعاد ماتریس کواریانس 65536×65536 خواهد شد که در بسیاری از موارد، تعداد مشاهدات M از تعداد متغیرها $(r \times c)$ کمتر خواهد شد. با این اوصاف، مراحل بعدی الگوریتم که شامل استخراج بردارهای ویژه‌ی ماتریس کواریانس است؛ از لحاظ محاسباتی پیچیده و زمان‌بر خواهد شد. علاوه بر این، ابعاد بردارهای ویژه نیز بسیار بزرگ می‌شود [۵-۶].

زمینه‌ی حل این مشکل از دو سو فراهم آمده است. بعضی از محققین به دنبال یافتن مؤلفه‌های اصلی دوبعدی برآمده‌اند. در حالی که بعضی دیگر از محققین به دنبال یافتن مؤلفه‌های اصلی تنگ می‌باشند تا ابعاد و پیچیدگی محاسبات را کاهش دهند.

تحلیل مؤلفه‌های اصلی دوبعدی نخستین بار توسط یانگ بیان شد [۷-۸]. این پژوهش مبتنی بر استفاده از ماتریس تصاویر بوده و آن‌ها را به بردار تبدیل نمی‌کند. نوآوری این مقاله در تعریف ماتریس کواریانس بوده که مستقیماً از روی ماتریس تصاویر انجام می‌شود؛ به این صورت که اگر تصاویر $\mathbf{F}_j, j=1, \dots, M$ با متوسط صفر و اندازه $r \times c$ در اختیار باشد، ماتریس کواریانس تصاویر از رابطه‌ی
$$\mathbf{C}_F = \frac{1}{M} \sum_{j=1}^M \mathbf{F}_j^T \cdot \mathbf{F}_j$$
 به دست می‌آید که ابعاد آن $c \times c$ است. سپس بردارهای ویژه‌ی این ماتریس محاسبه شده که به تعداد c بردار با ابعاد $c \times I$ هستند و به عنوان پایه‌های فضا در نظر گرفته می‌شوند که تصاویر به روی آن‌ها نگاشت می‌یابند. اگر به تعداد $k < c$ از این بردارهای ویژه در ماتریس $\mathbf{V}_{c \times k}$ قرار گیرند، نگاشت تصویر \mathbf{F} روی فضای جدید خواهد بود که فقط در راستای ستون‌ها کاهش ابعاد دارد.

بعد از این پژوهش، مراجع [۹-۱۲] از ایده‌ی تحلیل مؤلفه‌های اصلی دوبعدی استفاده کرده و یک تبدیل دو طرفه ارائه داده‌اند که تحلیل مؤلفه‌های اصلی تعمیم‌یافته نیز نام‌گذاری شده است. به عنوان مثال برای نگاشت تصویر \mathbf{F} به

متغیرهای اسکالر با حروف کوچک ایتالیک، بردارها با حروف بزرگ ایتالیک و ماتریس‌ها با حروف بزرگ پررنگ غیر ایتالیک نمایش داده خواهند شد.

ایده‌ی تحلیل مؤلفه‌های اصلی، کاهش ابعاد یک مجموعه داده است که تعداد زیادی از متغیرهای آن به طور ذاتی مرتبط هستند. این کاهش ابعاد باید به گونه‌ای باشد که تا حد امکان، انرژی داده‌های اولیه و یا در واقع اطلاعات داده‌های اولیه حفظ شود [۳-۴]. فرض کنید ماتریس \mathbf{X} با ابعاد $p \times n$ در اختیار باشد که ستون‌های این ماتریس $\mathbf{X}_j, j=1, \dots, n$ ، شامل n بردار مشاهده با متوسط صفر و حاوی p متغیر می‌باشند.

$$\mathbf{X} = [X_1 \ X_2 \ \dots \ X_{n-1} \ X_n] \quad (1)$$

در این صورت ماتریس کواریانس این مجموعه بردار، طبق رابطه‌ی (۲) تعریف می‌گردد.

$$\mathbf{C}_X = E\{\mathbf{X}\mathbf{X}^T\} \quad (2)$$

پس \mathbf{C}_X یک ماتریس $p \times p$ است که $C(i,i)$ واریانس i امین متغیر (سطر i ام از ماتریس \mathbf{X})، و $C(i,j)$ کواریانس بین متغیرهای i ام و j ام (سطرهای i ام و j ام از ماتریس \mathbf{X}) است. با توجه به فرضیات مذکور ماتریس \mathbf{C}_X حقیقی و متقارن بوده و دارای p مقدار ویژه (λ_i) و بردار ویژه^۱ متعامد متناظر آنان (V_i) با ابعاد $p \times I$ است. در صورتی که مقادیر ویژه به ترتیب نزولی مرتب گردند $(\lambda_i \geq \lambda_{i+1})$ ، ماتریس بردارهای ویژه مطابق رابطه (۳) تعریف می‌شود که در ستون اول، بردار ویژه‌ی مربوط به بزرگ‌ترین مقدار ویژه، یعنی V_1 و در ستون آخر بردار ویژه‌ی مربوط به کوچک‌ترین مقدار ویژه، یعنی V_p جایگذاری می‌شود.

$$\mathbf{V} = [V_1 \ V_2 \ \dots \ V_p]_{p \times p} \quad (3)$$

حال بردارهای جدید یعنی Y_i ‌ها می‌توانند به صورت زیر جایگزین X_i ‌ها گردند.

$$Y_i = \mathbf{V}^T \cdot X_i \quad \text{for } i=1, \dots, p \quad (4)$$

از آنجاکه در ماتریس \mathbf{V} بردارهای ویژه بنا بر ارزش خود چیده شده‌اند می‌توان به جای تمام ماتریس \mathbf{V} ، ماتریس $\mathbf{V}_k = [V_1 \ V_2 \ \dots \ V_k]_{p \times k}$ را در نظر گرفت که در این صورت اندازه‌ی بردارهای جدید Y_i به $k \times I$ کاهش یافته و در نتیجه کاهش ابعاد صورت می‌گیرد. در این صورت:

$$Y_j = [y_{1j} \ y_{2j} \ \dots \ y_{kj}]^T; \quad y_{ij} = \sum_{\ell=1}^p v_{i\ell} \cdot x_{\ell j} \quad (5)$$

$$\text{for } j=1, \dots, p; \quad i=1, \dots, k.$$

اما به دست آوردن مؤلفه‌های جدید با وجود توانایی بالا در پردازش اطلاعات و کاهش ابعاد داده‌ها، با مشکلاتی در امر محاسبه مواجه است. به عنوان اولین مشکل می‌توان به این مورد اشاره کرد که برای اعمال تحلیل مؤلفه‌های اصلی به

¹Eigen-Values and Eigen-Vectors

بیشتر باشد، این اتفاق خواهد افتاد و روش حداقل مربعات در ارائه‌ی یک پاسخ منحصر به فرد ناتوان می‌ماند [۲۲]. به‌طور کلی روش حداقل مربعات از نظر دقت پیش‌بینی و قابل تفسیر بودن مدلی که ارائه می‌دهد نیز ضعیف عمل می‌کند [۲۲].

برای بهبود عملکرد روش حداقل مربعات، الگوریتم‌هایی مانند رگرسیون ریبج^۱ پیشنهاد شده‌اند که مجموع مربعات را تحت قید نرم دو ($\|\cdot\|_2$) بردارهای ماتریس \mathbf{B} ، حداقل می‌کنند. رگرسیون ریبج با اعمال قید روی اندازه‌ی عناصر \mathbf{B} ، موجب کوچک شدن آن‌ها می‌شود. این روش با حل رابطه‌ی (۷) ضرایب را به دست می‌آورد [۲۱].

$$\mathbf{B}^{\text{ridge}} = \arg \min_{\mathbf{B}} \left\{ \sum_{i=1}^n (Y_i - \mathbf{B} \cdot X_i)^2 + \lambda \|\mathbf{B}\|_2^2 \right\} \quad (7)$$

$\lambda \geq 0$ پارامتری است که مقدار انقباض عناصر \mathbf{B} را کنترل می‌کند. هر چه λ بزرگ‌تر باشد عناصر \mathbf{B} کوچک‌تری حاصل خواهد شد. اگر از رابطه‌ی (۷) مشتق گرفته و مساوی صفر قرار دهیم؛ عناصر \mathbf{B} به‌سادگی محاسبه خواهد شد که در رابطه‌ی (۸) آورده شده است [۲۱].

$$\mathbf{B}^{\text{ridge}} = (\mathbf{X} \cdot \mathbf{X}^T + \lambda \mathbf{I})^{-1} \mathbf{X}^T \mathbf{Y} \quad (8)$$

در این رابطه \mathbf{I} ماتریس واحد با ابعاد $p \times p$ است و ماتریس‌های \mathbf{X} و \mathbf{Y} ماتریس مشاهدات ورودی و ماتریس تبدیل یافته می‌باشند. در حل رگرسیون ریبج قبل از انجام عملیات معکوس کردن، یک مقدار مثبت به درایه‌های روی قطر $\mathbf{X} \cdot \mathbf{X}^T$ اضافه گردیده و اگر هم $\mathbf{X} \cdot \mathbf{X}^T$ مرتبه‌ی کامل نباشد، مسئله منفرد نخواهد شد و جواب منحصر به فردی حاصل می‌شود. این موضوع انگیزه‌ی مطرح کردن رگرسیون ریبج در مباحث آماری بوده است. رگرسیون ریبج اگرچه تا حدی ضرایب را کاهش می‌دهد اما مانع از این می‌شود که ضرایب، دقیقاً صفر برآورد شوند و منجر به عدم حذف متغیر از مدل می‌گردد. این ایراد سبب شد که روش $LASSO$ مطرح شود. این روش، قید نرم یک بردار ضرایب ($\|\cdot\|_1$) را به مسئله‌ی حداقل مربعات خطای (۶) اضافه می‌کند که در رابطه‌ی (۹) مشاهده می‌شود [۲۱].

$$\mathbf{B}^{\text{Lasso}} = \arg \min_{\mathbf{B}} \left\{ \sum_{i=1}^n (Y_i - \mathbf{B} \cdot X_i)^2 + \lambda \|\mathbf{B}\|_1 \right\} \quad (9)$$

استفاده از قید نرم یک موجب می‌شود که جواب بسته‌ای برای مسئله‌ی $LASSO$ وجود نداشته باشد. اما به علت طبیعت این نرم، اگر λ به اندازه‌ی کافی بزرگ باشد تعدادی از اعضای مؤلفه‌ها، دقیقاً صفر برآورد می‌شوند. روش $LASSO$ علی‌رغم موفقیت‌هایی که در بسیاری از موقعیت‌ها به دست آورده، محدودیت‌هایی نیز دارد که به شرح زیر می‌باشند [۲۲].

