
**دانشکده مهندسی برق و رباتیک**

**گروه الکترونیک**

**گرایش الکترونیک**

**پایان نامه کارشناسی ارشد**

# احراز هویت با استفاده از چهره در مجموعه بزرگی از تصاویر افراد

**حسین سعیدی**

**استاد راهنما**

**دکتر حسین خسروی**

**استاد مشاور**

**دکتر وحید ابوالقاسمی**

**شهریور ۱۳۹۴**

# تقدیم به همسرم

# سپاس گزاری...

بر خود لازم می دانم از استاد گرامی دکتر حسین خسروی که علاوه بر علم از او درس ادب و اخلاق هم آموختم قدردانی کنم، همچنین شایسته است از استاد مشاور محترم دکتر وحید ابوالقاسمی نیز نهایت تشکر را داشته باشم.

# تعهد نامه

اینجانب **حسین سعیدی** دانشجوی دوره کارشناسی ارشد رشته **مهندسی برق – الکترونیک** دانشکده **مهندسی برق و رباتیک** دانشگاه شاهرود نویسنده پایان نامه **احراز هویت با استفاده از چهره در مجموعه بزرگی از تصاویر افراد** تحت راهنمائی **دکتر حسین خسروی** متعهد می شوم:

- تحقیقات در این پایان نامه توسط اینجانب انجام شده است و از صحت و اصالت برخوردار است.
- در استفاده از نتایج پژوهشهای محققان دیگر به مرجع مورد استفاده استناد شده است.
- مطالب مندرج در پایان نامه تاکنون توسط خود یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ جا ارائه نشده است.
- کلیه حقوق معنوی این اثر متعلق به دانشگاه شاهرود می باشد و مقالات مستخرج با نام « دانشگاه شاهرود » و یا « Shahrood University» به چاپ خواهد رسید.
- حقوق معنوی تمام افرادی که در به دست آمدن نتایج اصلی پایان نامه تأثیرگذار بوده اند در مقالات مستخرج از پایان نامه رعایت می گردد.
- در کلیه مراحل انجام این پایان نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته یا استفاده شده است اصل رازداری ، ضوابط و اصول اخلاق انسانی رعایت شده است.

.

## تاریخ و امضای دانشجو

**مالکی‌ت نتای‌ج و حق نشر**

- **کلی‌ه حقوق معنوی ای‌ن اثر و محصولات آن (مقالات مستخرج، کتاب، برنامه های رای‌انه ای، نرم افزار ها و تجهی‌زات ساخته شده است ) متعلق به دانشگاه صنعتی شاهرود می باشد. ای‌ن مطلب بای‌د به نحو مقتضی در تولی‌دات علمی مربوطه ذکر شود.**
- **استفاده از اطلاعات و نتای‌ج موجود در پای‌ان نامه بدون ذکر مرجع مجاز نمی باشد.**

## چکیده

با افزایش روز افزون تصاویر رقومی بر روی سیستم‌های کاربران و همچنین افزایش تعداد و حجم تصاویر بر روی شبکه، مسأله طبقه بندی تصاویر بیش از پیش مورد توجه قرار گرفته است؛ بر همین اساس ویژگی‌ها و دسته‌بندهای متنوعی در سال های اخیر برای حل مساله تشخیص چهره ارائه شده است. ترکیب ویژگی‌های متنوع با طبقه‌بند مبتنی بر شبکه عصبی یکی از پیشنهاداتی است که در این پایان نامه ارائه می شود. طبقه‌بندی مبتنی بر نمایش تنک روش دیگری است که در اینجا بدان خواهیم پرداخت و باتوجه به نو بودن این روش، در این پایان نامه بیشتر روی این روش تمرکز داریم. در این روش از بین تعداد زیادی سیگنال پایه که در حالت کلی تعدادشان خیلی بیشتر از بعدشان است، کمترین تعداد را برای نمایش یک سیگنال انتخاب می کنیم. عمل انتخاب بهترین و در عین حال کمترین تعداد اتم برای نمایش سیگنال مورد نظر، در حالت کلی بسیار دشوار است؛ اما محققین در سال های اخیر با ارائه‌ی الگوریتم‌های بهینه سرعت و دقت این روش را بالا برده‌اند. به این ترتیب این روش به سرعت در کاربردهای گوناگون پردازش سیگنال مورد استفاده قرار گرفت. در این پایان نامه علاوه بر کار روی پایگاه‌های معمول، روی پایگاه‌های داده با تعداد تصویر زیاد نیز کار شده است که این مهم در تعداد انگشت شماری از کارهای مشابه دیده می شود. نتایج شبیه‌سازی‌ها نشان‌دهنده‌ی بهبود نتایج است. به عنوان نمونه یکی از روش‌هایی که در این پایان‌نامه معرفی شده است روی پایگاه داده ORL نرخ تشخیص ٪ ۹۸/۱۳ را دارد که در مقایسه با کارهای گذشته بهبود یافته است.



**کلمات کلیدی :** احراز هویت، تشخیص چهره، نمایش تنک، آموزش واژه نامه، کدینگ سیگنال

# فهرست مطالب

# فهرست شکل‌ها

# فهرست جدول‌ها

# فصل اول :

# مقدمه

## ۱-۱- مقدمه

در ابتدای این فصل مساله طبقه بندی تصاویر را تعریف می‌کنیم. در ادامه علل اصلی استفاده از نمایش تنک را ذکر می‌کنیم. همچنین ایده اصلی نمایش تنک سیگنال‌ها را به شکل مختصر توضیح می‌دهیم. در انتها نیز چالش‌های اصلی در مساله طبقه-بندی تصاویر بیان می‌شود.

## ۱-۲- تعریف مساله طبقه بندی تصاویر

حجم بالای تصاویر رقومی بر روی شبکه اینترنت و پدید آمدن شبکه های اجتماعی و نیازهای کاربران این شبکه‌ها از یک سو و همچنین کاربردهای تصویر رقومی در مسائل امنیتی از سوی دیگر، مساله طبقه‌بندی تصویر را تبدیل به یکی از نیازهای اساسی جامعه امروزی کرده است. ورودی این مساله، تعدادی تصویر به عنوان داده آموزشی و تعدادی تصویر به عنوان داده آزمون است. طبقه بندی داده‌های آموزش از پیش مشخص است و طبقه بندی هر کدام از تصاویر داده‌های آموزشی با برچسب معین شده است. بر چسب داده‌ها همان طبقه‌ای است که داده‌ها به آن تعلق دارند. هدف سیستم طبقه‌بندی تصاویر آن است که پس از ورود یک تصویر از مجموعه‌ی آزمون، برچسب آن را مشخص کند.

به عبارت دیگر، فرض کنید $X$ مجموعه‌ی تصاویر آموزش بوده و تعداد این مجموعه $N$ باشد. مجموعه دیگری نیز با همین تعداد اعضا به عنوان برچسب‌های تصاویر در اختیار داریم. اگر $Y=\{y_1, y_2, \dots, y_N\}$ مجموعه برچسب‌ها باشد و در کل داده‌ها به C دسته مختلف تعلق داشته باشند، بنابراین $y_i \in \{1,2, \dots C\}$ است. حال اگر تصویری از مجموعه‌ی آزمون را به سیستم طبقه‌بندی تصویر بدهیم، خروجی سیستم باید برچسبی باشد که متعلق به مجموعه $\{1,2,\dots,C\}$ است. سیستمی ایده آل است که بتواند با بیشترین دقت برچسب داده‌های آزمون را پیش بینی کند.

بطور کلی روش‌های شناسایی چهره را می‌توان به دو دسته تقسیم کرد. دسته‌ی اول روش‌هایی هستند که مشخصه‌های کلی چهره را در نظر می‌گیرند و به روش های مبتنی بر ظاهر معروف‌اند؛ و دسته‌ی دوم روش‌هایی هستند که خصوصیات قسمت‌های مهم چهره را مدنظر قرار می‌دهند و به روش‌های مبتنی بر مدل معروف‌اند.

از روش‌های دسته اول می توان به چهره ویژه[1] که توسط تورک و دیگران پیشنهاد شد، اشاره کرد که از روش آنالیز المان‌های اصلی[2] استفاده می‌کند[۱]. روش چهره فیشر[3]، که از تحلیل جداساز خطی[4] استفاده می‌کند[۲] و روش‌های شبکه‌ی عصبی[۳] برخی دیگر از روش‌هایی هستند که از تمام چهره برای شناسایی استفاده می‌کنند. از روش‌های دسته دوم می‌توان به روش‌های تطابق گراف دسته‌ای الاستیک [۴] و مدل ظاهر فعال[۵] اشاره کرد.

### ۱-۳- چالش‌های اصلی در طبقه‌بندی تصاویر

تصاویر دوبعدی که حاوی اطلاعات شدت روشنایی چهره می باشند به عنوان اولین نوع داده‌هایی هستند که برای شناسایی چهره مورد استفاده قرار گرفته‌اند. در استفاده از این نوع داده‌ها برای شناسایی، دو مسئله‌ی اساسی وجود دارد. نکته ی اول این است که این نوع داده‌ها با استفاده از نور بازگشتی از چهره جمع آوری می‌شوند، لذا به نور محیط و جهت نور تابیده شده به چهره وابستگی زیادی دارند. به این مسئله تغییر شدت روشنایی[5] می گویند. مسئله ی دوم تغییر زاویه‌ی چهره است؛ چهره ی انسان وقتی تحت زوایای مختلف تصویر برداری شود، بدلیل حالت سه بعدی اش، به تصاویر متفاوتی منجر خواهد شد. به این مسئله، تغییر زاویه‌ی چهره[6] می گویند.

---

[1] Eigenface

[۲] ۱-۱-۱ Principal Component Analysis (PCA)

[3] FisherFace

[4] Linear Discriminative Analysis (LDA)

[5] Illumination Variant

[6] Pose Variation

## ۱-۴- چرا از نمایش تنک[7] استفاده می کنیم؟

تصاویر طبیعی که توسط دوربین‌های دیجیتال امروزی گرفته می‌شود، معمولا وضوحی در حدود ۱۰ میلیون پیکسل دارد. هر تصویر در یک بردار با ابعاد بالا ذخیره می شود. برای مثال تصویر $x= \{x_1, x_2, \dots x_n\}^T \in R^n$ را در نظر بگیرید. هر تصویر در فضایی با $n= 10^7$ بعد نمایش داده می شود ولی مجموعه تمام تصاویر طبیعی این فضا را پوشش نمی دهند.

اطلاعات موجود در نمایش تصویر به عنوان برداری از پیکسل‌ها دارای ساختارهای تکراری زیادی است. این ساختارهای تکراری دو علت عمده دارد. اول اینکه دنیای بیرونی که تصاویر از آن گرفته شده دارای ساختارهای تکراری است و دوما سیستم‌های تصویر برداری با وضوح بالایی کار می‌کنند که موجب می شود بسیاری اطلاعات حاوی این ساختارهای مکرر در داده‌ی مربوط به تصویر گنجانده شود. در واقع وضوح بالای تصاویر دیجیتال باعث می شود در قسمت هایی از تصویر که جزئیات کمتر است اطلاعات اضافی داشته باشیم. برای مثال تصویر ۱-۱ که نمونه ای از یک تصویر طبیعی است را در نظر بگیرید. همانطور که مشخص است، تصویر از سه قسمت کاملا مجزا شامل آسمان، درختان و ساحل تشکیل شده است. هر کدام از این قسمت‌ها سرشار از ساختارهای تکراری است. این اطلاعات تکراری سبب عملکرد ضعیف طبقه‌بندی خواهد شد. در مساله طبقه بندی تصویر هر چه نمایش ما از داده ها به مفهوم نزدیک‌تر باشد طبقه بندی با دقت بیشتری انجام خواهد شد. لذا تلاش های بسیاری برای مدل کردن فضایی که تصاویر طبیعی در آن قرار می‌گیرند انجام شده است.

به عنوان مثال در روش تحلیل مولفه های اصلی، داده ها بر روی پایه هایی تصویر می شوند که بیشترین واریانس را در فضای تبدیل یافته دارد. ضعف عمده این روش ثابت بودن پایه های تبدیل است.

[7] Sparse Representation

شکل ۱-۱- تصاویر طبیعت شامل ساختارهای تکراری فراوانی اند.

در مدل های پیچیده‌تر تعداد پایه ها از بعد فضا بیشتر است که باعث می‌شود بتوانیم روابط پیچیده و غیر خطی را مدل کنیم. روش‌های زیادی برای تخمین فضای واقعی تصاویر طبیعی ارایه شده است. از آنجایی که به دنبال مدل کردن سیگنال ها در فضایی با بعد کمتر هستیم عموما به این روش ها مدل های کم بعد سیگنال[8] گفته می شود[۶].

در برخی کاربردهای پردازش سیگنال، پایه هایی که از آن برای نمایش در فضای جدید استفاده می شود، مشخص است و نیازی به ارایه روش یا الگوریتم برای به دست آوردن پایه ها نیست. اما در مساله طبقه‌بندی تصاویر، مجموعه خاصی به ما داده شده است و برای نمایش تنک این تصاویر باید پایه‌هایی را به دست آوریم که برای نمایش تنک تصاویر مجموعه ما مفید باشد. در ادامه روش پایه که برای نمایش سیگنال ها به شکل تنک استفاده می شود را معرفی می کنیم.

[8] Low Dimensional Signal Models

## ۱-۵- نمایش تنک

آدمی همواره به دنبال ساده‌تر کردن امور خود است. از این رو علاقه دارد که امور مرکب و پیچیده را تا جای ممکن تجزیه کند. اما می‌بایست این امر را به نحو درستی انجام داد. یکی از اصلی‌ترین چالش‌های جامعه‌ی پردازش سیگنال هم، تجزیه‌ی سیگنال‌ها به عناصر سازنده‌ی آنها بوده است. تبدیل‌هایی مانند تبدیل فوریه، تبدیل موجک و ... از این رو بوجود آمدند.

اگرچه تبدیل‌هایی مانند تبدیل فوریه بدلیل داشتن جوابی یکتا و همچنین سادگی محاسبات همواره مورد توجه بوده‌اند، اما به این خاطر که تنها برای توصیف طیف خاصی از سیگنال‌ها "مناسب" اند، نمی‌توان تنها بدین تبدیل‌ها بسنده کرد. این تبدیل‌ها را اصطلاحاً "کامل[۹] " می‌خوانند. تبدیل‌های کامل، تبدیل-هایی هستند که در آنها تعداد پایه‌ها با بعد سیگنال برابر است. واژه "مناسب" را برای تبدیل‌هایی بکار می‌بریم که نمایش ساده‌ای از سیگنال ارائه دهند.

تبدیل فوریه و دیگر تبدیل‌های این خانواده، تنها برای نمایش سیگنال‌های سینوسی مناسب‌اند و اگر بخواهیم سیگنال‌هایی غیر از سیگنال‌های متناوب را توسط این تبدیل‌های کلاسیک نشان دهیم، باید پایه‌های زیادی را برای نشان دادن سیگنال بکار بریم که در این صورت دیگر نمایش مناسبی از سیگنال نخواهیم داشت. برای غلبه بر این مشکل تبدیل‌های "فرا کامل[۱۰] " معرفی شدند. در این تبدیل‌ها، تعداد پایه‌ها از بعد سیگنال بسیار بیشتر است.

در تبدیل‌های فرا کامل تعداد پایه‌های تبدیل را افزایش می‌دهیم به این امید که بتوانیم نمایش تنکی از طیف وسیع‌تری از سیگنال‌ها داشته باشیم. به عبارت دیگر تعداد پایه‌ها را افزایش می‌دهیم تا با تعداد کمتری از آنها بتوانیم سیگنال‌های

[9] Complete
[10] Over Complete

بیشتری را نشان دهیم. بگذارید بحث را به صورت ریاضی ادامه دهیم. معادله‌ی (۱-۱) را در نظر بگیرید.

$$y = Dx \qquad (۱-۱)$$

در معادله‌ی (۱-۱)، $y$ سیگنال مورد نظر، $D$ تبدیل یا نگاشت و $x$ ضرایب نمایش سیگنال است. می‌خواهیم سیگنال $y$ را توسط تبدیل $D$ طوری نشان دهیم که بردار ضرایب $x$ حداکثر صفر را داشته باشد. هر چقدر بردار ضرایب، صفر بیشتری داشته باشد، نمایش ساده‌تری از سیگنال ارائه داده‌ایم.

با توجه به معادله‌ی (۱-۱) و توضیحات داده شده هدف ما بدست آوردن ماتریس $D$ و بردار ضرایب $x$ است. اگر تبدیل مورد استفاده‌ی ما کامل باشد یا به عبارت دیگر ماتریس $D$ رتبه کامل[11] باشد، $x$ یکتا است، و به راحتی طبق معادله‌ی (۱-۲) بدست خواهد آمد.

$$x = D^{-1}y \qquad (۱-۲)$$

اما در این صورت تقریباً برای هیچ کدام از سیگنال‌های طبیعی نمایش ساده یا تنکی ارائه نداده‌ایم. روش نمایش تنک می‌گوید اگر تعداد ستون‌های ماتریس $D$ را زیاد کنیم با لحاظ کردن شرایطی می‌توان نمایش تنکی از بسیاری از سیگنال‌ها نشان داد. اما مشکلی که باید حل شود این است که در این صورت تعداد مجهولات از تعداد معادلات بیشتر است بنابراین معادله یا بیشمار جواب دارد یا جواب ندارد. اگر فرض کنیم تبدیل فرا کامل $D$، رتبه کامل باشد، بنابراین معادله بیشمار جواب دارد[۷][۸].

[11] Full Rank

حال با مشخص شدن موضوع بحث به سراغ ادبیات بکار رفته در نمایش تنک و چالش‌های فراروی ما در این حوزه خواهیم رفت.

### ۱-۶- ادبیات نمایش تنک و مسائل فرارو

در نمایش تنک، تبدیل $D$ را "واژه‌نامه[۱۲] " می‌گوییم و هر پایه از آن را یک "اتم[۱۳] " می‌خوانیم. اما همان طور که در بخش قبل اشاره شد، یافتن بردار ضرایب $x$ از طرفی و پیدا کردن ماتریس تبدیل $D$ از سوی دیگر دو مسئله‌ای است که در نمایش تنک پیش روی ماست. گفتیم با فرض اینکه واژه‌نامه رتبه کامل باشد، برای معادله (۱-۱) بیشمار جواب وجود دارد. اولین مساله‌ی بحث نمایش تنک انتخاب بهترین جواب ممکن است. بهترین جواب ممکن تنک‌ترین جواب ممکن است[۹].

مساله‌ی بعدی انتخاب واژه‌نامه‌ی مناسب است. هر چه واژه‌نامه ویژگی‌های بیشتر و بهتری از کلاس سیگنال‌های مورد بررسی را در خود جای داده باشد، مناسب‌تر است و مسلماً واژه‌نامه‌ای مناسب‌تر است که تنک کننده‌تر باشد. به طور کلی دو راه برای این انتخاب واژه‌نامه تنک کننده وجود دارد. راه اول استفاده از واژه‌نامه‌های ثابت و یا غیر وفقی[۱۴] است و راه دوم استفاده از واژه‌نامه‌های وفقی[۱۵] است[۱۰].

### ۱-۷- چالش های اصلی در طبقه بندی تصاویر با استفاده از نمایش تنک

هر تصویر در مجموعه‌ی داده، توسط یک بردار با ابعاد بسیار بالا ذخیره می شود. اگر بخواهیم این بردار را توسط یک واژه‌نامه به شکل تنک نمایش دهیم، عملا غیر ممکن خواهد بود، چرا که اندازه واژه‌نامه بسیار

[12] Dictionary
[13] Atom
[14] Non-Adaptive Dictionary
[15] Adaptive Dictionary

بزرگ خواهد بود و اعمال فرض تنک بودن نیز برای واژه‌نامه‌ای با این اندازه امکان پذیر نیست. روشی که برای حل این مشکل در [۱۱] در نظر گرفته شده، استفاده از تکه هایی از تصویر به جای کل تصویر است. یکی از مسائل مهم دیگر در حوزه نمایش تنک، زمان اجرای الگوریتم هاست. روشی که ارائه می شود باید از لحاظ تئوری و در عمل دقت مناسبی داشته باشد و نیز برای کاهش پیچیدگی زمانی راهکاری ارائه دهد.

### ۱-۸-جمع بندی

در این فصل مساله شناسایی چهره را تعریف کردیم و چالش‌های پیش رو در این زمینه را نیز برشمردیم. سپس مختصری از روش نمایش تنک و کارآیی آن در طبقه بندی تصاویر ارائه شد. در ادامه و در فصل دوم ابتدا تعدادی از الگوریتم‌هایی که تاکنون برای شناسایی چهره معرفی شده است را از نظر می‌گذرانیم؛ سپس مباحث پایه‌ای بحث نمایش تنک را به همراه تعدادی از الگوریتم‌های مطرح در زمینه‌ی بازیابی نمایش تنک سیگنال ها را به اختصار مرور می کنیم. و در انتهای فصل دوم آموزش واژه‌نامه را مطرح خواهیم کرد. در فصل سوم الگوریتم‌های پیشنهادی را مطرح می‌کنیم. در این فصل، مفهوم نمایش تنک را برای طبقه‌بندی تصویر به خدمت می‌گیریم؛ همچنین با ترکیب ویژگی‌های متنوع به همراه طبقه‌بند مبتنی بر شبکه عصبی الگوریتمی دیگر نیز مطرح خواهیم کرد. نهایتا، در فصل چهارم نتایج پیاده سازی مطالب ارائه شده را مطرح کرده و پیشنهاداتی برای کارهای آینده ارائه خواهیم کرد.

# فصل دوم :

# بررسی پژوهش‌های پیشین

## ۲-۱- مقدمه

تمرکز ما در این فصل بر مرور روش‌های تشخیص چهره تا به امروز است. ابتدا روش‌های شناسایی چهره مبتنی بر ظاهر را بررسی می‌کنیم و چند نمونه از این دسته را مورد بررسی قرار می‌دهیم، سپس به روش-های مبتنی بر مدل می‌پردازیم و نمونه‌هایی از این دسته را بررسی می‌کنیم. آنگاه به اصول تئوریک نمایش تنک به عنوان روشی نو در تشخیص چهره خواهیم پرداخت. بحث‌مان را با معرفی تابع نُرم و تابع شبه نُرم آغاز کرده و آن را با معرفی تعدادی از الگوریتم‌های بدست آوردن نمایش تنک سیگنال پی می-گیریم. در نهایت مساله‌ی آموزش واژه‌نامه را بررسی خواهیم کرد و بحث را با معرفی چند الگوریتم معروف در زمینه‌ی آموزش واژه‌نامه به اتمام می‌رسانیم.

## ۲-۲- شناسایی چهره بر مبنای ظاهر:

در روش‌های شناسایی بر مبنای ظاهر[۱۶]، تصویر دو بعدی به یک بردار $n$ بعدی تبدیل می‌شود، به عبارت دیگر، هر تصویر با یک نقطه در فضای $n$ بعدی بیان می‌شود. بدلیل ابعاد زیاد تصاویر انجام محاسبات بسیار زمان‌بر خواهد بود. لذا اولین قدم در این روش‌ها استفاده از تکنیک‌های آماری برای کاهش ابعاد بردارهای ویژگی است، تا به کمک آنها بتوان توزیع داده‌ها را بدست آورده و سپس بدون از دست دادن اطلاعات اساسی، تصاویر را در یک زیر فضای با ابعاد کمتر ارائه نمود. در این حالت با ورود هر تصویر جدید برای شناسایی، آن را به زیر فضای پایین‌تر برده و شباهت تصویر ورودی با تصاویر داده شده در این زیر فضا بررسی می‌شود. بدین ترتیب، هر تصویر که متشکل از داده‌های دو بعدی است را می‌توان به صورت یک بردار $n$ بعدی از داده بیان کرد. برای این کار سطرها یا ستون‌های داده را پشت سر هم قرار می‌دهند. مثلا یک تصویر دو بعدی با ابعاد $m*n$ را می‌توان بصورت یک بردار درفضای $X \in R^{m*n}$ بیان

[16] Appearance-based face recognition

نمود. این فضای با ابعاد بالا را می‌توان به کمک تکنیک‌های آماری در زیر فضای با ابعاد کمتر نیز بیان کرد. برای این کار می‌توان از زیر فضاهای خطی یا غیر خطی استفاده کرد. در این روش‌ها، ابتدا داده‌ها را به یک بردار $n$ بعدی تبدیل کرده و سپس در یک ماتریس قرار می‌دهند. برای مثال، $X = (x_1, x_2, ....., x_c)$ یک ماتریس $(m*n)*c$ است که هر $x_i$ یک تصویر است و $c$ تعداد کل تصاویر است[۱۲].

## ۲-۲-۱- تحلیل زیر فضاهای خطی

در ادامه دو زیر فضا که برای کاهش بعد و طبقه بندی داده‌ها استفاده می‌شوند را بررسی می‌کنیم. این دو زیر فضا، زیرفضاهای بدست آمده از روش‌های PCA و LDA هستند. هر کدام از این دو روش با توجه به دیدگاه آماری مربوط به خود، یک زیر فضا تعریف کرده و داده‌ها را به آن زیرفضا منتقل می‌کند. برای مقایسه داده‌ها کافیست شباهت بردارهای انتقال یافته در زیر فضای مورد نظر با داده‌ی ورودی که به آن زیرفضا منتقل شده است را بررسی کنیم. هر دوی این زیر فضاها را بدلیل خطی بودن میتوان به صورت $Y = W^T \times X$ بیان کرد که X ماتریس داده‌ها در فضای اولیه است، $W_{(n*m)*d}$ ماتریس انتقال و $Y_{d*c}$ بیان تصاویر در زیرفضای مورد نظر است و در اغلب موارد $d << n*m$ است.