فضای جدید از رابطه‌ی $\mathbf{L}^T \mathbf{F} \mathbf{R}$ استفاده کرده‌اند. به این ترتیب کاهش ابعاد در هر دو راستای سطر و ستون ممکن خواهد شد. برای محاسبه‌ی ماتریس‌های \mathbf{R} و \mathbf{L} ، مراجع [۸-۱۲] کمیته‌ی به نام ماتریس پراکنندگی مربوط به سطرها و ستون‌ها تعریف می‌کنند، سپس بردارهای ویژه‌ی این دو ماتریس محاسبه شده و مهم‌ترین آن‌ها به عنوان ماتریس‌های نگاشت \mathbf{L} و \mathbf{R} مورد استفاده قرار می‌گیرند.

هرچند هنوز یک مشکل در تحلیل مؤلفه‌های اصلی یک یا دوبعدی باقی مانده است. معمولاً تمامی این مؤلفه‌های اصلی هم ارزش نبوده و باید به نحوی ارزش آن‌ها جدا سازی گردند و یا در واقع به نحوی از همان ابتدا ضرایب صفر آن‌ها را شناسایی کرد. در اینجا بحث محاسبه‌ی مؤلفه‌های اصلی تُنک پیش می‌آید که سعی دارد با حفظ تقریبی خواص تحلیل مؤلفه‌های اصلی استاندارد، ضرایب را تُنک نماید به گونه‌ای که مؤلفه‌های اصلی، ترکیب خطی تعداد کمی از متغیرهای اولیه گردند. بهبود قابلیت تفسیر مؤلفه‌ها تنها مزیت تُنک کردن ضرایب نیست بلکه می‌توان متغیرهایی را که در همه‌ی مؤلفه‌ها ضریب صفر گرفته‌اند به‌طور کامل حذف کرد. علاوه بر این، ذخیره‌سازی ماتریس‌های تُنک و کار کردن با آن‌ها سهولت بیشتری دارد. پژوهش‌های [۱۳-۲۰] به ارائه‌ی روش‌های متعدد برای فرمول‌بندی و بیان تحلیل مؤلفه‌های اصلی تُنک پرداخته و با ترفندهای مختلف مانند آستانه گذاری، روش‌های جستجوی حریم‌ها^۱ و الگوریتم‌های بهینه‌سازی از نوع رگرسیون تعداد زیادی از درایه‌های آن‌ها را صفر کرده‌اند. این روش‌ها معمولاً در حالت متغیرهای یک‌بعدی عمل می‌نمایند.

جهت یافتن پاسخ‌های تُنک بایستی به حل مسائل بهینه‌سازی پرداخت. ساده‌ترین حالت، روش حداقل مربعات خطا است که با کمینه کردن رابطه‌ی (۶)، تخمینی برای اعضای ماتریس \mathbf{B} به عنوان ماتریس مؤلفه‌های اصلی ارائه می‌دهد [۲۱].

$$\Psi(\mathbf{B}) = \sum_{i=1}^n (Y_i - \mathbf{B} \cdot X_i)^2; \quad \mathbf{B} = [B_1, \dots, B_p] \quad (6)$$

که در آن X_i ستون i ام مربوط به مشاهده‌ی i ام از ماتریس ورودی \mathbf{X} و Y_i نگاشت بردار مشاهده‌ی i ام روی مؤلفه‌های اصلی موجود در ماتریس \mathbf{B} می‌باشند. حل این رابطه به صورت $\mathbf{B} = (\mathbf{X} \cdot \mathbf{X}^T)^{-1} \mathbf{X}^T \cdot \mathbf{Y}$ است. حال اگر سطرهای \mathbf{X} مستقل خطی نباشند؛ در این صورت ماتریس \mathbf{X} مرتبه‌ی کامل نیست. بنابراین $\mathbf{X} \cdot \mathbf{X}^T$ ماتریس منفرد خواهد بود و روش حداقل مربعات، ماتریس \mathbf{B} منحصر به فردی ارائه نمی‌دهد. همچنین در مواردی که تعداد متغیرها از مشاهدات

²Ridge regression

¹Greedy Algorithm

در مقاله‌ی حاضر به دنبال یافتن روشی جهت تحلیل مؤلفه‌های اصلی تَنک برای تصاویر به صورتی هستیم که ملزم به تبدیل یک تصویر دوبعدی به بردار نباشیم. تبدیل تصویر دوبعدی به بردار در همان وهله‌ی اول موجب از بین رفتن موقعیت مکانی پیکسل‌ها نسبت به یکدیگر شده و در واقع همبستگی بین پیکسل‌های نزدیک به هم را از بین می‌برد.

در ادامه مقاله به بررسی و تعمیم ایده‌های موجود جهت استفاده در حالت دوبعدی خواهیم پرداخت. و در قسمت سوم نیز نتایج شبیه‌سازی ارائه خواهد شد.

۲- روش انتخابی برای تحقق تحلیل مؤلفه‌های اصلی تَنک

بعد از بررسی اجمالی روش‌های مختلف برای تحقق تحلیل مؤلفه‌های اصلی تَنک، الگوریتم ارائه‌شده در مرجع [۱۴] جهت تعمیم آن به حالت دوبعدی انتخاب گردید. ابتدا این الگوریتم به تفصیل تعریف می‌شود:

فرض می‌کنیم که ماتریس مشاهدات \mathbf{X} دارای ابعاد $p \times n$ است. p متغیر n بار مشاهده شده‌اند. هدف آن است که ماتریس \mathbf{B} به سایز $p \times k$ ($k < p$) به عنوان فضای پایه یا مؤلفه‌های اصلی یافت شود تا بردارهای جدید با ابعاد کاهش یافته به صورت ذیل به دست آیند:

$$Y_i = \mathbf{B}^T X_i, \text{ for } i = 1, \dots, n \quad (11)$$

که در فرم ماتریسی به صورت $\mathbf{Y} = \mathbf{B}\mathbf{X}$ خواهند بود. جهت بازیابی احتیاج به ماتریس دیگر \mathbf{A} به سایز $p \times k$ به نحوی هستیم که:

$$X_i = \mathbf{A}Y_i \text{ for } i = 1, \dots, n \quad (12)$$

و در فرم ماتریسی به صورت $\mathbf{X} = \mathbf{A}\mathbf{Y}$ برقرار گردد. روش‌های بهینه‌سازی برای به دست آوردن ماتریس‌های تَنک \mathbf{A} و \mathbf{B} برای ورودی‌های برداری توسط [۱۴] به صورت رابطه‌ی (۱۳) قابل حل است:

$$(\hat{\mathbf{A}}, \hat{\mathbf{B}}) = \arg \min_{\mathbf{A}, \mathbf{B}} \left\{ \begin{array}{l} \sum_{i=1}^n \|X_i - \mathbf{A}\mathbf{B}^T X_i\|_2^2 \\ + \lambda_2 \sum_{j=1}^k \|B_j\|_2^2 + \sum_{j=1}^k \lambda_{1,j} \|B_j\|_1 \end{array} \right\} \quad (13)$$

subject to $\mathbf{A}^T \mathbf{A} = \mathbf{I}_{k \times k}$

X_i نشان‌دهنده‌ی ستون i ام از ماتریس \mathbf{X} یعنی i امین مشاهده از بردار متغیرها است. در این مسئله k مؤلفه‌ی اصلی اول مدنظر است؛ هرچند محاسبه‌ی تمام مؤلفه‌ها (p مؤلفه) نیز ممکن است. ماتریس‌های \mathbf{A} و \mathbf{B} نیز به صورت $\mathbf{A}_{p \times k} = [A_1, \dots, A_k]$ و $\mathbf{B}_{p \times k} = [B_1, \dots, B_k]$ می‌باشند. یک مقدار مشابه λ_2 برای تمامی k مؤلفه استفاده شده اما برای

اولین محدودیت آن است که تعداد متغیرهایی که در روش $LASSO$ انتخاب می‌شوند به تعداد مشاهدات محدود می‌گردد. در صورتی که در بعضی از موارد مانند داده‌های یک ریزآرایه، هزاران متغیر وجود دارد که ژن‌های مختلف هستند ($p > 1000$) درحالی‌که تعداد نمونه‌های در دسترس کمتر از ۱۰۰ است ($n < 100$). حال روش $LASSO$ فقط قادر است حداکثر n ضریب را انتخاب کند که در برخی موارد رضایت‌بخش نیست.

دومین محدودیت، زمانی است که بین متغیرها همبستگی بسیار بالایی وجود داشته باشد. در این صورت روش $LASSO$ مایل است که از آنان تنها یک متغیر را انتخاب نموده و اهمیتی نمی‌دهد که کدام متغیر انتخاب می‌شود.