**۲-۲-۱-۱- روش چهره ویژه :**

روش چهره ویژه در سال ۱۹۹۱ توسط تورک و همکارانش پیشنهاد شد[۱][۱۳]. این روش از تحلیل المان‌های اصلی برای کاهش بعد استفاده می‌کند تا بتواند زیرفضایی با بردارهای متعامد پیدا کند که در آن زیرفضا، پراکندگی داده‌ها را به بهترین حالت نشان دهد. پس از مشخص شدن بردارها تمامی تصاویر به این زیر فضا منتقل می‌شوند تا وزن‌هایی که بیانگر تصویر در آن زیرفضا هستند بدست آیند. با مقایسه شباهت وزن‌های موجود با وزن تصویر جدیدی که به این زیر فضا منتقل شده می‌توان تصویر ورودی را شناسایی کرد. بردارهای پایه در روش PCA همان بردارهای ویژه‌ی ماتریس X هستند که از ماتریس کوواریانس بدست می‌آیند. ماتریس کوواریانس به صورت معادله زیر بدست می آید:

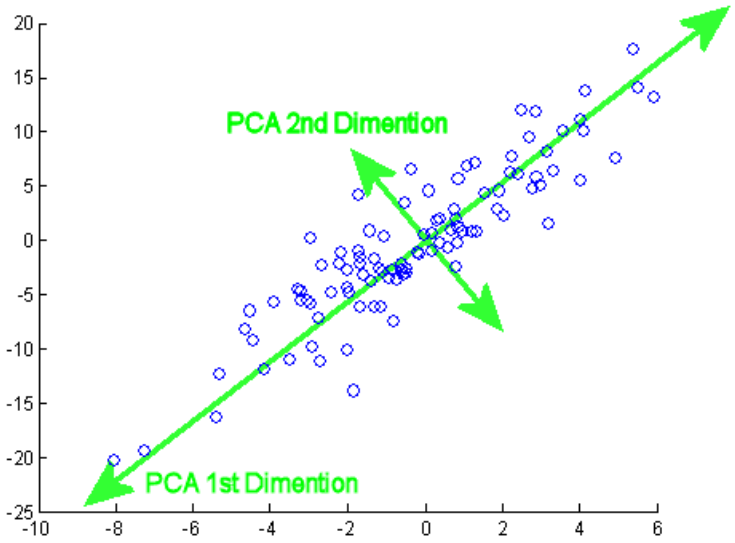
$$COV(X,Y) = \frac{\sum_{i=1}^{N}(X_i - \overline{X})(Y_i - \overline{Y})}{n-1} \qquad (۱-۲)$$

پس از پیدا کردن ماتریس بردارهای ویژه می‌توان از همه‌ی المان‌ها یا d المان آن برای انتقال استفاده کرد. نمونه‌ای از اعمال این تبدیل بر داده‌های دو بعدی در شکل ۱-۲ آمده است. در این زیر فضا، بردار متناسب با بزرگترین مقدار ویژه، حاوی اطلاعات بیشترین تغییرات (پراکندگی) و بردار متناسب با کوچکترین مقدار ویژه، حاوی کمترین تغییرات (پراکندگی) است. در ادامه و برای روشن شدن بحث، الگوریتم محاسبه‌ی این بردارها را بیان می‌کنیم:

الف- ابتدا داده‌های هر تصویر در یک ستون قرار داده می‌شوند $X_i = [x_i^1 .... x_i^{m*n}]^T$.

ابعاد این ماتریس $1*(m*n)$ است که $m,n$ ابعاد تصویر هستند.

ب- ماتریس $X$ را تشکیل داده بطوری که هر ستون آن بیانگر داده های یک تصویر باشد $X = (X^1, X^2, ....., X^c)$ . ابعاد این ماتریس $(m*n)*c$ است که $c$ تعداد تصاویر است.




شکل ۲-۱ : مولفه های اصلی نقاط دو بعدی. همان طور که در تصویر مشخص است داده‌ها در جهت اولین بردار ویژه، پراکندگی بیشتری نسبت به دومین بردار ویژه دارند.

ج- تصاویر حول مرکز منتقل می‌شوند. برای این کار، داده ها از داده میانگین کسر می‌شوند:

$$\overline{x^i} = x^i - m \qquad (۲-۲)$$

که در معادله بالا $m = \frac{1}{c}\sum_{i=1}^{c} x_i$ و نهایتا داریم :

$$\overline{X} = [\overline{X}^1, \overline{X}^2, ...., \overline{X}^c] \qquad (۳-۲)$$

د - ماتریس کوواریانس از ضرب ماتریس $\overline{X}$ در ترانهاده آن بدست می‌آید :

$$\Omega = \overline{X}.\overline{X^T} \qquad (۴-۲)$$

ه- ماتریس $\Omega$ دارای ابعاد $N*N$ است. بردارهای ویژه و مقادیر ویژه این ماتریس به ترتیب بردارها و مقادیری هستند که در رابطه (۲-۵) صدق کنند. در این معادله $V$ بردار ویژه و $\Lambda$ مقدار ویژه است.

$$\Omega V = \Lambda V \qquad (۵-۲)$$

و - مقادیر و بردارهای ویژه را به ترتیب، از بزرگترین به کوچکترین مقدار ویژه مرتب می‌کنیم. بزرگترین مقدار ویژه متناسب با برداری است که در جهت بیشترین تغییرات است و به تدریج که به سمت مقادیر ویژه‌ی کوچکتر می‌رویم، به بردارهایی با میزان پراکندگی کمتر می‌رسیم، که در برخی حالات تغییراتی که این بردارها نماینده آن هستند به حدی کوچکند که می‌توان از آنها صرفنظر کرد.

ز - داده ها را به زیرفضای مورد نظر منتقل می‌کنیم:

$$X_{project} = V^T . \overline{X} \qquad (۶-۲)$$

حال برای شناسایی تصویر ورودی جدید باید مراحل زیر را انجام داد

$$\overline{X}_{test} = X_{test} - m \qquad (۷-۲)$$

ب - آن را به همان زیر فضای $V$ میبریم:

$$X_{projecttest} = V^T . \overline{X_{test}} \qquad (۸-۲)$$

ج - در این حالت تصویری که بیشترین شباهت به تصویر منتقل شده را داشته باشد، به عنوان نتیجه‌ی شناسایی معرفی می‌کنیم.

در شکل ۲-۲ نمونه‌ای از چهره ویژه های بدست آمده آورده شده است:

شکل ۲-۲- (راست) چهره ویژه‌های بدست آمده به ازای مجموعه تصاویر

(چپ) میانگین تصاویر[۱۱]

### ۲-۲-۱-۲- روش چهره فیشر:

در روش چهره فیشر از آنالیز جداساز خطی[۱۷] برای انتقال داده‌ها به زیرفضای مورد نظر استفاده می‌شود[۲]. در روش PCA به ارتباط بین داده‌های یک کلاس (تصاویر متعلق به یک کلاس) توجهی نمی‌شود و کلاسه‌بندی بدون در نظر گرفتن کلاسی که داده متعلق به آن است انجام می‌شود.

ایده اصلی در مورد روش LDA این است که بردارهایی از فضای چهره‌ها را بیابیم که تمایزات بین کلاس‌های چهره (افراد مختلف موجود در بانک چهره) را به بهترین نحو نمایش دهد؛ در حالی که در PCA هدفمان یافتن بردارهایی بود که ویژگی‌های چهره را به بهترین نحو توصیف کند.

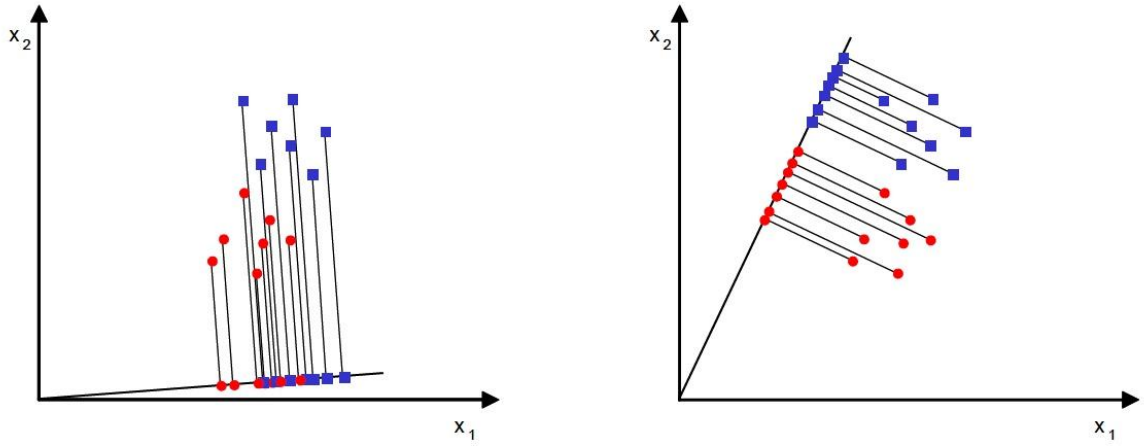
به عبارت دیگر با داشتن ویژگی‌های مستقل هر کلاس از چهره ها، LDA یک ترکیب خطی از آنها را به نحوی ایجاد می‌کند که بیشترین تفاوت‌ها را بین کلاس‌های مختلف چهره بدست آورد. برای مثال در شکل ۲-۳ فرض کنیم دو کلاس داده با پراکندگی داده شده موجود می‌باشد. حال این داده ها را به یک خط منتقل می‌کنیم، بسته به جهت این خط، داده‌ها می‌توانند بخوبی از هم جدا شوند (شکل سمت راست) یا اینکه داده‌ها دچار درهم تنیدگی شوند (شکل سمت چپ). جداساز فیشر سعی می‌کند بهترین خطی که این داده ها را از هم جدا می‌کند، پیدا کند.

[17] Linear Discriminant Analysis

در LDA سعی داریم به یک تعریف واحد از کلاس چهره‌ها برسیم. در ابتدا هر کلاس از چهره‌ها را به عنوان یک قسمت از فضای چهره‌ها تعریف می نماییم. قدم بعدی محاسبه یک انتقال خطی است به شکلی که بیشترین اختلاف‌ها را بین هر یک از کلاس‌های چهره نمایش دهد. برای محاسبه این انتقال به دو تعریف نیاز داریم. اولین تعریف که تفاوت‌های داخلی هر یک از کلاس‌ها را مشخص می‌کند، ماتریس پراکندگی داخل کلاسی نامیده می شود:

$$S_w = \sum_{j=1}^{c} \sum_{i=1}^{N_j} (x_i^j - \mu_j)(x_i^j - \mu_j)^T \qquad (۹-۲)$$

در تعریف فوق $x_i^j$، $i$امین نمونه از کلاس چهره‌ی فرد $j$ ام، $\mu_j$ بردار میانگین چهره های کلاس $j$ و $c$ تعداد کلاس ها هستند. $N_j$ نیز تعداد نمونه های موجود برای هر کلاس در بانک چهره ها را تعیین می کند. تعریف دوم به تفاوت بین کلاس‌های چهره توجه دارد و ماتریس پراکندگی بین کلاسی نام دارد:




شکل ۲-۳- جدا ساز Fisher

خطی که داده ها را به خوبی جدا می‌کند (شکل سمت راست) و خطی که نمی‌تواند داده‌ها را به خوبی جدا کند (شکل سمت چپ)

$$S_b = \sum_{j=1}^{c} (\mu_j - \mu)(\mu_j - \mu)^T \qquad (۱۰-۲)$$

در ماتریس پراکندگی بین کلاسی، $\mu$ میانگین تمام چهره‌ها را در بانک چهره‌ها مشخص می‌کند. هدف این است که معیار پراکندگی بین کلاسی را به بیشترین مقدار خود برسانیم، درحالی که باید پراکندگی داخل کلاسی را حداقل کنیم. به عبارت دیگر باید مقدار $\frac{S_b}{S_w}$ ماکزیمم شود. یک راه محاسبه مقادیر ویژه معادله ۲-۱۱ است:

$$(S_w^{-1}S_b)W = \lambda W \qquad (۱۱-۲)$$

ماتریس $W$ بدست آمده از معادله بالا همان ماتریس انتقال از فضای تصاویر به زیرفضای چهره‌ها می‌باشد. این ماتریس، در حقیقت ماتریس ستونی از بردارهای ویژه ماتریس $S_w^{-1}S_b$ است. می‌توان برای کاهش بعد زیر فضای حاصل، که به زیرفضای LDA معروف است، تعداد $M$ ستون از $N$ ستون ماتریس $W$ که بهترین مقادیر ویژه را دارند، انتخاب کنیم و سایر ستون‌ها را حذف کنیم. حال اگر چهره $X$ برای شناسایی به سیستم داده شود، با استفاده از فاصله اقلیدسی، نزدیک‌ترین کلاس را به کلاس چهره پیدا می کنیم. فرمول محاسبه فاصله بین چهره $X$ و کلاس $i$ام با میانگین $\mu_i$ به شکل زیر است :

$$D_i(X) = W^T(X - \mu_i) \qquad i = 1,2,\ldots,m \qquad (۱۲-۲)$$

از معایب روش LDA این است که چنانچه توزیع داده ها از توزیع گوسی پیروی نکند (از آنجا که این کلاسه‌بند نمی‌تواند حالت های غیر خطی را تولید کند) نمی‌توان آنها را از هم جدا کرد[۱۴].

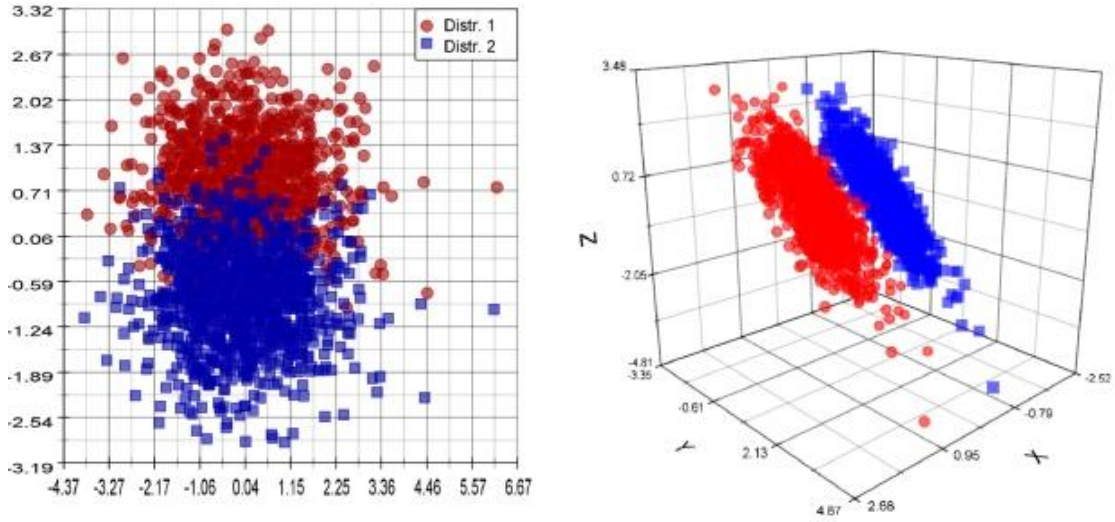
### ۲-۲-۲-تحلیل زیرفضای غیرخطی:

روش‌های خطی که در بخش قبل به آن‌ها اشاره شد، امکان جداسازی اطلاعاتی که توزیع آنها غیرخطی است را ندارند. لذا برای حل این مشکل ابتدا داده‌های اصلی را به فضایی با ابعاد بالاتر منتقل کرده و سپس از توابع خطی برای جداسازی آنها استفاده می‌کنند. به عبارت دیگر با یک نگاشت غیرخطی اطلاعات $x_1,.....,x_n \in \Re^N$ به فضای ویژگی F که می‌تواند ابعاد بسیار بالاتری داشته باشد، نگاشت می‌شوند:

$$\begin{aligned} \Phi &: \Re^N \to F \\ x &\to \Phi(x) \end{aligned} \qquad (۱۳-۲)$$

برای جداسازی اطلاعات نگاشت شده می‌توان همان الگوریتم‌هایی را که در فضای $\Re^N$ اجرا می‌شد را در فضای $F$ اجرا کرد. با داشتن اطلاعات نگاشت شده می‌توان از کلاسه‌بندهای عادی برای جداسازی استفاده کرد. برای مثال، پراکندگی داده ها در شکل۲-۴ را در نظر بگیرید. در فضای دو بعدی امکان جداسازی این دو توزیع وجود ندارد. در حالی که همین داده‌ها در فضای بالاتر با یک صفحه جدا خواهند شد.

پس از انتقال داده ها به فضاهای غیرخطی، از توابع جداساز عادی، مثل PCA و LDA استفاده می‌شود. پس از اعمال این جداسازها، به آنها KPCA و KLDA می‌گویند. نتایج بررسی‌هایی که در [۱۵] انجام شده، نشان می‌دهد که جداسازهای غیر خطی بهتر از سایر روش‌ها عمل می‌کنند.




شکل ۲-۴- داده‌هایی که امکان جداسازی آنها در فضای دو بعدی وجود ندارد (شکل الف) را می‌توان با یک صفحه در فضای سه بعدی از هم جدا کرد (شکل ب) .

## ۲-۳- روش‌های مدل مبنا :

هدف این گونه از روش‌ها ارایه‌ی مدلی از چهره است که تفاوت‌های چهره را نشان دهد. روش‌های قدیمی شناسایی چهره که بر مبنای پارامترهای چهره بودند کمک زیادی به این روش‌ها کرده‌اند، مثل روشهای تطبیق پارامتر[18] که از فاصله ی بین اجزای صورت استفاده می‌کنند[۱۶]. در روش‌های مدل مبنا چهره افراد با مدلی از پیش تعیین شده تطبیق داده شده و داده‌های این تطبیق به عنوان ویژگی‌های استخراج شده، ذخیره می‌شوند. از جمله این روش‌ها روشی مبتنی بر تطابق گراف الاستیک[19] است که توسط ویسکوت و دیگران ارایه شد[۴]. همچنین با ترکیب ظاهر و خصوصیات چهره، روش دیگری بنام مدل شکل فعال[20] توسط کوتز و دیگران ارایه شد[۱۷] که بعدها توسط خود او و همکارانش گسترش داده شد

[18] Feature-Base Matching
[19] Elastic Bunch Graph Matching (EBGM)
[20] Active Shape Model (ASM)

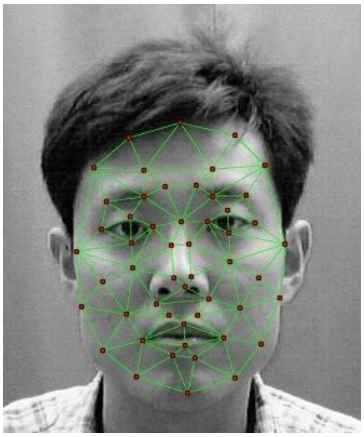
و روش مدل ظاهر فعال[۲۱] بوجود آمد[۵]. روش‌های مدل مبنا از سه مرحله اصلی تشکیل شده‌اند. الف- ساخت مدل ب- تطبیق مدل با چهره‌ی افراد ج- استفاده از پارامترهای تطبیق برای شناسایی.

در ادامه روش‌های تطابق گراف الاستیک و مدل شکل فعال بررسی شده است.

**۱-۱-۱**

**۱-۱-۲ ۲-۳-۱- تطابق گراف الاستیک :**

در این روش چهره به صورت یک گراف متشکل از نقاطی است که در چهره مشخص شده اند (مثل گوشه‌های چشم، دهان، بینی و ...). یک نمونه از این گراف ها، در شکل ۲-۵ آورده شده است.



شکل ۲-۵- گراف چهره

هر نقطه شامل ضرایب ۴۰ موجک گابور است. این ضرایب با اعمال موجک گابور در ۵ فرکانس و ۸ جهت بدست می‌آیند. مجموعه‌ی این نقاط تشکیل یک گراف را می‌دهند. در این گراف برای هر نقطه مهم[۲۲] در چهره یک گره داریم. در این روش از شباهت بین دو گراف تصویر برای شناسایی استفاده می‌شود.

[21] Active Apearance Model (AAM)
[22] Landmark

برای انجام این کار ابتدا نقاط کلیدی چهره مثل بینی، چشم، ابرو و ... در تصویر شناسایی می‌شوند، سپس با کمک موجک گابور خصوصیات هر یک از نقاط مهم استخراج می‌شوند که به آنها گابورجت[۲۳] می‌گویند. گره‌های گراف چهره شامل این نقاط مهم می‌باشند و هر گره شامل گابورجت هایی است که از آن نقطه استخراج شده‌اند. شباهت بین دو تصویر از طریق شباهت بین دو گراف چهره به دست می‌آید.

**۲-۳-۲- مدل شکل فعال :**

مدل شکل فعال بوسیله کوتز و در [۱۷] معرفی شده است. این روش یک مدل آماریست که در آن شکل-ها بوسیله‌ی مجموعه‌ای از نقاط توصیف می‌شوند. این نقاط، مجاز به تغییر شکل مطابق با مدهای تغییر شکل اصلی بوده که خود این مدهای اساسی از یک مجموعه آموزشی از تصاویر بوسیله تحلیل مولفه های اصلی بدست آمده‌اند. به این ترتیب که بردارهای $X_i$ که مطابق معادله‌ی (۲-۱۵) تعریف شده‌اند، از مجموعه‌ای شامل $n$ تصویر آموزشی که بطور دستی علامت گذاری شده اند، استخراج می‌شوند. هر بردار $X_i$ عبارت است از :

$$X_i = (x_{00}y_{00},\ldots,x_{ij}y_{ij}) \qquad (۱۵-۲)$$

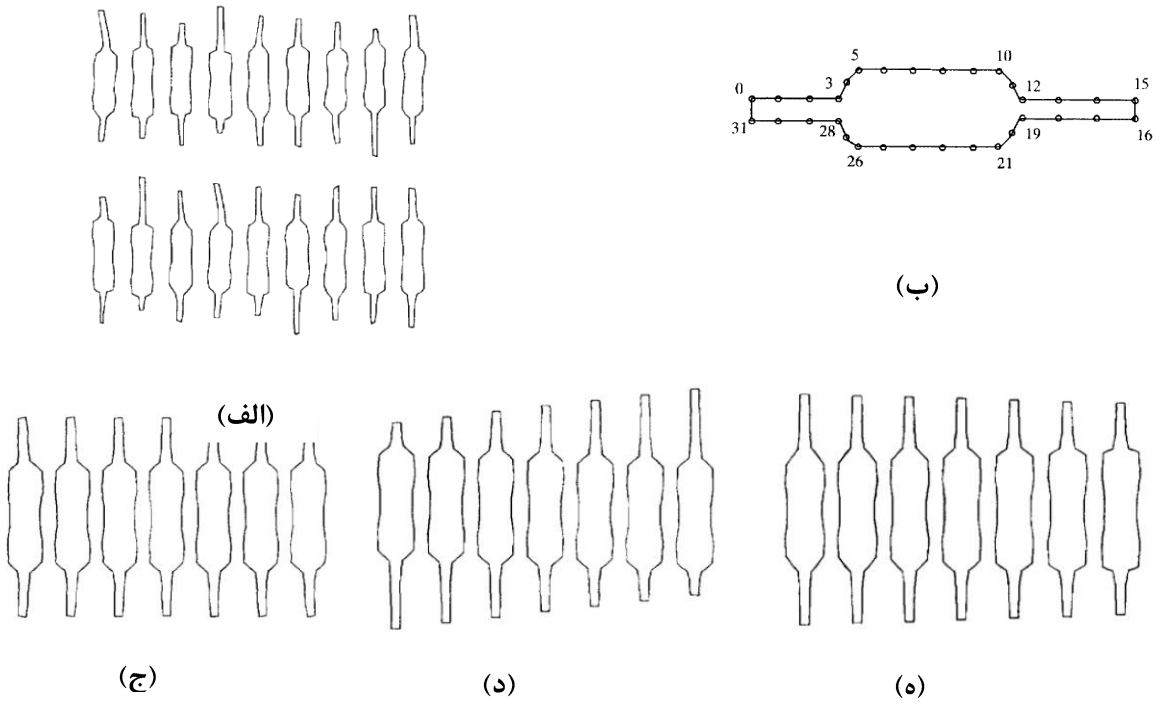
در این رابطه $(x_{ij}y_{ij})$، مختصات $x$ و $y$ نقطه $j$ ام از عکس $i$ ام است. تصاویر الگوی مورد استفاده باید دربردارنده تنوع کاملی از ترکیب‌های متفاوت چهره باشند. سپس روی $n$ بردار $X_i$، ماتریس کووراینس و سپس مقادیر ویژه و بردارهای ویژه محاسبه می شود. بردارهای ویژه‌ای که دارای مقادیر ویژه بزرگتر باشند، توصیف کننده پر اهمیت ترین مدهای پراکندگی بوده و مولفه های اساسی این فضا را تشکیل می دهند. پس برای شناسایی مدهای اساسی چهره کافیست تنها $d$ بردار ویژه اول نگاه داشته شود. به طور کلی، مختصات نقاط کانتور یک شکل دلخواه $X$ که بوسیله مدل شکل فعال ارائه می‌شود، می‌تواند با استفاده از معادله زیر تقریب زده شود:

[23] Gabor jet

$$X = \overline{X} + pb \qquad (۱۶-۲)$$

که در معادله بالا $\overline{X}$ شکل متوسط، $P = (p_1, p_2, ..., p_d)$، ماتریس اولین $d$ بردار ویژه و $b$، برداری شامل وزن‌هایی برای هر بردار ویژه می‌باشد که این وزن‌ها از تصویر کردن بردارهای اولیه در راستای مولفه‌های اساسی در این فضا بدست می‌آیند.

به عنوان نمونه، در شکل (۲-۶) اعمال روش مدل شکل فعال را روی شکل مقاومت‌ها نشان داده‌ایم. در (الف) چند نمونه از انواع اشکال مقاومت به عنوان نمونه‌های آموزش آورده شده و نقاط گسسته‌ای از مرزهای شکل همان‌طور که در (ب) نشان داده شده، استنتاج شده است. از (ج) تا (ه) تاثیرات تغییر وزن سه مولفه اصلی ابتدایی روی تغییر اشکال نشان داده شده است.




شکل ۲-۶- اعمال روش ASM روی مقاومت [۱۸]

### ۲-۴- مقدماتی در مورد نمایش تنک:

روش نمایش تنک در تشخیص چهره را می‌توان یکی از روش‌های مبتنی بر ظاهر قلمداد کرد. اما از آنجا که بحث اصلی ما در این پایان‌نامه روی این موضوع است، این بخش را بصورت مجزا آورده‌ایم.