بررسی‌ها نشان می‌دهند در حالت‌های معمولی که تعداد مشاهدات بیشتر از متغیرها بوده و همبستگی بسیار زیادی نیز بین متغیرها وجود دارد، عملکرد رگرسیون ریج بهتر از روش $LASSO$ است [۲۲]. در ادامه، الگوریتم $LARS-EN$ ^۱ معرفی شد که تعمیمی برای روش $LASSO$ است و نقطه‌ضعف‌های آن را برطرف می‌نماید. این الگوریتم از طریق اعمال قید نُرم یک و دو به صورت هم‌زمان، مزایای رگرسیون ریج و روش $LASSO$ را ترکیب می‌کند. برای مقادیر غیر منفی λ_1 و λ_2 ضرایب الاستیک نت به صورت رابطه‌ی (۱۰) تخمین زده می‌شوند [۲۱]:

$$\mathbf{B}^{en} = (1 + \lambda_2) \arg \min_{\mathbf{B}} \left\{ \begin{array}{l} \sum_{i=1}^n (Y_i - \mathbf{B}X_i)^2 \\ + \lambda_1 \|\mathbf{B}\|_1 + \lambda_2 \|\mathbf{B}\|_2^2 \end{array} \right\} \quad (10)$$

روش $LARS-EN$ با استفاده از الگوریتم الاستیک نت مسئله را حل می‌کند که در حالت $\lambda_1 = 0$ به رگرسیون ریج و در حالت $\lambda_2 = 0$ به $LASSO$ تبدیل می‌شود. قید نُرم دو تضمین می‌کند که در هر شرایطی جواب منحصر به فرد به دست آید. قید نُرم یک نیز، بردار تَنکی از ضرایب در اختیار می‌گذارد. علاوه بر این، استفاده از قید نُرم دو باعث می‌شود متغیرهایی که بسیار همبسته هستند ضرایب رگرسیون مشابهی اتخاذ نمایند که اثر گروهی نامیده می‌شود. به ازای یک مقدار ثابت λ_2 ، الگوریتم $LARS-EN$ به مانند الاستیک نت ابتدا مسئله را برای تمامی مقادیر λ_1 حل می‌کند و بار محاسباتی آن به اندازه‌ی الگوریتم حداقل مربعات است. λ_2 در حالتی که متغیرها بیش از مشاهدات هستند ($p > n$) یک عدد مثبت انتخاب می‌شود. در این صورت تمامی متغیرها می‌توانند در مدل ارائه‌شده توسط الاستیک نت حضور داشته باشند و محدودیت اول $LASSO$ که در این مورد بود؛ برطرف می‌شود [۲۲].

^۱Least Angle Regression- Elastic Net

اعمال می‌شوند نادیده گرفت و عبارت $\sum_{i=1}^n \|X_i - \mathbf{A}\mathbf{B}^T X_i\|_2^2$ را تحت قید $\mathbf{A}^T \mathbf{A} = \mathbf{I}_{k \times k}$ حدافل کرد. ثابت می‌شود اگر تجزیه به مقدار تکین روی $(\mathbf{X}^T \mathbf{X})\mathbf{B}$ اعمال شود، ماتریس \mathbf{A} به صورت زیر می‌تواند در نظر گرفته شود. [۱۴]:

$$(\mathbf{X}^T \mathbf{X})\mathbf{B} = \mathbf{U}_1 \mathbf{D}_1 \mathbf{V}_1^T \rightarrow \hat{\mathbf{A}} = \mathbf{U}_1 \mathbf{V}_1^T \quad (15)$$

توسط ماتریس \mathbf{A} جدید، ماتریس \mathbf{B} جدید محاسبه شده و الگوریتم ادامه می‌یابد. در نهایت β_j های به دست آمده نرمالیزه شده و به عنوان بردارهای مؤلفه‌های اصلی که \mathbf{X} هستند مورد استفاده قرار می‌گیرند. یکی از دلایل انتخاب این الگوریتم این است که با اندک تغییری در ورودی الگوریتم می‌توان آن را برای حالت دوبعدی حل کرد که هدف مورد نظر ما است.

در مقابل روشی که به طور هم‌زمان تمام بردارهای ضرایب مورد نظر را به دست می‌آورد؛ روش دیگری نیز ارائه شده که یکی پس از دیگری بردارهای ضرایب را محاسبه می‌کند. در مرجع [۲۵] مزایای این روش نسبت به روش هم‌زمان بیان گردیده است؛ مثلاً اگر روش محاسبه‌ی پی‌درپی مؤلفه‌ها، یک‌بار برای به دست آوردن k مؤلفه و بار دیگر برای محاسبه‌ی $k+1$ مؤلفه اجرا شود؛ k مؤلفه‌ی اول محاسبه‌شده در هر دو مورد، دقیقاً یکسان هستند. اما در روش محاسبه‌ی هم‌زمان بردارهای ضرایب، مؤلفه‌های محاسبه‌شده در حالتی که k مؤلفه نیاز است نسبت به حالتی که $k+1$ مؤلفه مورد نظر است، کاملاً متفاوت هستند. علاوه بر این، زمان اجرای الگوریتم پی‌درپی نسبت به روش هم‌زمان کمتر است.

۳- تحلیل مؤلفه‌های اصلی دوبعدی \mathbf{X}

در این مقاله ما می‌خواهیم مسئله را برای حالت دوبعدی تحلیل مؤلفه‌های اصلی \mathbf{X} تعریف کنیم. با دقت در رابطه‌ی (۱۴) درمی‌یابیم که اگر α_j مشخص باشد؛ محاسبه‌ی β_j تنها به $\mathbf{X}^T \mathbf{X}$ وابسته است. به عبارتی برای حل رابطه‌ی (۱۴) دانستن $\mathbf{X}^T \mathbf{X}$ کفایت می‌کند. یعنی اگر خود ماتریس \mathbf{X} که شامل مشاهدات مختلف از متغیرها می‌باشد در اختیار نبود، مشخص بودن $\mathbf{X}^T \mathbf{X}$ هم برای حل رابطه‌ی (۱۴) کافی است؛ عبارت $\mathbf{X}^T \mathbf{X}$ ، ماتریس کوواریانس نمونه‌ای را تداعی می‌نماید.

از همین نکته برای رسیدن به تحلیل مؤلفه‌های اصلی دوبعدی \mathbf{X} استفاده نموده و به جای $\mathbf{X}^T \mathbf{X}$ ، ماتریس کوواریانس تصویر و یا ماتریس کوواریانس چپ و راست را که در تحلیل مؤلفه‌های اصلی دوبعدی مطرح شد؛ جایگزین می‌نماییم. در حقیقت دیگر ماتریس \mathbf{X} را تشکیل نخواهیم داد که تصاویر به شکل بردار درآمده و در سطرهاى آن قرار بگیرند؛ بلکه کل عبارت $\mathbf{X}^T \mathbf{X}$ را ورودی الگوریتم در نظر

مقید کردن بردارهای ضرایب مختلف، می‌توان برای هر مؤلفه از $\lambda_{1,j}$ متفاوتی بهره گرفت [۱۴].

جهت روشن شدن رابطه‌ی (۱۳) باید تصریح کرد که قسمت دوم و سوم این رابطه قیدهای نرم دو و نرم یک هستند که موجب می‌شوند مسئله جواب منحصر به فردی داشته باشد؛ در عین حال که \mathbf{X} بودن پاسخ‌ها نیز تحقق می‌یابد. اما در قسمت اول این رابطه، ابتدا متغیرهای مربوط به مشاهده‌ی i ام توسط $\mathbf{B}^T X_i$ به روی بردارهایی که در ماتریس \mathbf{B} قرار دارند؛ نگاشت داده می‌شوند سپس با ضرب شدن در ماتریس \mathbf{A} (یعنی عبارت $\mathbf{A}\mathbf{B}^T X_i$) به فضای اولیه بازمی‌گردند. کل عبارت $X_i - \mathbf{A}\mathbf{B}^T X_i$ نیز بیانگر خطای بازسازی است. بنابراین می‌توان این‌گونه بیان کرد که \mathbf{B} ماتریس تبدیل و \mathbf{A} ماتریس عکس تبدیل است [۲۳].

حل این مسئله یا در واقع به دست آوردن ماتریس‌های \mathbf{A} و \mathbf{B} طی مراحل زیر و در یک الگوریتم تکراری صورت می‌گیرد. مشابه قسمت‌های قبل، داده‌ها در ماتریس \mathbf{X} با ابعاد $n \times p$ قرار دارند و اگر از تجزیه به مقادیر منفرد استفاده شود، این ماتریس به صورت $\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^T$ خواهد بود [۱۴]. حال جهت کاهش ابعاد ابتدا بایستی تعداد مؤلفه‌های اصلی که قرار است تخمین زده شوند؛ مشخص گردد ($k < p$ مؤلفه). ماتریس \mathbf{A} توسط k ستون اول \mathbf{V} که در واقع k بردار اول مؤلفه‌های اصلی استاندارد هستند؛ به صورت $\mathbf{A} = \mathbf{V}[:, 1:k]$ مقداردهی اولیه می‌شود. با معلوم بودن ماتریس \mathbf{A} ، اندک تغییری در نوشتن رابطه‌ی (۱۳) به کمک قضایای بهینه‌سازی در [۱۴] به k مسئله‌ی الاستیک نت مجزا برای هر کدام از ستون‌های \mathbf{B} به صورت رابطه (۱۴) می‌انجامد:

$$\begin{aligned} \hat{\beta}_j &= \arg \min_{\beta_j} \left\{ \left\| X\alpha_j - X\beta_j \right\|_2^2 + \lambda_2 \left\| \beta_j \right\|_2^2 + \lambda_{1,j} \left\| \beta_j \right\|_1 \right\} \\ &= \arg \min_{\beta_j} \left\{ \left(\alpha_j - \beta_j \right)^T \mathbf{X}^T \mathbf{X} \left(\alpha_j - \beta_j \right) + \lambda_2 \left\| \beta_j \right\|_2^2 + \lambda_{1,j} \left\| \beta_j \right\|_1 \right\} \\ &\text{subject to } \alpha_j^T \alpha_j = 1 \end{aligned} \quad (14)$$

با مشخص بودن ماتریس \mathbf{A} ، ستون‌های آن یعنی $\alpha_j, j = 1, \dots, k$ و نیز $\mathbf{X}\alpha_j$ معلوم خواهند بود. پس در رابطه‌ی (۱۴) که به شکل مسئله‌ی رگرسیون است، هدف تخمین $\beta_j, j = 1, \dots, k$ است که قیدهای نرم یک و دو نیز روی آن‌ها اعمال شده و یک مسئله‌ی الاستیک نت را رقم زده است. مراجع [۲۳-۲۴] به طور کامل روش حل مسئله‌ی الاستیک نت را بررسی کرده‌اند که در این مقاله مجال پرداختن به آن وجود ندارد.