### ۲-۴-۱- تابع نرم و تابع شبه نرم :

تابع حقیقی $\|.\|$ تعریف شده بر فضای برداری $X$ را نرم می‌نامیم اگر در سه خاصیت زیر صدق کند:

$$\forall a \in F, v \in V : \left\| a \vec{v} \right\| = |a| . \left\| \vec{v} \right\| \quad (۱۷-۲)$$

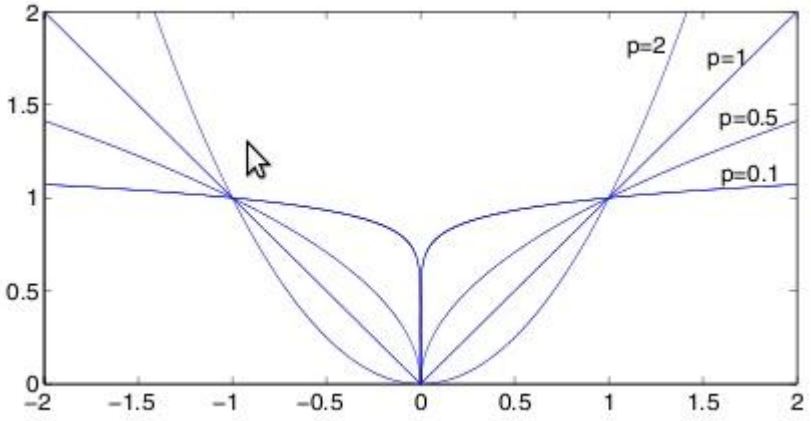
$$\forall \vec{u}, \vec{v} \in V : \left\| \vec{u} + \vec{v} \right\| \le \left\| \vec{u} \right\| + \left\| \vec{v} \right\| \quad (۱۸-۲)$$

$$\forall \vec{u} \in V : \left\| \vec{u} \right\| \ge 0, \forall \vec{u} \in V : \left\| \vec{u} \right\| = 0 \Leftrightarrow \vec{u} = 0 \quad (۱۹-۲)$$

از این تابع برای بدست آوردن فاصله از مبدا نیز استفاده می شود. در حالت کلی نرم $P$ بصورت زیر است:

$$\|x\|_p = \sqrt[p]{|x_1|^p + \ldots + |x_n|^p} \quad (۲۰-۲)$$

همان طور که از معادله‌ی بالا برمی آید، تابع نرم برای $0<P<1$ همه‌ی خواص تابع نرم را نداشته و عموما شبه نرم خوانده می‌شوند[۱۹]. در شکل ۲-۷ تابع نرم را برای چند مقدار مختلف $P$ آورده ایم.




شکل ۲-۷- نرم های مختلف [۸]

### ۲-۴-۲- مباحث ریاضی در مورد نمایش تنک

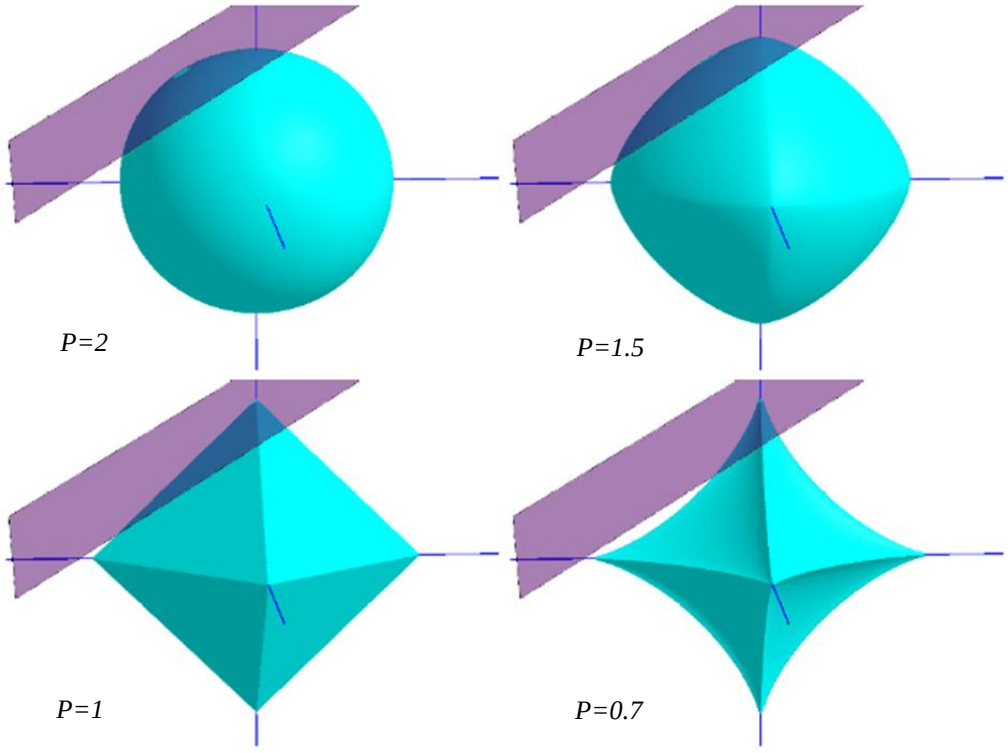
در فصل ۱ گفتیم که اگر واژه‌نامه‌ی ما فرا کامل باشد، آنگاه با فرض رتبه کامل بودن واژه‌نامه بیشمار جواب برای معادله (۱-۱) وجود خواهد داشت. از آنجایی که از ابتدا به دنبال ساده‌ترین نمایش سیگنال بودیم، مناسب ترین جواب برای ما تنک‌ترین جواب ممکن است. برای به دست آوردن این جواب باید مساله‌ی زیر را حل کنیم :

$$\min_x \| x \|_0 \qquad .s.t \qquad y=Dx \qquad (۲۱-۲)$$

در معادله‌ی (۲-۲۱) $\| x \|_0$ تعداد درایه های غیر صفر x است. معادله‌ی (۲-۲۱) غیر محدب است و از طرفی همان‌طور که در شکل (۲-۸) دیده می‌شود، تابع $\| x \|_0$ ناهموار و مشتق ناپذیر بوده و علاوه بر این، حل این مساله مستلزم یک جستجوی ترکیباتی است که در ابعاد بالا امکان پذیر نیست[۲۰]. از آنجا که عملاً نمی‌توان از نرم صفر استفاده کرد باید جایگزینی برای آن معرفی کرد. همان‌طور که در شکل (۲-۸) دیده می‌شود تابع نرم یک نزدیک‌ترین تابع محدب به نرم صفر است[۸]. بنابراین داریم :

$$\min_x \| x \|_1 \qquad \text{s.t.} \qquad y=Dx \qquad (۲۲-۲)$$

حال می‌خواهیم با ذکر مثالی نشان دهیم که نرم‌های یک و پایین‌تر جواب تنک دارند و نرم های بالاتر از یک، منجر به جواب تنک نخواهند شد:




شکل ۲-۸- فصل مشترک توپ های نُرم با دسته معادلات $y=Dx$ به ازای مقادیر مختلف $P$ [۸]

در ابتدا باید توجه کنیم که در شکل (۲-۱۰) تنک‌ترین پاسخ ممکن را روی محورهای مختصات خواهیم داشت. با توجه به این شکل مشخص است که محل تقاطع مجموعه $y=Dx$ و توپ‌های نُرم، تنها در

فضاهای محدب که دارای $p<1$ است، روی محورهای مختصات خواهند بود. در این شکل، فصل مشترک مجموعه $y=Dx$ با توپ های نُرم برای چند مقدار مختلف $p$ نشان داده شده است. همان‌طور که مشاهده می‌شود تنها نُرم یک می‌تواند به عنوان بهترین تقریب محدب شبه نرم صفر، به جواب‌های تنکی دست پیدا کند.

## ۲-۵- روش‌های تولید ضرایب تنک

الگوریتم‌های به دست آوردن تنک‌ترین نمایش سیگنال را می توان به دو دسته‌ی کلی تقسیم کرد[۲۱]: الگوریتم های تکرار و الگوریتم‌های بهینه سازی. الگوریتم‌های تکرار، تلاش می‌کنند بر اساس همبستگی بین واژه‌نامه و باقیمانده، یک یا چند اتم را انتخاب کرده و با استفاده از آنها این باقیمانده را به روز کنند. روش عمومی در این دسته از الگوریتم‌ها مبتنی بر تخمین گام به گام سیگنال با استفاده از اتم‌های واژه-نامه است. این الگوریتم‌ها جزو روش‌های عددی محسوب می‌شوند. اما دسته‌ی دوم الگوریتم‌ها، همان‌طور که از نامشان پیداست بدنبال حل یک مساله‌ی بهینه سازی هستند. عموماً، الگوریتم‌های تکراری نسبت به الگوریتم‌های بهینه سازی سریع‌ترند؛ چون پیچیدگی الگوریتم‌های بهینه‌سازی با افزایش ابعاد مساله بیشتر خواهد شد و البته دقت روش‌های بهینه سازی نسبت به روش‌های تکراری در حالت کلی بیشتر است. طی بخش‌های بعد، مختصری از این الگوریتم‌ها را خواهیم آورد.

### ۲-۵-۱- الگوریتم‌های تکراری

الگوریتم‌های تکراری بدلیل سادگی نسبی و همچنین سرعت بالا همواره مورد توجه محققین بوده‌اند. از شمار این روش‌ها می‌توان به الگوریتم‌های MP [۲۴]، OMP[۲۵] و IHT[۲۶] اشاره کرد. ما در این بخش روش-های MP و OMP را معرفی خواهیم کرد.

[24] Matching Pursuit
[25] Orthogonal Matching Pursuit

### ۲-۵-۱-۱ الگوریتم MP

این الگوریتم در مرجع [۲۲] معرفی شد. این روش مانند دیگر روش‌های این گروه، در زمره‌ی روش‌های عددی است. شبه کد این الگوریتم در شکل ۲-۹ آورده شده است. همان‌طور که از این شبه کد و از شکل-های ۲-۱۰ مشهود است، باید گام‌های زیر را تا رسیدن باقیمانده به یک مقدار مشخص برداشت:

در ابتدا، بردار ضرایب را صفر در نظر گرفته و بنابراین باقیمانده با سیگنال اصلی برابر خواهد شد. سپس در گام‌های بعدی اتمی از واژه‌نامه را انتخاب می‌کنیم که بیشترین همبستگی را با باقیمانده‌ی سیگنال دارد. حال باقیمانده و بردار ضرایب را به روز می‌کنیم.

$$\min_{\alpha\in\mathbb{R}^p} \|\underbrace{\mathbf{y}-\mathbf{D}\alpha}_{\mathbf{r}}\|_2^2 \quad \text{s.t.} \quad \|\alpha\|_0 \le L$$

1: $\alpha \leftarrow 0$
2: $\mathbf{r} \leftarrow \mathbf{y}$ (residual).
3: **while** $\|\alpha\|_0 < L$ **do**
4: Select the atom with maximum correlation with the residual

$$\hat{i} \leftarrow \arg\max_{i=1,\dots,p} |\mathbf{d}_i^T \mathbf{r}|$$

5: Update the residual and the coefficients

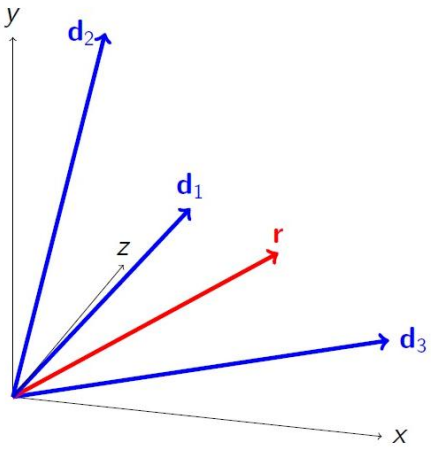
$$\begin{aligned} \alpha[\hat{i}] &\leftarrow \alpha[\hat{i}] + \mathbf{d}_{\hat{i}}^T \mathbf{r} \\ \mathbf{r} &\leftarrow \mathbf{r} - (\mathbf{d}_{\hat{i}}^T \mathbf{r})\mathbf{d}_{\hat{i}} \end{aligned}$$

6: **end while**

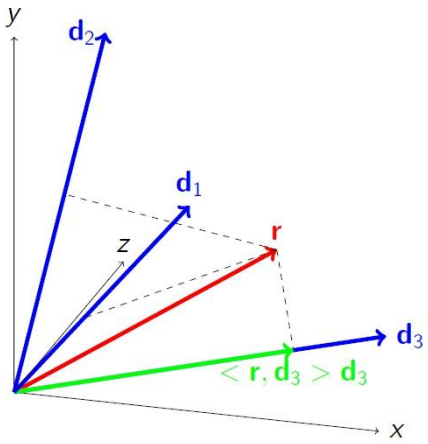
شکل ۲-۹- الگوریتم MP [۲۳]

[26] Iterative Hard Thresholding

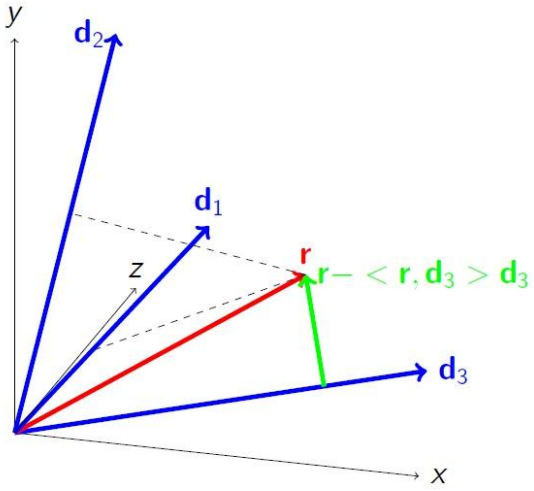
شکل های زیر روال کدینگ تنک توسط روشMP را نشان می‌دهد:



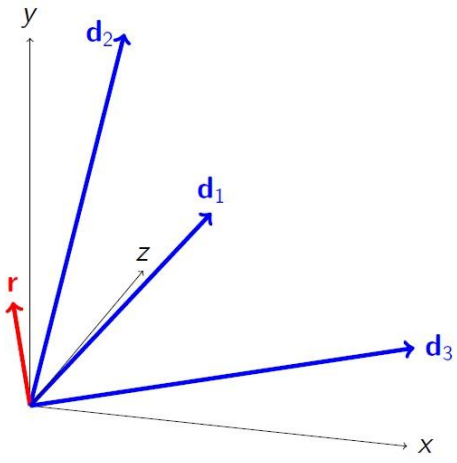

الف - $\alpha = (0,0,0)$



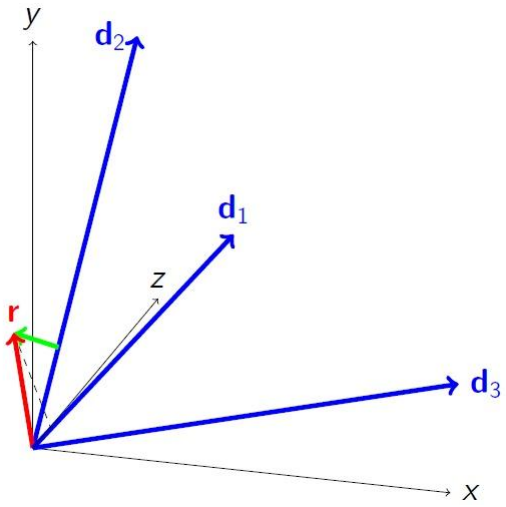

ب - $\alpha = (0,0,0)$



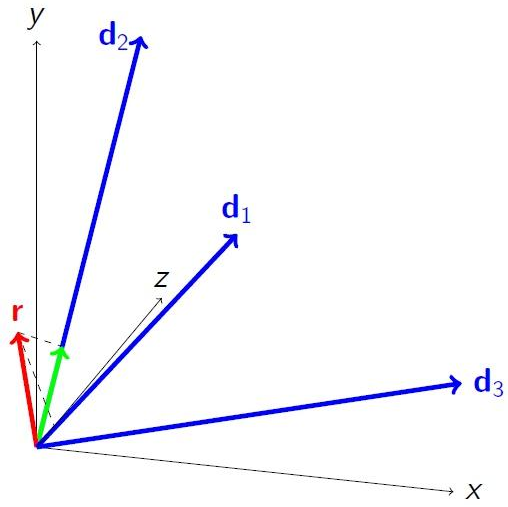

ج - $\alpha = (0,0,0)$



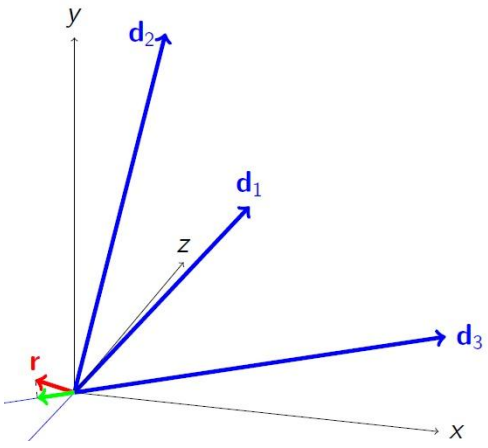

د - $\alpha = (0,0,0.75)$



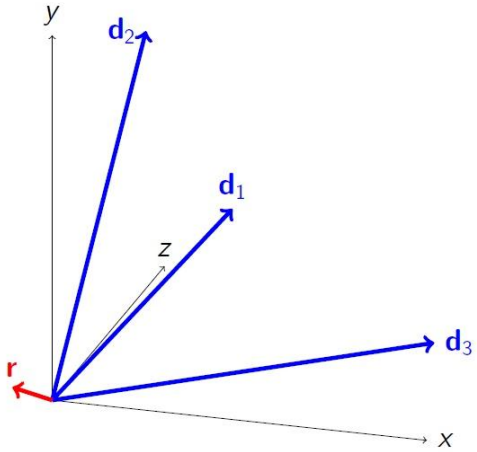

هـ - $\alpha = (0,0,0.75)$



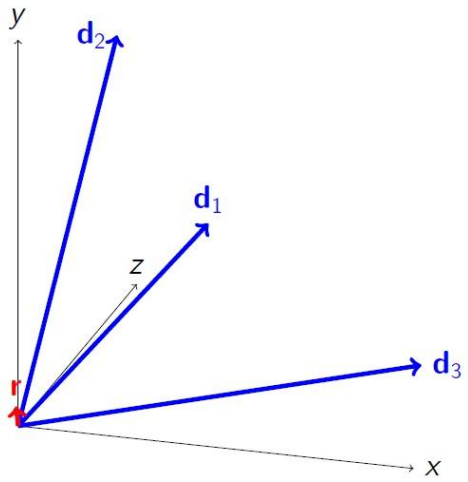

و - $\alpha = (0,0,0.75)$




ح - $\alpha = (0,0.24,0.75)$




ز - $\alpha = (0,0.24,0.75)$




ط - $\alpha = (0,0.24,0.75)$

شکل ۲-۱۰- روال عملکرد الگوریتم MP. بردارهای d اتم های ماتریس واژه نامه، $\alpha$ بردار ضرایب و r بردار باقیمانده است. همان طور که مشهود است، بردار d در هر مرحله کوچکتر می شود[۲۳].

### ۲-۵-۱-۲- الگوریتم OMP

الگوریتم OMP از آنجا که قطعا تنکی را لحاظ می‌کند، یکی از پرکاربردترین روش‌های تکراری است. این روش در [۲۴] معرفی شد. شالوده‌ی این الگوریتم شبیه الگوریتم MP است. اما در این الگوریتم بر خلاف الگوریتم MP، سیگنال را روی تمام اتم‌های برنده تا آن مرحله تصویر می‌کنیم و باقیمانده را بر این اساس

بدست خواهیم آورد. همان‌طور که در گام چهارم از شبه کد۲-۱۱ مشهود است، تمام اتم‌های برنده را در مجموعه‌ای به اسم $\Gamma$ نگاه می‌داریم.

$$\min_{\alpha \in \mathbb{R}^p} \|\mathbf{y} - \mathbf{D}\alpha\|_2^2 \quad \text{s.t.} \quad \|\alpha\|_0 \leq L$$

1: $\Gamma = \emptyset$.
2: **for** *iter* $= 1, \ldots, L$ **do**
3: Select the atom which most reduces the objective

$$\hat{\imath} \leftarrow \arg\min_{i \in \Gamma^C} \left\{ \min_{\alpha'} \|\mathbf{y} - \mathbf{D}_{\Gamma \cup \{i\}} \alpha'\|_2^2 \right\}$$

4: Update the active set: $\Gamma \leftarrow \Gamma \cup \{\hat{\imath}\}$.
5: Update the residual (orthogonal projection)

$$\mathbf{r} \leftarrow (\mathbf{I} - \mathbf{D}_\Gamma (\mathbf{D}_\Gamma^T \mathbf{D}_\Gamma)^{-1} \mathbf{D}_\Gamma^T)\mathbf{y}.$$

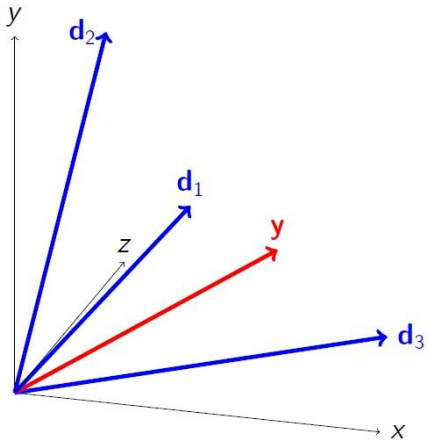
6: Update the coefficients

$$\alpha_\Gamma \leftarrow (\mathbf{D}_\Gamma^T \mathbf{D}_\Gamma)^{-1} \mathbf{D}_\Gamma^T \mathbf{y}.$$

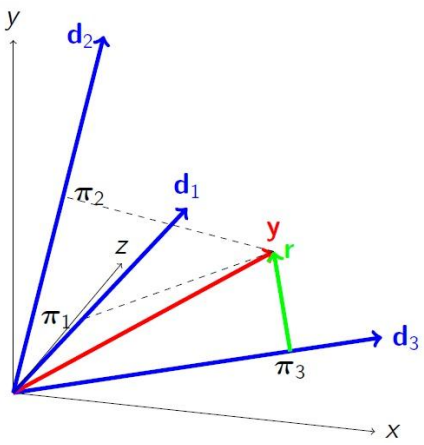
7: **end for**

شکل ۲-۱۱- الگوریتم OMP [۲۳]

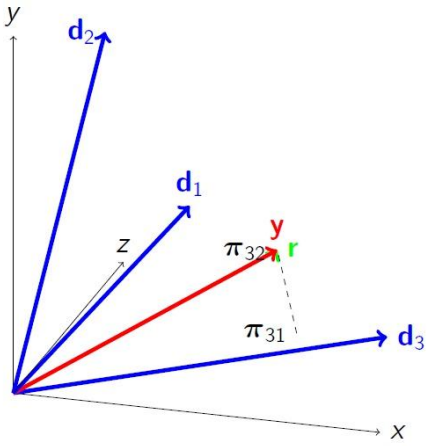
شکل های زیر روال کدینگ تنک توسط روش OMP را نشان می‌دهد:




الف - $\alpha = (0,0,0)$ و $\Gamma = \phi$




ب - $\alpha = (0,0,0)$ و $\Gamma = \{3\}$




ج - $\alpha = (0,0.29,0.63)$ و $\Gamma = \{3,2\}$

شکل ۲-۱۲ - در اشکال بالا، روال عملکرد الگوریتم OMP نشان داده شده است. بردارهای d اتم های ماتریس واژه نامه، $\alpha$ بردار ضرایب، r بردار باقیمانده و $\Gamma$ مجموعه اتم های برنده اند[۲۳].

### ۲-۵-۲- الگوریتم‌های بهینه سازی

در این بخش دسته دوم از الگوریتم‌های تنک کردن سیگنال را با معرفی یکی از الگوریتم‌های معروف در این زمینه بررسی می کنیم. در بسیاری از روش‌های متداول در این گروه می‌خواهیم مساله غیرمحدب تنک کردن ضرایب را با روشی خاص به مساله‌ای محدب تبدیل کنیم.

#### ۲-۵-۲-۱- الگوریتم FOCUSS

این روش در [۲۵] معرفی شده است. در این روش از الگوریتم کلی IRLS[۲۷] برای تقریب زدن نرم صفر توسط نرم دو وزن‌دار استفاده می‌شود. به عبارت دیگر در این الگوریتم می‌خواهیم مساله (۲-۲۴) را با معادله (۲-۲۵) حل کنیم.

$$min((x))_p^p \qquad s.t. \qquad y = Dx \qquad (۲-۲۴)$$

$$x^{(k+1)} = \arg\min_x \sum_i w_i^{(k)} x_i^2 = x^T W_k x \qquad \text{s.t.} \qquad y = Dx \qquad (۲-۲۵)$$

در معادله‌ی (۲-۲۵) $w_k$ با استفاده از معادله (۲-۲۶) بدست خواهد آمد.

$$w_i^k = |x_i^k + \sigma|^{p-2} \qquad (۲-۲۶)$$

[27] Iteratively Re-Weighted Least Squares

$\sigma$برای جلوگیری از بد حالت شدن ماتریس آورده شده است. بنابراین می‌توان بجای مساله (۲-۲۴) معادله‌ی (۲-۲۵) را با ضرایب لاگرانژ حل کرد.

## ۲-۶- آموزش واژه‌نامه

### ۲-۶-۱- مقدمه

تا ابتدای این بخش، فرض ما این بود که واژه‌نامه‌ی ما مشخص است. اما باید برای هر کاربری خاصی واژه-نامه‌ی مشخص آن کاربرد را استفاده کرد. مثلا واژه‌نامه‌ای که برای تشخیص تصویر در یک پایگاه داده استفاده می‌شود، مسلماً با واژه‌نامه‌ای که برای حذف نویز از همان پایگاه داده بکار خواهد رفت متفاوت است. واژه‌نامه‌ای مناسب خواهد بود که به روند تنک کردن سیگنال کمک کند. در این بخش سعی خواهیم کرد بحث آموزش واژه‌نامه را روشن کنیم و در نهایت دو الگوریتم متداول برای آموزش واژه‌نامه را به اختصار بررسی خواهیم کرد.

### ۲-۶-۲- چرایی آموزش واژه‌نامه

اگر بخواهیم مجموعه‌ای از سیگنال‌ها را به روشی ساده و در عین حال با خطای کم نشان دهیم، بی‌گمان یکی از روش‌ها این است که برای همه‌ی این سیگنال‌ها، پایه‌ها یا اتم‌هایی یکسان و برای هر سیگنال ضرایبی متفاوت در نظر بگیریم. همیشه می‌خواهیم ساده‌ترین اتم‌ها را برگزینیم. ساده‌ترین مجموعه‌ی اتم‌ها یا همان واژه‌نامه، پایه‌های مستقل خطی هستند و معروف‌ترین واژه‌نامه‌ی مستقل خطی تبدیل فوریه است.

واژه‌نامه‌های مربعی مانند تبدیل فوریه که در آنها تعداد اتم‌ها با بعد سیگنال مساوی است را واژه‌نامه‌های کامل می‌گویند. این واژه‌نامه‌ها تنها برای نمایش مجموعه‌ی بسیار اندکی از سیگنال‌ها کاربردی است. واژه-

نامه‌های فوق کامل برای حل این مشکل معرفی شدند. در این واژه نامه‌ها، تعداد اتم ها از بعد سیگنال بیشتر است و بنابراین قادرند ویژگی های بیشتری از سیگنال را توصیف کنند[۲۶].