حال با داشتن ماتریس \mathbf{B} ، بایستی ماتریس \mathbf{A} را به روزرسانی کرد. با توجه به اینکه ماتریس \mathbf{B} محاسبه شده و معلوم است می‌توان قیدهای نرم یک و دو را که روی بردارهای ضرایب

۲. مقداردهی اولیه‌ی A^L و A^R با اعمال تجزیه به مقدار منفرد روی ماتریس‌های کوواریانس چپ و راست انجام می‌گیرد.
 ۳. با معلوم بودن ماتریس‌های A^L و A^R ، مسئله به حل رابطه‌ی (۱۸) منجر می‌شود. پیش‌ازین اشاره شد که رابطه‌ی مذکور یک مسئله‌ی رگرسیون الاستیک نت است. به دلیل اینکه به دنبال تبدیل دوبعدی دوطرفه هستیم؛ بردارهای β_j چپ و راست را محاسبه می‌نماییم که از بردارهای α_j چپ و راست و نیز ماتریس‌های کوواریانس چپ و راست حاصل می‌شوند.

$$\hat{\beta}_j^L = \arg \min_{\beta_j} \beta_j^T (S^L + \lambda_2 \mathbf{I}) \beta_j - 2(\alpha_j^L)^T S^L \beta_j + \lambda_1 \|\beta_j\|_2 \quad (19)$$

$$\hat{\beta}_j^R = \arg \min_{\beta_j} \beta_j^T (S^R + \lambda_2 \mathbf{I}) \beta_j - 2(\alpha_j^R)^T S^R \beta_j + \lambda_1 \|\beta_j\|_2$$

۴. بعد از محاسبه‌ی β_j بایستی α_j را به‌روزرسانی کرد. مرجع [۲۵] روش متفاوتی نسبت به مرجع [۱۴] برای به‌روزرسانی α_j در پیش می‌گیرد؛ اما نتایج یکسان خواهد شد. فرض کنید $A_{(j-1)}$ ماتریسی با ابعاد $p \times (j-1)$ باشد که بردارهای α_1 تا α_{j-1} به‌دست‌آمده تا این مرحله، ستون‌های آن را تشکیل می‌دهند و رابطه‌ی $A_{(j-1)}^T A_{(j-1)} = I_{(j-1) \times (j-1)}$ در مورد آن صادق است. در این صورت از طریق رابطه‌ی زیر، α_j را به دست می‌آوریم:

$$q = (I - A_{(j-1)} A_{(j-1)}^T) X^T X \beta_j \rightarrow \alpha_j = \frac{q}{\|q\|_2} \quad (20)$$

در واقع باید هر دو بردار β_j^L و β_j^R را در رابطه‌ی (۲۰) جایگزین کرده و بردارهای α_j^L و α_j^R را به دست آورد. لازم به ذکر است که تا این مرحله، $(j-1)$ ستون از ماتریس A محاسبه‌شده و عبارت $A \cdot A^T$ که در رابطه‌ی (۲۰) حضور دارد، تنها زمانی $I_{p \times p}$ خواهد شد که تمام p ستون آن محاسبه شود.

۵. مراحل ۳ و ۴ تا رسیدن به همگرایی ادامه می‌یابند و بعد از نهایی شدن بردارهای β_j و α_j چپ و راست سراغ β_{j+1} و α_{j+1} چپ و راست می‌رویم تا اینکه تمام بردارهای ضرایب موردنظر محاسبه شوند. در واقع تفاوت پیاده‌سازی الگوریتم تحلیل مؤلفه‌های اصلی تُنک در مرجع [۲۵] و [۱۴] این است که در مرجع [۲۵] بردارهای مؤلفه‌های اصلی یکی پس از دیگری محاسبه می‌شوند و هر بار که الگوریتم اجرا شده و همگرا می‌شود یک ستون از A و یک ستون از B خروجی آن خواهد بود و بایستی به تعداد ستون‌های موردنیاز ماتریس A و B مراحل ذکرشده را تکرار کرد. اما مرجع [۱۴] بردارهای ضرایب را هم‌زمان محاسبه می‌کند؛ یعنی بعد از محاسبه‌ی کامل ماتریس B از طریق مسئله‌ی الاستیک نت، ماتریس A را به‌روزرسانی می‌کند و این مراحل را تا جایی تکرار می‌کند که الگوریتم همگرا شده و تفاوتی در ماتریس A و B ایجادشده در تکرارهای متوالی مشاهده نگردد.

تا اینجای کار با بهره گرفتن از روابطی که برای تحلیل مؤلفه‌های اصلی تُنک مطرح‌شده بود و تنها با تغییر ورودی الگوریتم از X به $X^T X$ ، حالت دوبعدی تحلیل مؤلفه‌های اصلی تُنک محقق شده

گرفته و آن را با ماتریس کوواریانس تصویر جایگزین می‌کنیم. به این ترتیب با استفاده از رابطه‌ای که برای تحلیل مؤلفه‌های اصلی تُنک به‌دست‌آمده و تنها به‌وسیله‌ی جایگزینی ورودی این الگوریتم با ماتریس کوواریانس تصویر، تحلیل مؤلفه‌های اصلی دوبعدی تُنک محقق می‌شود.

برای به دست آوردن ماتریس کوواریانس یک مجموعه تصویر، نیازی نیست آن‌ها را به بردار تبدیل کنیم؛ بلکه مستقیماً از روی ماتریس تصاویر محاسبه می‌شود. جهت یادآوری باید ذکر کرد که اگر به تعداد M تصویر F_j با ابعاد $r \times c$ در اختیار بوده و میانگین کل آن‌ها با \bar{F} نمایش داده شود؛ در این صورت ماتریس کوواریانس چپ و راست به شکل زیر محاسبه می‌شوند [۱۲].

$$S^L = \frac{1}{M} \sum_{j=1}^M (F_j - \bar{F})(F_j - \bar{F})^T \quad (16)$$

$$S^R = \frac{1}{M} \sum_{j=1}^M (F_j - \bar{F})^T (F_j - \bar{F})$$

برای داشتن یک تبدیل دوطرفه باید بردارهای ویژه چپ و راست را به دست آورد، که به ترتیب از طریق جایگزینی $X^T X$ در رابطه‌ی (۱۴) با ماتریس کوواریانس چپ و راست حاصل می‌شوند.

$$\hat{\beta}_j^L = \arg \min_{\beta_j} (\alpha_j - \beta_j)^T S^L (\alpha_j - \beta_j) + \lambda_2 \|\beta_j\|_2^2 + \lambda_1 \|\beta_j\|_2 \quad (17)$$

$$\hat{\beta}_j^R = \arg \min_{\beta_j} (\alpha_j - \beta_j)^T S^R (\alpha_j - \beta_j) + \lambda_2 \|\beta_j\|_2^2 + \lambda_1 \|\beta_j\|_2$$

روابط (۱۷) را می‌توان با ضرب کردن پرانتزها در یکدیگر و حذف عباراتی که متغیر β_j در آن‌ها وجود ندارد، به شکل ساده‌تر زیر نوشت:

$$\hat{\beta}_j^L = \arg \min_{\beta_j} \beta_j^T (S^L + \lambda_2 \mathbf{I}) \beta_j - 2\alpha_j^T S^L \beta_j + \lambda_1 \|\beta_j\|_2 \quad (18)$$

$$\hat{\beta}_j^R = \arg \min_{\beta_j} \beta_j^T (S^R + \lambda_2 \mathbf{I}) \beta_j - 2\alpha_j^T S^R \beta_j + \lambda_1 \|\beta_j\|_2$$

مجدداً یادآوری می‌شود معادلات (۱۸) مربوط به مرحله‌ی اول الگوریتم تکرار است که بردارهای α_j معلوم هستند و محاسبه‌ی بردارهای ضرایب β_j مدنظر است. مسلماً بعد از محاسبه‌ی بردارهای β_j ، بایستی جهت به‌روز کردن بردارهای α_j که در بازسازی داده‌های اولیه شرکت دارند، اقدام کرد. به دلیل اینکه تبدیل دوبعدی دوطرفه موردنظر است و بردارهای ضرایب β_j چپ و راست محاسبه می‌شوند؛ متناظر با آن‌ها بردارهای α_j چپ و راست خواهیم داشت. پس مراحل عملکرد الگوریتم تکرار برای تحلیل مؤلفه‌های اصلی دوبعدی تُنک به شرح زیر خلاصه می‌شود.

۱. ابتدا تعداد بردارهای پایه (یا مؤلفه‌های اصلی) چپ و راست که قرار است تخمین زده شوند، مشخص می‌گردد.

در قسمت بعدی تحلیل مؤلفه‌های اصلی دوبعدی دوطرفه را در فشرده‌سازی یک تصویر به کار گرفته و نتایج حاصل را با تحلیل مؤلفه‌های اصلی دوبعدی و تبدیل کسینوسی گسسته مقایسه خواهیم کرد.

۴- نتایج شبیه‌سازی

به‌طور کلی پژوهش‌هایی که روی تحلیل مؤلفه‌های اصلی تئک صورت گرفته، همگی بر استخراج ویژگی تأکید داشته و از تحلیل مؤلفه‌های اصلی تئک به‌عنوان روشی یادکرده‌اند که تفسیر مؤلفه‌های اصلی را تسهیل می‌نماید. در این مقاله با توجه به توانایی بالای تحلیل مؤلفه‌های اصلی تئک در استخراج ویژگی و پیدا کردن الگوهای نهفته در داده‌ها، سعی می‌شود از این روش برای کاهش ابعاد داده و فشرده‌سازی تصویر نیز استفاده گردد.

ذکر این نکته نیز الزامی است که در کاربردهای قبلی، همواره یک پایگاه داده از تصاویر پردازش شده‌اند. اما در این مقاله ما بر روی یک تصویر نیز تمرکز کرده‌ایم؛ یعنی پایگاه داده را از بلوک‌های یک تصویر می‌سازیم.