در تعریفی دیگر واژه نامه‌ها را به دو دسته‌ی وفقی[۲۸] و غیر وفقی[۲۹] تقسیم می‌کنند. واژه‌نامه‌های وفقی به سیگنال وابسته‌اند[۲۷][۲۸]. واژه‌نامه‌های غیر وفقی نمی‌توانند نمایش به اندازه‌ی کافی تنکی، ارائه دهند. برای غلبه بر این مشکل، بحث آموزش دیکشنری مطرح شد. با این روش، اتم‌های دیکشنری را طوری بدست خواهیم آورد که مهم‌ترین ویژگی‌های مشترک پایگاه داده را نشان دهد. این واژه‌نامه قطعاً نمایش به اندازه‌ی کافی تنکی برای سیگنال‌های آزمون آن پایگاه داده نیز ارائه خواهد داد. برای آموزش واژه‌نامه باید مساله‌ی کلی زیر را حل نمود:

$$\min_{D,X} \|Y - DX\|_F \qquad \text{S.t.} \qquad \|x_i\|_0 \le T_0 \qquad (۲۷-۲)$$

در معادله‌ی بالا $T_0$ حداکثر تعداد اتم‌ها برای نمایش هر سیگنال آموزشی است[۲۹]. دو دسته‌ی کلی برای آموزش واژه‌نامه مطرح است. روش "جستجوی جامع[۳۰] " و روش‌های "شبه K-Means" . از آنجا که روش‌های جستجوی جامع تمام حالت‌های ممکن برای واژه‌نامه را در نظر می‌گیرند، عملاً برای مجموعه‌های بزرگ ناکارآمدند. اما الگوریتم‌های گروه دیگر یا در چگونگی یافتن نمایش تنک سیگنال‌ها در واژه‌نامه فعلی و یا در چگونگی به روز کردن واژه‌نامه طوری که خطای کلی نمایش تنک سیگنال‌ها کاهش یابد، اختلاف دارند.

[28] Adaptive Dictionaries
[29] Non Adaptive Dictionaries
[30] brute-force search

### ۲-۶-۳- الگوریتم‌های آموزش واژه‌نامه

در این بخش، دو الگوریتم معروف آموزش دیکشنری را به اختصار مرور می کنیم. از آنجا که روش‌های جستجوی جامع برای بسیاری از مجموعه‌ها غیرکاربردی هستند، فقط الگوریتم‌های شبه K-Means را معرفی کرده‌ایم.

#### ۲-۶-۳-۱- الگوریتم MOD

روش MOD[31] توسط Engan و دیگران و در [۳۰] معرفی شد. در این روش برای بدست آوردن بردار ضرایب تنک، از هر الگوریتمی می‌توان استفاده کرد. سپس همان‌طور که در بخش قبل گفته شد باید معادله‌ی (۲-۲۷) را حل کرد. به عبارت دیگر باید مشتق این معادله را نسبت به D برابر با صفر قرار دهیم. بنابراین داریم:

$$D = YX^T(XX^T + \lambda I)^{-1} \qquad (۲-۲۸)$$

عبارت $\lambda I$ برای جلوگیری از بد حالت شدن ماتریس $XX^T$ اضافه شده است. در معادله‌ی بالا با داشتن $X$ از مرحله‌ی قبل $D$ را بدست آورده و این دو گام را تکرار می‌کنیم تا شرط معادله‌ی (۲-۲۷) حاصل شود. در ادامه شبه کد الگوریتم MOD را در شکل ۳-۱ آورده‌ایم.

[31] Method of Optimal Directions

**Task:** Train a dictionary $\mathbf{A}$ to sparsely represent the data $\{\mathbf{y}_i\}_{i=1}^{M}$, by approximating the solution to the problem posed in Equation (12.2).
**Initialization:** Initialize $k = 0$, and

- **Initialize Dictionary:** Build $\mathbf{A}_{(0)} \in \mathbb{R}^{n\times m}$, either by using random entries, or using $m$ randomly chosen examples.
- **Normalization:** Normalize the columns of $\mathbf{A}_{(0)}$.

**Main Iteration:** Increment $k$ by 1, and apply

- **Sparse Coding Stage:** Use a pursuit algorithm to approximate the solution of

$$\hat{\mathbf{x}}_i = arg\min_{\mathbf{x}} \|\mathbf{y}_i - \mathbf{A}_{(k-1)}\mathbf{x}\|_2^2 \quad \text{subject to} \quad \|\mathbf{x}\|_0 \le k_0.$$

  obtaining sparse representations $\hat{\mathbf{x}}_i$ for $1 \le i \le M$. These form the matrix $\mathbf{X}_{(k)}$.
- **MOD Dictionary Update Stage:** Update the dictionary by the formula

$$\mathbf{A}_{(k)} = arg\min_{\mathbf{A}} \|\mathbf{Y} - \mathbf{A}\mathbf{X}_{(k)}\|_F^2 = \mathbf{Y}\mathbf{X}_{(k)}^T\left(\mathbf{X}_{(k)}\mathbf{X}_{(k)}^T\right)^{-1}.$$

- **Stopping Rule:** If the change in $\|\mathbf{Y} - \mathbf{A}_{(k)}\mathbf{X}_{(k)}\|_F^2$ is small enough, stop. Otherwise, apply another iteration.

**Output:** The desired result is $\mathbf{A}_{(k)}$.

شکل ۲-۱۳- الگوریتم MOD [۲۳]

### ۲-۶-۳-۲- الگوریتم K-SVD

این روش توسط Aharon و دیگران معرفی شد[۳۱]. این الگوریتم هم مانند روش MOD شامل دو گام نمایش تنک سیگنال و بروز رسانی اتم‌های واژه‌نامه است. همچنین شبیه روش قبل، برای فاز اول از تمام الگوریتم‌های تکراری یافتن پاسخ تنک می‌توان استفاده کرد اما بر خلاف روش MOD که کل اتم‌ها را یکجا بروزمی‌کند، این روش اتم‌ها را یک به یک بروز می‌کند. اگر بخواهیم بطور خلاصه این روش را مرور کنیم، در گام اول واژه‌نامه $D$ را ثابت فرض کرده و با یکی از روش های کدینگ تنک مقدار مناسب برای

ماتریس $X$ را پیدا می کنیم. در گام بعد برای به روز کردن اتم $d_k$ بقیه‌ی اتم ها را ثابت می گیریم. در این صورت داریم :

$$E=\|Y-DX\|_F^2 = \|Y - \sum_{j=1}^{m} d_j x_j^T)\|_F^2 = \|(Y - \sum_{j\neq k}^{m} d_j x_j^T) - d_k x_k^T\|_F^2 = \|E_k - d_k x_k^T\|_f^2 \quad (۲۹-۲)$$

در ادامه شبه کد الگوریتم K-SVD را در شکل ۲-۱۳ نشان می‌دهیم.

**Task:** Train a dictionary $\mathbf{A}$ to sparsely represent the data $\{\mathbf{y}_i\}_{i=1}^{M}$, by approximating the solution to the problem posed in Equation (12.2).
**Initialization:** Initialize $k = 0$, and

- **Initialize Dictionary:** Build $\mathbf{A}_{(0)} \in \mathbb{R}^{n \times m}$, either by using random entries, or using $m$ randomly chosen examples.
- **Normalization:** Normalize the columns of $\mathbf{A}_{(0)}$.

**Main Iteration:** Increment $k$ by 1, and apply

- **Sparse Coding Stage:** Use a pursuit algorithm to approximate the solution of

$$\hat{\mathbf{x}}_i = arg\min_{\mathbf{x}} \|\mathbf{y}_i - \mathbf{A}_{(k-1)}\mathbf{x}\|_2^2 \quad \text{subject to} \quad \|\mathbf{x}\|_0 \le k_0.$$

  obtaining sparse representations $\hat{\mathbf{x}}_i$ for $1 \le i \le M$. These form the matrix $\mathbf{X}_{(k)}$.
- **K-SVD Dictionary-Update Stage:** Use the following procedure to update the columns of the dictionary and obtain $\mathbf{A}_{(k)}$: Repeat for $j_0 = 1, 2, \ldots, m$
  - Define the group of examples that use the atom $\mathbf{a}_{j_0}$,

$$\Omega_{j_0} = \{i \mid 1 \le i \le M,\ \mathbf{X}_{(k)}[j_0, i] \ne 0\}.$$

  - Compute the residual matrix

$$\mathbf{E}_{j_0} = \mathbf{Y} - \sum_{j \ne j_0} \mathbf{a}_j \mathbf{x}_j^T,$$

    where $\mathbf{x}^j$ are the $j$'th **rows** in the matrix $\mathbf{X}_{(k)}$.
  - Restrict $\mathbf{E}_{j_0}$ by choosing only the columns corresponding to $\Omega_{j_0}$, and obtain $\mathbf{E}_{j_0}^R$.
  - Apply SVD decomposition $\mathbf{E}_{j_0}^R = \mathbf{U}\Delta\mathbf{V}^T$. Update the dictionary atom $\mathbf{a}_{j_0} = \mathbf{u}_1$, and the representations by $\mathbf{x}_{j_0}^R = \Delta[1,1] \cdot \mathbf{v}_1$.
- **Stopping Rule:** If the change in $\|\mathbf{Y} - \mathbf{A}_{(k)}\mathbf{X}_{(k)}\|_F^2$ is small enough, stop. Otherwise, apply another iteration.

**Output:** The desired result is $\mathbf{A}_{(k)}$.

شکل ۲-۱۴- الگوریتم K-SVD [۲۳]

## ۲-۷- جمع بندی

این فصل را با معرفی برخی روش‌های معروف تشخیص چهره آغاز کردیم. دو دسته از این روش‌ها را برشمردیم و چند نمونه از هر یک را بررسی کردیم. سپس مباحث تئوری نمایش تنک را به همراه تعدادی از الگوریتم‌های موجود برای بازیابی این نمایش، به اختصار آوردیم. گفتیم که هسته‌ی مرکزی بحث نمایش تنک، یک دستگاه معادلات خطی فرو معین است و تعدادی از الگوریتم های موجود را، که در حالت کلی شامل دسته ی الگوریتم های تکراری و الگوریتم‌های مبتنی بر حل یک مساله‌ی بهینه سازی است، به اختصار مرور کردیم. در نهایت مساله‌ی آموزش واژه‌نامه را بررسی کردیم و بحث را با معرفی چند الگوریتم معروف در زمینه‌ی آموزش واژه‌نامه به اتمام رساندیم. در فصل بعد روش‌های پیشنهادی را معرفی خواهیم کرد.

# فصل سوم :

# معرفی روش پیشنهادی

## ۳-۱- مقدمه

بسیاری از روش‌های مطرح شده در علوم مهندسی برگرفته از روش‌هایی است که طبیعت از این روش‌ها برای طبقه بندی، تصحیح و... استفاده می‌کند. بعضی از این روش‌ها نیز روش‌هایی‌اند که یک انسان معمولی در زندگی روزمره، برای انجام کارهای خود بکار می بندد.

روش نمایش تنک سیگنال نیز از این قاعده مستثنی نیست. ما برای یافتن معنی یک واژه جدید، به یک واژه‌نامه مراجعه می کنیم؛ اگر واژه مورد نظر در واژه‌نامه موجود بود، به معنای آن رجوع کرده، ولی اگر عبارت دقیق آن واژه در واژه‌نامه موجود نبود به کلمات نزدیک به واژه مورد نظر رجوع می‌کنیم و با استفاده از آن کلمات، معنای واژه مورد نظر را استنباط خواهیم کرد. در روش نمایش تنک نیز دقیقا از همین الگوریتم استفاده می کنیم. در این روش واژه‌نامه‌ای متشکل از تصاویر آموزش تولید خواهیم کرد. و هر تصویر جدید را به این ماتریس واژه نامه عرضه می کنیم. از آنجا که تصاویری که برای تست به الگوریتم خواهیم سپرد با تصاویر واژه نامه دقیقا یکسان نیستند، سعی می کنیم تصاویر آزمون را با چند تصویر از تصاویر واژه‌نامه بسازیم یا به عبارت دیگر معنای آن را بیابیم. این کار را با اعمال ضرایبی به تصاویر واژه‌نامه انجام خواهیم داد.

یافتن معنای صحیح واژه جدید مشروط به رعایت چند نکته است. ابتدا آنکه واژه نامه‌مان تا حد ممکن غنی از واژه‌ها باشد. ثانیا فردی که می خواهد واژه جدید را با استفاده از واژه‌های واژه‌نامه استنباط کند در تشخیص واژه‌های نزدیک به واژه‌ی مورد نظر اشتباه نکند. نکته‌ی سوم، هم در سرعت و هم در دقت کار موثر است؛ و آن اینکه معنای واژه جدید را با کمترین کلمات ممکن توضیح دهد.

از طرفی در مسئله‌ی احراز هویت مانند سایر مسایل هوش مصنوعی باید راهبردمان نسبت به دو چالش را مشخص کنیم. ابتدا آنکه "قرار است از چه ویژگی هایی برای حل مسئله استفاده کنیم؟" و سپس "چه الگوریتمی را بکار خواهیم بست؟"

به کرات دیده ایم که یک استخراج ویژگی خوب، حتی با وجود یک الگوریتم ساده جواب بسیار خوبی را در بر داشته است. نکته‌ی دیگر آن که ویژگی مستخرج از داده ها باید با الگوریتم پیشنهادی همگام باشد؛ به دیگر سخن، ممکن است یک ویژگی روی الگوریتمی خاص جواب خوبی دهد، حال آنکه همان ویژگی روی الگوریتمی دیگر، نتیجه‌ی دلخواه را نداشته باشد. استخراج درست ویژگی برای یک الگوریتم خاص، مستلزم فهم درست ویژگی و همچنین الگوریتم است. و البته مسلم است که ویژگی خوب در کنار الگوریتمی مناسب نتیجه مورد نظرمان را به همراه خواهد داشت.