در آزمایشات اولیه، بلوک‌های 8×8 یک تصویر به عنوان پایگاه داده‌ی دو بعدی در نظر گرفته شد. به همین دلیل اگر N بلوک $F_i, i=1, \dots, N$ با میانگین صفر داشته باشیم؛ ماتریس کواریانس چپ و راست از طریق رابطه‌ی (۲۴) محاسبه می‌شوند.

$$S^L = \frac{1}{N} \sum_{i=1}^N F_i F_i^T, \quad S^R = \frac{1}{N} \sum_{i=1}^N F_i^T F_i \quad (24)$$

از روی این رابطه ملاحظه می‌شود که ابعاد ماتریس کواریانس چپ و راست 8×8 می‌باشد و حداکثر ۸ بردار ویژه با ابعاد 8×1 خواهند داشت. در اینجا روش مورد نظر را با روش تحلیل مؤلفه‌های اصلی دوبعدی دوطرفه [۱۲] مقایسه خواهیم کرد. در روش مرجع [۱۲] بردار ویژه‌ی ماتریس‌های کواریانس چپ و راست را برداشته و ماتریس‌های L و R را تشکیل می‌دهیم. البته تعداد بردارهای ویژه‌ای که در ماتریس L و R قرار می‌دهیم؛ می‌توانند متفاوت باشند. در این صورت طبق رابطه‌ی (۲۵) یک بلوک 8×8 به یک بلوک $d \times d$ تبدیل می‌شود.

$$(Y_i)_{d \times d} = (L^T)_{d \times 8} (F_i)_{8 \times 8} R_{8 \times d} \quad (25)$$

از آنجایی که ستون‌های ماتریس L و R متعامد هستند؛ رابطه‌ی (۲۶) بازسازی را انجام می‌دهد.

$$(F_i)_{8 \times 8} = L_{8 \times d} (Y_i)_{d \times d} (R^T)_{d \times 8} \quad (26)$$

شکل (۲) تصویر بازسازی شده‌ی Lena را توسط تحلیل مؤلفه‌های اصلی دوبعدی دوطرفه و به ازای مقادیر مختلف d نشان می‌دهد. در حالت $d = 2$ که با ۲ بردار مؤلفه‌ی اصلی اقدام به بازسازی هر بلوک تصویر می‌نماییم، نتیجه‌ی به‌دست‌آمده کیفیت خوبی ندارد. به‌ازای $d = 4$ تصویر بازسازی شده کیفیت نسبتاً قابل قبولی دارد تا اینکه در $d = 8$ تصویر بازسازی شده کاملاً بر تصویر اصلی منطبق خواهد شد. در صورت استفاده از تحلیل مؤلفه‌های اصلی دوبعدی دوطرفه برای فشرده‌سازی تصویر، به‌جای یک تصویر با N بلوک به

است. برای حل الگوریتم تکراری که مطرح شد می‌توان از جعبه‌ابزار مرجع [۲۵] (Spasm toolbox) که برای تحلیل مؤلفه‌های اصلی تئک طراحی شده استفاده کرد و تغییر در ورودی آن ایجاد نمود. همچنین می‌توان با جعبه‌ابزار بهینه‌سازی نرم‌افزار متلب، پیاده‌سازی را انجام داد. ما هر دو روش را به کار برده و نتایج یکسانی گرفته‌ایم.

بعد از محاسبه‌ی بردارهای $\beta_j, j=1, \dots, k$ و $\alpha_j, j=1, \dots, k$ چپ و راست از طریق الگوریتم تکرار ارائه‌شده، آن‌ها را در یک ماتریس قرار می‌دهیم. k تعداد بردارهای مؤلفه اصلی مورد نظر است که نرخ فشرده‌سازی را تعیین می‌کند. در حالت تحلیل مؤلفه‌های اصلی دوبعدی تئک که روی بلوک‌های تصویر انجام می‌شود؛ حداکثر مقدار پارامتر k می‌تواند به اندازه‌ی بلوک‌های تصویر باشد. یادآوری می‌شود که پس از محاسبه‌ی بردارهای α_j و β_j چپ و راست، ماتریس‌های زیر را خواهیم داشت:

$$B^L = [\beta_1^L, \dots, \beta_k^L]_{8 \times k}, \quad B^R = [\beta_1^R, \dots, \beta_k^R]_{8 \times k} \quad (21)$$

$$A^L = [\alpha_1^L, \dots, \alpha_k^L]_{8 \times k}, \quad A^R = [\alpha_1^R, \dots, \alpha_k^R]_{8 \times k}$$

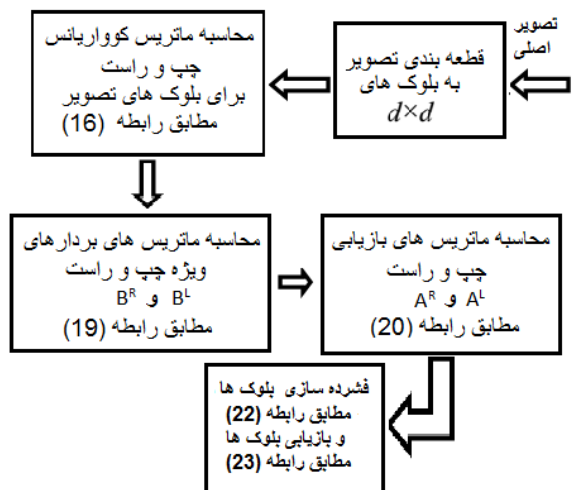
البته تعداد بردارهای ضرایب چپ و راست می‌توانند متفاوت باشند؛ اما معمولاً یکسان در نظر گرفته می‌شوند. همین ماتریس‌ها برای انتقال تمامی بلوک‌های تصویر (F_i) به فضای جدید و بازسازی آن‌ها کفایت می‌کنند. با این اوصاف نگاشت یک بلوک از تصویر به فضای جدید با رابطه‌ی (۲۲) بیان می‌شود:

$$(Y_i)_{k \times k} = (B^L)^T F_i B^R \quad (22)$$

به این ترتیب یک بلوک با ابعاد $d \times d$ به بلوک دیگری با ابعاد $k \times k$ تبدیل می‌گردد که اگر $k < d$ باشد؛ کاهش ابعاد حاصل می‌شود. البته می‌توان تنها از ماتریس تبدیل چپ یا راست استفاده کرد و تبدیل یک‌طرفه انجام داد که فقط در یک راستا کاهش ابعاد داشته باشیم. برای بازسازی بلوک F_i در گیرنده، با الهام از روابط تحلیل مؤلفه‌های اصلی تئک به صورت زیر عمل کرده‌ایم.

$$F_i = A^L Y_i (A^R)^T \quad (23)$$

شکل (۱) بلوک دیاگرامی از مراحل مختلف تحلیل مؤلفه‌های اصلی دوبعدی تئک ارائه می‌دهد.



شکل ۱: بلوک دیاگرام مراحل مختلف الگوریتم پیشنهادی

دهیم؛ می‌توان توسط رابطی (۲۸) بلوک F_i را به فضایی با ابعاد کمتر به نام G_i نگاشت داد.

$$(G_i)_{k \times k} = (B^L)^T F_i B^R \quad (28)$$

در اینجا جهت تحلیل مؤلفه‌های اصلی دو بعدی تنگ، دو عامل بایستی مشخص گردد: اول چه تعداد بردار ضرایب محاسبه شود و دوم چه تعداد از درایه‌های هر بردار ضریب، صفر باشد. بسته به اینکه چه عددی به دو پارامتر مذکور اختصاص داده شود؛ حالت‌های مختلفی برای بازسازی یک تصویر وجود خواهد داشت. مؤلفه‌ی اصلی اول باعث می‌شود عمده‌ی انرژی هر بلوک لحاظ گردد؛ به همین دلیل اگر در بردار ضریب اول درایه‌ی صفر را ایجاد نکنیم، نتایج بهتری حاصل می‌شود.

بازگرداندن بلوک‌ها به فضای اولیه، بر اساس رابطه‌ی (۲۹) و به کمک ماتریس‌های $A^R = [\alpha_1^R, \dots, \alpha_k^R]_{8 \times k}$ و $A^L = [\alpha_1^L, \dots, \alpha_k^L]_{8 \times k}$ صورت می‌گیرد که ستون‌های هر دو ماتریس متعامد هستند. اگر هیچ درایه‌ی صفری در بردارهای ضرایب ایجاد نشود، ماتریس‌های A و B یکسان خواهند بود و نتایج حاصل، مشابه تحلیل مؤلفه‌های اصلی دو بعدی می‌شود.

$$\hat{F}_i = A^L G_i (A^R)^T \quad (29)$$

شکل (۳) تصویر بازسازی شده‌ی *Lena*، *Peppers* و *Newyork* را نشان می‌دهد. تعداد عضوهای غیر صفر در هر بردار زیر تصاویر درج گردیده است.

برای دو تصویر *Lena* و *Newyork* ابعاد ماتریس‌های تبدیل و عکس تبدیل، 8×6 بوده و هر بلوک 8×8 به یک بلوک 6×6 تبدیل می‌شود. برای تصویر *Peppers* ابعاد ماتریس‌های تبدیل و عکس تبدیل، 8×4 بوده و هر بلوک 8×8 به یک بلوک 4×4 تبدیل می‌شود. به دلیل اینکه تصویر *Peppers* نسبت به *Lena* پیچیدگی کمتری دارد؛ با ۴ بردار ویژه تقریباً نتایجی مشابه با بازسازی تصویر *Lena* توسط ۶ بردار ویژه حاصل شده است. در صورتی که از آنجا که تصویر *Newyork* بسیار پیچیده‌تر از دو تصویر دیگر است، نتایج *SSIM* و *PSNR* شبیه‌سازی‌ها نشانگر آن است که با تعداد مشابه بردارهای ویژه با تصویر *Lena* کمتر از تصویر *Lena* می‌باشد. بدین جهت تصویر (ل) که تعداد عناصر صفر آن از همه کمتر است وضعیت بهتری از دیگر تصاویر دارد. در واقع به دلیل پیچیدگی بیشتر در تصویر *Newyork* لازم است تا برای فشرده‌سازی و بازسازی چنین تصاویری از تعداد بردار بیشتری و با تعداد صفرهای کمتری بهره گرفت. در جدول (۲) مقادیر *SSIM*، *PSNR* و *CR* مربوط به تصاویر شکل (۳) نشان داده شده است. مقایسه‌ی مقدار *SSIM* در دو حالت "الف" و "ب" و نیز "ه" و "و" و همچنین "ط" و "ی" نشان می‌دهد که اگر در بردار ضریب اول، درایه‌ی صفر ایجاد نشود کیفیت تصویر بازسازی شده بسیار بهتر می‌شود. با توجه به *CR* یکسان تصاویر الف، ب، ج و د از شکل ۳ با تصویر حالت $d=6$ از شکل ۲، دیده می‌شود که نتایج روش پیشنهادی با وجود در نظر گرفتن تعدادی صفر در ماتریس

ابعاد 8×8 بایستی N بلوک $d \times d$ به‌اضافه‌ی دو ماتریس L و R با ابعاد $8 \times d$ ذخیره گردد.

تصویر بازسازی شده با $d=2$

تصویر اصلی

تصویر بازسازی شده با $d=6$ تصویر بازسازی شده با $d=4$

شکل ۲: تصویر اصلی *Lena* و تصاویر بازسازی شده توسط مقادیر مختلف d توسط روش مؤلفه‌های اصلی دو بعدی دوطرفه [۱۲].

علاوه بر این، ماتریس میانگین بلوک‌های تصویر نیز بایستی ذخیره شود. با این اوصاف نرخ فشرده‌سازی در یک تصویر با ابعاد $r \times c$ که شامل N بلوک 8×8 می‌باشد؛ توسط رابطه‌ی (۲۷) محاسبه می‌گردد.

$$CR = \left(1 - \frac{N \times (d \times d) + (2 \times 8 \times d) + N}{N \times (8 \times 8)} \right) \times 100 \quad (27)$$

جهت محاسبه‌ی معیارهای عددی معیار متداول مقایسه‌ی کیفیت تصاویر (*SSIM*) [۲۶]، *PSNR* و همچنین میزان صرفه‌جویی در حافظه (*CR*) به‌ازای مقادیر مختلف d جدول ۱ آورده شده است.

جدول ۱: معیارهای ارزیابی کیفیت تصویر بازسازی شده‌ی *Lena* و نرخ فشرده‌سازی در تحلیل مؤلفه‌های اصلی دو بعدی دوطرفه

d	۲	۳	۴	۵	۶	۷
SSIM	۰,۴۷۰۶	۰,۶۴۴۷	۰,۷۵۳۴	۰,۸۳۱۳	۰,۸۹۳۸	۰,۹۴۹۰
PSNR	۲۳,۴۳	۲۵,۹۱	۲۸,۰۵	۳۰,۰۶	۳۲,۴۶	۳۵,۹۲
CR	۹۳,۶۰	۸۵,۷۷	۷۴,۸۰	۶۰,۷۲	۴۳,۵۰	۲۳,۱۷

در تحلیل مؤلفه‌های اصلی دو بعدی تنگ نیز تصویر را به بلوک‌های 8×8 تقسیم می‌کنیم، ابعاد ماتریس کوواریانس چپ و راست نیز 8×8 خواهد شد. سپس بردارهای مؤلفه‌ی اصلی β_j چپ و راست مطابق رابطه‌ی (۱۹) و بردارهای α_j چپ و راست مطابق رابطه‌ی (۲۰) محاسبه می‌شوند و به‌وسیله‌ی آن‌ها کاهش ابعاد هر بلوک محقق می‌شود. اگر k بردار مؤلفه چپ و راست را محاسبه کرده و ماتریس‌های $B^L = [\beta_1^L, \dots, \beta_k^L]_{8 \times k}$ و $B^R = [\beta_1^R, \dots, \beta_k^R]_{8 \times k}$ را تشکیل

بردارهای ویژه با حالتی که در ماتریس بردارهای ویژه هیچ صفری موجود نباشد مطابقت دارد.

تعمیم به پایگاه داده:

در این قسمت سعی می‌کنیم به این پرسش پاسخ دهیم که آیا می‌توان با در نظر گرفتن تعداد زیادی بلوک‌های 8×8 و اعمال تحلیل مؤلفه‌های اصلی دوبعدی تُنگک به آن‌ها، ماتریس‌های تبدیل و عکس تبدیلی به دست آورد که بتوانند هر تصویری را فشرده کرده و بازسازی نمایند؟ به این ترتیب به جای ذخیره‌سازی مثلاً M تصویر به اندازه‌ی 8×8 ، فقط دو ماتریس تُنگک به اندازه‌ی 8×6 ، بلوک‌های 6×6 هر تصویر و میانگین آنان برای بازیابی تمام M تصویر نیاز خواهد بود. با فرض آنکه هر تصویر دارای N بلوک می‌باشد. به این ترتیب مقدار ذخیره‌سازی مطابق رابطه‌ی (۳۰) به طور تقریب برابر ۴۲٪ خواهد بود.

$$CR = \left(1 - \frac{36NM + MN + 2(48)}{64NM}\right) \times 100 \quad (30)$$

به این منظور ابتدا ۶۰ تصویر مختلف از مجموعه پایگاه داده‌ی USC-SIPI^۱ استفاده گردید [۲۷]. بخش‌های مختلف این پایگاه داده عبارتند از:

Aerials: تصاویر هوایی گرفته‌شده از مناطق مختلف؛

Textures: ۱۵۵ تصویر از انواع مختلف بافت‌ها؛

Miscellaneous: انواع تصاویر استاندارد که در پژوهش‌های مربوط به پردازش تصویر به کار می‌روند.



و- [۸۰۶۰۶۰۶]



ه- [۶۰۶۰۶۰۶]



ح- [۸۰۷۰۶۰۵]



ز- [۸۰۴۰۴۰۴]



ی- [۸۶۶۶۶۶۶]



ط- [۶۶۶۶۶۶۶]



ل- [۸۸۸۶۶۶۶]



ک- [۸۰۳ و ۵ و ۷ و ۸]

شکل ۳: بازسازی *Lena*، *Peppers* و *New York* با ۶ و ۴ مؤلفه اصلی و با تعداد عضوهای غیر صفر ذکر شده در زیر تصاویر.

جدول ۲: معیارهای ارزیابی کیفیت تصویر بازسازی شده‌ی *Lena* و *Peppers* و نرخ فشرده‌سازی در تحلیل مؤلفه‌های اصلی تُنگک دوبعدی

د	ج	ب	الف	
۰,۸۴۷۹	۰,۸۲۷۸	۰,۸۸۱۵	۰,۸۲۴۲	SSIM
۳۰,۱۸	۲۹,۵۵	۳۱,۸۱	۲۵,۰۶	PSNR
۴۳,۵۰	۴۳,۵۰	۴۳,۵۰	۴۳,۵۰	CR
ح	ز	و	ه	
۰,۸۴۵۴	۰,۸۲۰۷	۰,۸۴۵۱	۰,۷۷۱۷	SSIM
۳۰,۶۲	۲۹,۶۵	۳۰,۷۱	۲۳,۳۰	PSNR
۷۴,۸۰	۷۴,۸۰	۷۴,۸۰	۷۴,۸۰	CR
ل	ک	ی	ط	
۰,۹۰۹۴	۰,۸۷۵۲	۰,۹۰۰۶	۰,۸۱۹۱	SSIM
۲۹,۱	۲۷,۹۴	۲۸,۶	۲۱,۱	PSNR
۴۳,۵۰	۴۳,۵۰	۴۳,۵۰	۴۳,۵۰	CR



ب- [۸۰۶۰۶۰۶۰۶]



الف- [۶۰۶۰۶۰۶۰۶]



د- [۸۰۷۰۶۰۵۰۴۰۳]



ج- [۸۰۴۰۴۰۴۰۴۰۴]

^۱University of Southern California- Signal and Image Processing Institute (USC-SIPI)

$$A^L = \begin{bmatrix} -0.3382 & 0.4557 & 0.4415 & -0.4304 & 0.2643 & -0.4432 \\ -0.3519 & 0.4296 & 0.2325 & 0.0556 & -0.1215 & 0.6245 \\ -0.3603 & 0.3029 & -0.1405 & 0.4792 & -0.3906 & -0.0070 \\ -0.3640 & 0.1077 & -0.4805 & 0.2747 & 0.1337 & -0.4868 \\ -0.3638 & -0.1001 & -0.4853 & -0.2544 & 0.4637 & 0.3164 \\ -0.3601 & -0.2986 & -0.1521 & -0.4943 & -0.3135 & 0.1265 \\ -0.3517 & -0.4322 & 0.2204 & -0.0688 & -0.4699 & -0.2292 \\ -0.3373 & -0.4674 & 0.4395 & 0.4391 & 0.4592 & 0.0885 \end{bmatrix}$$

$$A^R = \begin{bmatrix} -0.3349 & 0.4583 & 0.4415 & -0.4398 & 0.4781 & -0.0849 \\ -0.3513 & 0.4268 & 0.2218 & 0.0513 & -0.5320 & 0.2238 \\ -0.3611 & 0.3005 & -0.1393 & 0.4958 & -0.2641 & -0.1552 \\ -0.3659 & 0.1081 & -0.4562 & 0.2885 & 0.4656 & -0.2688 \\ -0.3663 & -0.0975 & -0.4834 & -0.2648 & 0.1257 & 0.4962 \\ -0.3621 & -0.2917 & -0.1708 & -0.4807 & -0.3518 & -0.0235 \\ -0.3509 & -0.4355 & 0.2057 & -0.0627 & -0.1163 & -0.6282 \\ -0.3344 & -0.4706 & 0.4629 & 0.4119 & 0.2211 & 0.4527 \end{bmatrix}$$

میانگین ماتریس‌های 6×6 بدست آمده از 61440 بلوک 8×8 در سمت راست شکل (۴) نشان داده شده که پس از اعمال ماتریس کوانتیزاسیون استاندارد استفاده شده در روش *JPEG* به صورت سمت چپ شکل ۴ بدست می‌آید.

$$\begin{bmatrix} 20 & 6 & 1 & 2 & 1 & 0 \\ 5 & 3 & 2 & 1 & 0 & 0 \\ 2 & 2 & 1 & 1 & 0 & 0 \\ 2 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 319 & 68 & 38 & 25 & 17 & 13 \\ 62 & 36 & 24 & 17 & 12 & 9 \\ 34 & 23 & 17 & 13 & 9 & 8 \\ 21 & 16 & 13 & 10 & 8 & 7 \\ 14 & 11 & 9 & 8 & 6 & 5 \\ 10 & 8 & 7 & 6 & 5 & 5 \end{bmatrix}$$

شکل ۴: راست: میانگین ماتریس‌های تبدیل شده و چپ: بعد از استفاده از ماتریس چندی کننده *JPEG*.

همانگونه که در شکل (۴) قابل مشاهده می‌باشد ماتریس تبدیل یافته 6×6 پس از استفاده از ماتریس چندی کننده به مقادیر بسیار کوچکی رسیده که امکان مرتب کردن و خواندن آن به صورت زیگزاگ وجود دارد. میانگین ماتریس‌های حاصل از اعمال تبدیل کسینوسی به بلوک‌های 8×8 از تصاویر، قبل و بعد استفاده از ماتریس چندی کننده نیز در شکل (۵) نشان داده شده است. همانگونه که دیده می‌شود ماتریس‌های نتیجه پس از استفاده از ماتریس چندی کننده مشابه بوده و ماتریس حاصل از ایده‌ی جدید پیشنهادی، ابعاد کوچکتری نیز دارد.

$$\begin{bmatrix} 20 & 6 & 4 & 2 & 1 & 0 & 0 & 0 \\ 5 & 3 & 2 & 1 & 1 & 0 & 0 & 0 \\ 2 & 2 & 1 & 1 & 0 & 0 & 0 & 0 \\ 2 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 318 & 68 & 39 & 25 & 18 & 13 & 9 & 6 \\ 62 & 35 & 25 & 18 & 13 & 9 & 7 & 5 \\ 34 & 24 & 20 & 15 & 11 & 8 & 6 & 5 \\ 21 & 17 & 15 & 12 & 9 & 7 & 5 & 4 \\ 15 & 12 & 11 & 9 & 7 & 6 & 4 & 3 \\ 11 & 9 & 8 & 7 & 6 & 5 & 4 & 3 \\ 7 & 6 & 6 & 5 & 4 & 4 & 3 & 2 \\ 5 & 4 & 4 & 4 & 3 & 3 & 2 & 2 \end{bmatrix}$$

شکل ۵: راست: میانگین ماتریس‌های تبدیل شده و چپ: بعد از استفاده از ماتریس چندی کننده *JPEG*.

60 تصویر از گروه‌های مختلف این پایگاه به‌طور تصادفی انتخاب و به‌عنوان مجموعه یادگیری در نظر گرفته شدند. سپس تمام تصاویر به بلوک‌های 8×8 تقسیم‌بندی شده و با استفاده از آن‌ها، ماتریس‌های تبدیل و عکس تبدیل را محاسبه می‌نماییم. در واقع به این ترتیب، ماتریس‌های تبدیل و عکس تبدیل ثابتی به‌دست آورده و با استفاده از آن‌ها هر تصویر آزمونی را به فضای با ابعاد کمتر نگاشت داده و بازسازی خواهیم کرد که در اینجا به‌عنوان نمونه 8 تصویر نشان داده شده است که در آن 6 بردار مؤلفه‌ی اصلی چپ و راست از ماتریس کوواریانس چپ و راست 8×8 تمامی بلوک‌ها محاسبه شده است. تعداد درایه‌های غیر صفر آنان نیز به صورت $[8, 6, 6, 6, 6, 6]$ در نظر گرفته شده‌اند. برای بردار اول جهت داشتن خطای بازسازی کمتر هیچ عضو صفر در نظر گرفته نشده است.

در ادامه برای حفظ این تصاویر می‌توان از روش *JPEG* استفاده کرد. همانطور که می‌دانیم در روش فشرده‌سازی *JPEG* ابتدا تصاویر به بلوک‌های 8×8 تبدیل گشته و سپس تبدیل کسینوسی گرفته شده و پس از استفاده از ماتریس چندی کننده، اعداد غیر صفر بنا بر جدول‌های استاندارد *JPEG* کد گشته و ارسال می‌گردند. در این روش می‌توان بلوک‌های 6×6 بدست آمده را در حکم ماتریس‌های خروجی تبدیل کسینوسی در نظر گرفت که ابعاد آنان کوچکتر از 8×8 بوده و همچنین از آنجا که توسط ماتریس‌های تبدیل تُنگ بدست آمده‌اند دارای مقادیر کوچکتری می‌باشند. از این پس می‌توان، هر تصویر آزمونی را با استفاده از این ماتریس‌ها نگاشت داده و بازسازی نمود. محاسبه‌ی بردارهای مؤلفه‌ی اصلی توسط تحلیل مؤلفه‌های اصلی دو بعدی تُنگ و برای 60 تصویر، $0/49$ ثانیه زمان می‌برد. البته اگر زمان خواندن تصاویر، بلوک‌بندی آن‌ها و محاسبه‌ی ماتریس کوواریانس 60 تصویر هم لحاظ شود، کل فرآیند 16 ثانیه به طول می‌انجامد. ماتریس‌های A^L ، B^L ، B^R و A^R که در حالت اعمال تحلیل مؤلفه‌های اصلی دو بعدی تُنگ به 61440 بلوک از 60 تصویر 256×256 به شرح زیر می‌باشند. ملاحظه می‌شود که ستون‌های دوم تا ششم ماتریس‌های B^L و B^R دارای 6 درایه‌ی غیر صفر هستند. ستون‌های ماتریس A^L و A^R نیز، بر هم عمود می‌باشند.

$$B^L = \begin{bmatrix} -0.3382 & 0.4552 & 0.4710 & -0.4016 & 0.1813 & -0.4683 \\ -0.3519 & 0.4147 & 0.0954 & 0 & 0 & 0.6552 \\ -0.3603 & 0.3435 & 0 & 0.5378 & -0.3688 & 0 \\ -0.3640 & 0 & -0.5117 & 0.1967 & 0 & -0.4890 \\ -0.3638 & 0 & -0.5291 & -0.1729 & 0.5263 & 0.3228 \\ -0.3601 & -0.3287 & 0 & -0.5635 & -0.2859 & 0.0434 \\ -0.3517 & -0.4209 & 0.0776 & 0 & -0.5003 & -0.0783 \\ -0.3373 & -0.4665 & 0.4704 & 0.4042 & 0.4713 & 0 \end{bmatrix}$$

$$B^R = \begin{bmatrix} -0.3349 & 0.4580 & 0.4792 & -0.4140 & 0.5028 & 0 \\ -0.3513 & 0.4150 & 0.0792 & 0 & -0.6066 & 0.0814 \\ -0.3611 & 0.3387 & 0 & 0.5491 & -0.1766 & -0.0898 \\ -0.3659 & 0 & -0.4796 & 0.2256 & 0.4857 & -0.2556 \\ -0.3663 & 0 & -0.5313 & -0.2012 & 0 & 0.4771 \\ -0.3621 & -0.3153 & 0 & -0.5398 & -0.3096 & 0 \\ -0.3509 & -0.4302 & 0.0415 & 0 & 0 & -0.6786 \\ -0.3344 & -0.4678 & 0.5001 & 0.3799 & 0.1279 & 0.4816 \end{bmatrix}$$



شکل ۶- نتایج بازسازی ۸ تصویر آزمون. ح ز

جدول ۳: معیارهای ارزیابی تصاویر آزمون بازسازی شده توسط مؤلفه‌های اصلی تُنک دوبعدی.

د	ج	ب	الف	
۰,۹۱۵۴	۰,۸۱۶۰	۰,۸۸۳۰	۰,۸۷۱۱	SSIM
۳۳,۸۱	۳۰,۵۴	۳۲,۶۴	۲۸,۷	PSNR
ح	ز	و	ه	
۰,۹۲۳۳	۰,۹۴۶۲	۰,۸۷۲۰	۰,۸۷۲۰	SSIM
۲۶,۱	۳۶,۷۴	۳۲,۱۶	۳۶,۱۵	PSNR

۵- خلاصه و نتیجه‌گیری

تحلیل مؤلفه‌های اصلی، توانایی کاهش ابعاد و حذف افزونگی داده‌ها را دارد؛ تحلیل مؤلفه‌های اصلی دوبعدی نیز، حفظ همبستگی مکانی پیکسل‌ها را به قابلیت‌های تحلیل مؤلفه‌های اصلی اضافه می‌کند. از طرفی، تحلیل مؤلفه‌های اصلی تُنک با ایجاد عناصر صفر در بردارهای ضرایب، پیچیدگی عملیات انتقال داده به فضای جدید را کم کرده و موجب می‌شود درایه‌های نگاشت یافته از نظر مقداری، کوچک شوند. تحلیل مؤلفه‌های اصلی دوبعدی تُنک مجموع مزایای فوق را دارا می‌باشد. با توجه به مجموع خواص شمرده شده، تصمیم گرفته شد عملکرد الگوریتم مذکور، در ابتدا در فشرده‌سازی تصویر ارزیابی گردد. لازم به ذکر است که در این مرحله تمرکز ما بر روی یک تصویر می‌باشد؛ یعنی پایگاه داده را از مجموعه بلوک‌های یک تصویر می‌سازیم.

زمانی که در هیچ کدام از بردارهای پایه‌ی تحلیل مؤلفه‌های اصلی تُنک، درایه‌ی صفر ایجاد نشود؛ عملکردی مشابه تحلیل مؤلفه‌های اصلی استاندارد خواهد داشت. ایجاد نکردن مقادیر صفر در بردارهای پایه‌ی تحلیل مؤلفه‌های اصلی دوبعدی تُنک نیز موجب می‌شود نتایجی مشابه تحلیل مؤلفه‌های اصلی دوبعدی به دست آید. اما نشان دادیم که تُنک کردن بردارهای ضرایب، سبب می‌شود درایه‌های نگاشت یافته از نظر مقداری کوچک‌تر گردند؛ بدون اینکه تغییر زیادی در معیارهای ارزیابی کیفیت رخ دهد. به‌طور کلی ملاحظه شد که اگر بردار ضریب اول را دست نزده و اجازه دهیم مشابه بردار پایه‌ی تحلیل جزء استاندارد باقی بماند؛ تُنک کردن سایر بردارهای ضرایب تا ۴۰ و یا حتی ۵۰ درصد نیز، منجر به

شکل ۶ نتایج بازسازی هشت تصویر آزمون که در میان تصاویر مورد استفاده در یادگیری نبودند را با استفاده از ماتریس‌های A^R ، A^L ، B^R ، B^L بدست آمده از اعمال تحلیل مؤلفه‌های اصلی دوبعدی تُنک روی ۶۰ تصویر یادگیری، نشان می‌دهد. مقادیر $PSNR$ و $SSIM$ برای این تصاویر، در جدول ۳ خلاصه گردیده و نشان می‌دهند بازسازی تصاویر آزمون به‌خوبی صورت گرفته است که این نتیجه‌ی ارزشمندی می‌باشد. در حقیقت دیگر نیازی نیست در فرستنده، هر بار و به ازای هر تصویر، الگوریتم مذکور انجام شده و ماتریس‌های تبدیل و عکس تبدیل محاسبه شوند. بلکه تصویر ورودی به بلوک‌های 8×8 تقسیم شده و با ضرب ماتریسی به شکل $(Y_i)_{k \times k} = (B^L)^T f_i B^R$ ، نگاشت هر بلوک به فضای جدید محاسبه می‌گردد. گیرنده نیز می‌تواند ماتریس‌های A^R و A^L را ذخیره کرده و هر بلوک را با ضرب ماتریسی و به‌صورت $\hat{F}_i = A^L Y_i (A^R)^T$ بازسازی نماید. هر چه تعداد تصاویر یادگیری افزایش یابد، نتایج بهتری حاصل خواهد شد.

نکته‌ی قابل توجه آن است که با وجود آنکه در تصاویر آزمون، تقریباً تمامی گونه‌های تصویر (ساده و هموار، بافت، نقاشی و ..) قرار دارند ولی نتیجه‌ی بازسازی برای تمامی آنان مناسب بوده است.



- مقادیر $PSNR$ و $SSIM$ قابل قبولی برای تصویر بازسازی شده می‌گردد.
- در این مقاله، تبدیل کسینوسی گسسته را به‌عنوان معیاری از $jpeg$ در نظر گرفته و نتایج به‌دست‌آمده از تحلیل مؤلفه‌های اصلی دوبعدی تُنک با آن مقایسه شده است. تمامی بررسی‌ها نیز نشان می‌دهند که روش پیشنهادی، عملکردی نزدیک به تبدیل کسینوسی گسسته داشته و خواص مورد استفاده در $jpeg$ را دارا می‌باشد. از جمله اینکه درایه‌ی سطر و ستون اول از بلوک‌های نگاشت‌یافته، بسیار بزرگ‌تر از سایر درایه‌ها است. با چندی کردن بلوک‌های نگاشت‌یافته، تعداد زیادی از درایه‌های آن‌ها صفر می‌شود. نحوه‌ی اسکن کردن درایه‌های نگاشت‌یافته نیز شبیه $jpeg$ می‌باشد؛ به‌طوری‌که ابتدا مقادیر بزرگ‌تر و مهم‌تر خوانده شوند و بعد از آن‌ها به یک رشته از مقادیر صفر رسیده و پایان بلوک اعلام می‌شود.
- از همه مهم‌تر اینکه، توانستیم با استفاده از ۶۰ تصویر یادگیری که جمعاً ۶۱۴۴۰ بلوک با ابعاد 8×8 در اختیار ما قرار می‌دهند؛ ماتریس‌های تبدیل و عکس تبدیل ثابتی برای تحلیل مؤلفه‌های اصلی دوبعدی تُنک به‌دست آوریم که توسط آن‌ها، هر تصویر آزمونی به‌خوبی بازسازی شده و نیازی نیست هر بار در فرستنده و گیرنده تبدیل‌های فوق انجام شده و بردارهای پایه محاسبه شوند.
- ### مراجع
- [1] S. V. B. Jardim, "Sparse and robust signal reconstruction," *Theory and Applications of Mathematics & Computer Science*, vol. 5, no. 1, pp. 1-19, 2015.
 - [2] M. Elad, M. A. T. Figueiredo, and Y. Ma, "On the role of sparse and redundant representations in image processing," *Proc. IEEE*, vol. 98, no. 6, pp. 972-982, Jun. 2010.
 - [3] I. T. Jolliffe, "Principal Component Analysis," 2th ed. Springer, Verlag New York, ch.1, pp. 1-6, 2002.
 - [4] N. O'Rourke and L. Hatcher, "Principal component analysis," in *A Step-by-Step Approach to Using SAS® for Factor Analysis and Structural Equation Modeling*, 2th ed. SAS institute, Mar. 2013.
 - [5] J. Hou, L.P. Chau, N. Magnenat-Thalmann and Y. he, "SLRMA: Sparse low Rank Matrix Approximation for data Compression," *IEEE Trans. Circuits and Systems for Video technology*, vol. PP, no. 99, Doi: 10.1109/TCSVT.2015.2513698, 2016.08.26
 - [6] O. Gokhan, O.G. Guleryuz and Y. Altunbasak, "Approximation and Compression with Sparse orthonormal Transforms," *IEEE Trans. Image Processing*, vol. 24, no. 8, pp. 2328-2343, 2015.
 - [7] J. Yang and J.Y. Yang, "From image vector to matrix: A straightforward image projection technique-IMPCA vs. PCA," *pattern recognition society*, vol. 35, issue. 9, pp. 1997-1999, 2002.
 - [8] J. Yang, D. Zhang, A. F. Farangi, and J. Y. Yang, "Two-dimensional PCA: A new approach to appearance-based face representation and recognition," *IEEE Trans on Pattern Analysis and Machine Intelligence*, vol. 26, no. 1, pp. 131-137, Jan. 2004.
 - [9] J. Ye, R. Janardan, and Q. Li, "GPCA: An efficient dimension reduction scheme for image compression and retrieval," *Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Seattle, Washington, USA, Aug. 2004.
 - [10] D. Zhang and Z. H. Zhou, "(2D)PCA: Two-directional two-dimensional PCA for efficient face representation and recognition," *Neurocomputing* 69, vol. 69, issue. 1-3, pp. 224-231, Dec. 2005.
 - [11] J. Yang and C. Liu, "Horizontal and vertical 2dPCA-based discriminant analysis for face verification on a large-scale database," *IEEE Trans on Information Forensics and Security*, vol. 2, no. 4, pp. 781-792, Dec. 2007.
 - [12] W. Zou, D. Zhang, and K. Wang, "Bidirectional PCA with assembled matrix distance metric for image recognition," *IEEE Trans on Systems, Man, and Cybernetics- Part B: Cybernetics*, vol. 36, no. 4, pp. 863-872, Dec. 2006.
 - [13] H. Shen and J. Z. Huang, "Sparse principal component analysis via regularized low rank matrix approximation," *Multivariate Analysis* 99, vol. 99, issue. 6, pp. 1015-1034, Jul. 2008.
 - [14] H. Zou, T. Hastie, and R. Tibshirani, "Sparse principal component analysis," *Computational and Graphical Statistics*, vol. 15, no. 2, pp. 265-286, 2006.
 - [15] Z. Lu and Y. Zhang, "An augmented Lagrangian approach for sparse principal component analysis," *Mathematical Programming*, vol. 135, issue. 1, pp. 149-193, Apr. 2011.
 - [16] M. Jornee, Y. Nesterov, P. Richtarik, and R. Sepulchre, "Generalized power method for sparse principal component analysis," *Machine Learning Research* 11, vol. 11, pp. 517-553, Jan. 2010.
 - [17] Z. Lai, Y. Xu, Q. Chen, J. Yang, and D. Zhang, "Multilinear sparse principal component analysis," *IEEE Trans on Neural Networks and Learning Systems*, vol. 25, no. 10, pp. 1942-1950, Oct. 2014.
 - [18] T. D. X. Duong and H. V. Nguyen, "Some extension of sparse principal component analysis," *Machine Learning and Computing*, vol. 2, no. 5, pp. 701-705, Oct. 2012.
 - [19] I .T .Jolliffe, N. T. Trendafilov, and M. Uddin, "A modified principal component technique based on the LASSO," *Computational and Graphical Statistics*, vol. 12, issue. 3, pp. 531-547, 2003.
 - [20] A. Majumdar, "Image compression by sparse PCA coding in curvelet domain," *Signal, Image and Video Processing*, vol. 3, issue.1, pp. 27-34, Feb. 2009.
 - [21] T. Hastie, R. Tibshirani, and J. Friedman, "Linear methods for regression," in *the elements of statistical*

- learning, data mining, inference, and prediction*, 2th ed., Springer-Verlag, 2009, ch. 3, pp. 44-77.
- [22] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Royal Statistical Society, Series B: Statistical Methodology*, vol. 67, issue. 2, pp. 301-320, 2005.
- [23] K. Sjostrand, M. B. Stegmann, and R. Larsen, "Sparse Principal Component Analysis in Medical Shape Modeling", *International Symposium on Medical Imaging*, San Diego, CA, USA, vol. 6144, Feb. 2006.
- [24] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *The Annals of Statistics*, vol. 32, no. 2, pp. 407-499, 2004.
- [25] K. Sjostrand, L. H. Clemmensen, R. Larsen, and B. Ersboll, "SpaSM: A matlab toolbox for sparse statistical modeling," *Statistical Software*, vol. 10, issue. 2, 2012, Software available at <http://www2.imm.dtu.dk/projects/spasm/>
- [26] Z. Wang and A. C. Bovik, "A universal image quality index," *IEEE Signal Processing Letters*, vol. 9, no. 3, pp. 81-84, march 2002.
- [27] A. G. Weber, "The USC-SIPI Image Database: Version 5," Signal and Image Processing Institute, University of Southern California, Department of Electrical Engineering, 2006.



زهرا یک کلام مقطع کارشناسی را در رشته مهندسی برق گرایش مخابرات، در دانشگاه شهید بهشتی گذرانده و سپس سال ۱۳۹۲ در دانشگاه شهید بهشتی و با استفاده از سهمیه استعدادهای درخشان در مقطع کارشناسی ارشد، رشته مخابرات گرایش سیستم مشغول به تحصیل شد. ایشان در سال ۱۳۹۴ با کسب رتبه اول در گرایش خود، فارغ التحصیل شده است. زمینه های پژوهشی مورد علاقه ی ایشان، تشخیص الگو، داده کاوی و پردازش تصویر است. ایشان در حال حاضر در دانشگاه غیرانتفاعی تابناک استان فارس مشغول تدریس می باشند.



فرح ترکمنی آذر مدرک کارشناسی و کارشناسی ارشد خود را به ترتیب در رشته مهندسی برق گرایش الکترونیک از دانشگاه صنعتی امیر کبیر و گرایش مخابرات از دانشگاه صنعتی اصفهان دریافت نمود. سپس ایشان دکتری خود را از دانشگاه New South wales استرالیا در زمینه پردازش تصویر با استفاده از شبکه های عصبی در سال ۱۳۷۴ اخذ کرد. از سال ۱۳۷۴ در دانشگاه صنعتی اصفهان و از سال ۱۳۸۰ در دانشگاه شهید بهشتی به عنوان عضو هیئت علمی مشغول به فعالیت می باشند. علاقمندی های علمی ایشان پردازش سیگنال ها و تصاویر دیجیتال است.