ما در این فصل و طی بخش های بعد، سعی خواهیم کرد هم ویژگی های مناسبی معرفی کنیم و هم الگوریتم های کارایی را پیشنهاد دهیم.

### ۳-۲- روش‌های پیشنهادی:

در فصل ۲ بحث آموزش دیکشنری برای پردازش تنک سیگنال را به همراه تعدادی از الگوریتم‌های موجود بررسی کردیم. اهمیت یک دیکشری "تنک کننده" را نیز بیان کردیم. گفتیم که این دیکشنری می‌تواند برجسته‌ترین ویژگی‌های یک کلاس مشخص از داده ها را استخراج کند. همچنین گفتیم که یکی از مشکلات روش‌های مبتنی بر بهینه سازی، زمان‌بر بودن و نیز حجم محاسبات بالای آنهاست. این موضوع مخصوصا در مورد داده‌های با حجم بالا رخ می‌نمایاند؛ تا آنجا که حل برخی مسائل را با امکانات سخت افزاری معمول ناممکن می‌کند. از این رو در روش‌های پیشنهادی که در ادامه خواهد آمد سعی شده با ارائه الگوریتم‌های بهینه شده تا حد ممکن از محاسبات بکاهد و از طرف دیگر حتی المقدور در صد بازشناسی را نیز بالا برد.

برای انسجام بیشتر مطلب، ابتدا روش‌های پیشنهادی را مطرح کرده و سپس، ویژگی هایی که به کمک آنها نتیجه‌ی بهتری عایدمان می‌شود را مطرح خوهیم کرد.

### ۳-۲-۱-روش نمایش تنک نظارتی بهبود یافته[۳۲] :

در روش نمایش تنک متداول، همه‌ی نمونه‌های آموزش در ماتریس واژه‌نامه جای می‌گرفتند؛ با این تصور که واژه‌نامه‌ای غنی از تعداد زیادی تصویر ساخته شده است. اگرچه این تصور درست به نظر می آید ولی بدلیل غیر هوشمند بودن رایانه، امکان تخصیص ضرایب غیر واقعی را بالا خواهد برد. از سوی دیگر سرعت کار نیز بسیار پایین خواهد آمد. بنابراین روش‌های تنک کردن واژه‌نامه مطرح شد. اما در روش پیشنهادی ما، به جای آنکه یک ماتریس واژه‌نامه را با همه‌ی نمونه‌های آموزشی بسازیم، برای هرکدام از این کلاس‌ها یک ماتریس واژه‌نامه می‌سازیم؛ سپس نمونه آزمون را به صورت ترکیب خطی هر کدام از این ماتریس‌های واژه‌نامه بیان می‌کنیم. در نتیجه ماتریس واژه‌نامه کوچکتر شده و یافتن ماتریس معکوس بسیار سریع‌تر انجام می شود. با این روش علاوه بر سرعت، دقت تشخیص نیز بهبود می یابد.

به عبارتی در این روش پیشنهادی، بجای استفاده از روش‌های ریاضی از مفهوم نمایش تنک برای تنک کردن واژه‌نامه استفاده خواهیم کرد. تفصیل بیشتر این روش بدین صورت است که در مرحله ی اول ابتدا داده‌های آزمون را توسط روش K-means خوشه یابی می کنیم. حال از هر خوشه یک یا چند پایه را به عنوان نماینده برمی گزینیم و در واژه نامه قرار می دهیم.

در مرحله‌ی دوم تمام این کلاس‌های نماینده را در ماتریس واژه نامه جایگذاری می کنیم. حال طبق معادله‌ی (۳-۱) سعی می کنیم یک ترکیب خطی از نمونه‌های آموزشی را برای هر کدام از نمونه های آزمون تعیین کنیم.

$$y \approx d_1 x_1 + d_2 x_2 + ... + d_n x_n \qquad (۱-۳)$$

[32] Modified Supervised Sparse Representation classification (MSSRC)

در هر صورت نمی توان نمونه‌ی آزمون را بطور دقیق با نمونه‌های آموزش ساخت. برای همین معادله (۳-۱) را بصورت تقریبی نوشتیم. در اینجا y نشان‌دهنده نمونه آزمون، d نمایانگر نمونه‌های آموزشی و x نشانگر ضرایب نمونه‌های آموزش است. اگر معادله‌ی تقریبی (۳-۱) را به صورت معادله (۳-۲) بیان کنیم :

$$Y = DX \qquad (۳-۲)$$

که در این معادله $X^T = [x_1, x_2, x_3, ... x_n]$ و $D = [d_1, d_2, d_3, ..., d_n]$ است؛ و هرکدام از ستون‌های واژه-نامه $D$ برابر با یکی از تصاویر نمونه‌های آموزش است. n تعداد همه‌ی نمونه‌های آموزش خوشه‌بندی شده است. آنگاه می‌توان بردار ضرایب $X$ را به صورت معادله (۳-۳) بدست آورد :

$$\tilde{X} = (D^T X + \mu I)^{-1} D^T y \qquad (۳-۳)$$

در رابطه (۳-۳) متغیر $\mu$ یک ثابت مثبت کوچک، و $I$ ماتریس واحد است. عبارت $\mu I$ را برای جلوگیری از بد حالت شدن ماتریس معکوس اضافه کرده‌ایم. حال باقیمانده هر کلاس را می‌توان به صورت معادله (۳-۴) بدست آورد :

$$\text{Residual} = y - \sum x_i d_i \qquad (۳-۴)$$

حال در اینجا M کلاس برتر را انتخاب می کنیم. مسلم است که کلاس‌هایی برنده‌اند که کمترین باقیمانده را داشته باشند. واضح است که تا این مرحله تعداد معتنابهی از ضرایب صفر خواهند شد. به عبارت ساده‌تر ماتریس واژه‌نامه‌ای که قرار بود شامل تعداد زیادی اتم باشد، حال بسیار کوچکتر شده است.

از این به بعد مرحله سوم الگوریتم آغاز می شود. در اینجا برای هر کدام از M کلاس برنده، یک ماتریس واژگان تشکیل می دهیم. به عبارت دیگر به جای آنکه یک ماتریس واژه‌نامه را با همه‌ی نمونه‌های آموزشی بسازیم، به ازای هرکدام از کلاس‌ها یک ماتریس واژه‌نامه می سازیم؛ سپس نمونه آزمون را به صورت ترکیب خطی هر کدام از این ماتریس‌های واژه نامه بیان می کنیم.

$$y_1 \approx a_1 t_1 + \ldots + a_n t_n \qquad (۵-۳)$$

در اینجا y نشان‌دهنده نمونه آزمون، t نمایانگر نمونه‌های آموزشی برنده و a نشانگر ضرایب نمونه‌های آموزش است. اگر معادله‌ی تقریبی (۳-۵) را به صورت معادله (۳-۵) بازنویسی کنیم :

$$Y = GA \qquad (۶-۳)$$

که در اینجا $A^T = [a_1,\ldots,a_r]$ و $G = [t_1,\ldots,t_r]$ است. برچسب r به معنی تعداد نمونه های آموزش از هر کلاس است. همان طور که گفته شد در مرحله ی سوم، برای هر کدام از کلاس ها، یک ماتریس واژه نامه می‌سازیم. بنابراین بهتر است معادله ی (۳-۶) را به صورت دسته معادلات (۳-۷) بیان کنیم:

$$\begin{aligned} Y &= G_1 A_1 \\ Y &= G_2 A_2 \\ &\ldots. \\ Y &= G_M A_M \end{aligned} \qquad (۷-۳)$$

$M$ در اینجا تعداد کلاس‌های برنده در مرحله‌ی دوم است. آنگاه می‌توان بردارهای $A$ را توسط معادلات (۳-۸) بدست آورد :

$$\begin{aligned}\tilde{A}_1 &= (G_1{}^T G_1 + \gamma I)^{-1} G_1{}^T y \\ \tilde{A}_2 &= (G_2{}^T G_2 + \gamma I)^{-1} G_2{}^T y \\ &\cdots \\ \tilde{A}_M &= (G_M{}^T G_M + \gamma I)^{-1} G_M{}^T y\end{aligned} \qquad (۳-۸)$$

در رابطه (۳-۸) متغیر $\gamma$ یک ثابت مثبت کوچک، و $I$ ماتریس واحد است. حال باقیمانده هر کلاس را می‌توان به صورت معادله (۳-۹) بدست آورد :

$$\begin{aligned}\mathrm{Re}\,sidual_1 &= y - \sum t_{1,i}\,\tilde{d}_{1,i} \\ \mathrm{Re}\,sidual_2 &= y - \sum t_{2,i}\,\tilde{d}_{2,i} \\ &\cdots \\ \mathrm{Re}\,sidual_M &= y - \sum t_{M,i}\,\tilde{d}_{M,i}\end{aligned} \qquad (۳-۹)$$

حال کلاسی برنده است که کمترین باقیمانده را داشته باشد. همان‌طور که دیدیم بردارهای ضرایب را تنک کردیم، اما نه به صورت‌های معمول بلکه از مفهوم نمایش تنک استفاده کردیم. به عبارت دیگر، ما به دنبال یک واژه نامه به اندازه کافی تنک بودیم؛ بنابراین ماتریس واژگان را مجبور کردیم، فقط $M$ کلاس را برای ما نگه دارد؛ سپس در مرحله‌ی آخر نمونه‌ی تست را فقط با این چند کلاس ساختیم. در ادامه خلاصه‌ای از روش پیشنهادی و همچنین در شکل (۳-۱) یک بلوک دیاگرام از این روش را آورده‌ایم.

**۳-۲-۱-۱- خلاصه ای از روش پیشنهادی :**

گام اول – خوشه‌بندی داده‌های آزمون توسط روش K-means

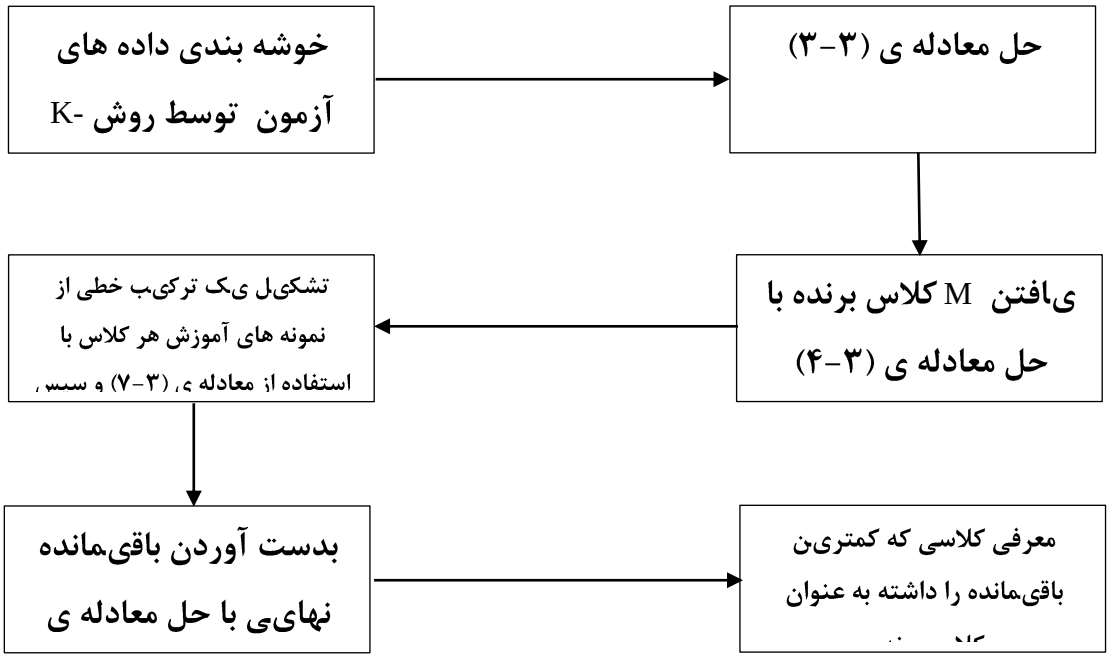
گام دوم – حل معادله ی (۳-۳)

گام سوم – یافتن M کلاس برنده با حل معادله ی (۳-۴)

گام چهارم – تشکیل یک ترکیب خطی از نمونه های آموزش هر کلاس با استفاده از معادله‌ی (۳-۷) و سپس حل معادله‌ی (۳-۸)

گام پنجم – بدست آوردن باقیمانده نهایی با حل معادله ی (۳-۹)

گام ششم – معرفی کلاسی که کمترین باقیمانده را داشته به عنوان کلاس برنده.




شکل ۳-۱- بلوک دیاگرام روش MSSRC

### ۳-۲-۱-۲- تفسیر و توجیه روش پیشنهادی :

این روش را می‌توان به سه مرحله‌ی کلی تقسیم بندی کرد. در مرحله اول، داده‌های آزمون را به خوشه-های مختلفی تقسیم می کنیم. این کار درست مانند آن است که کلمات یک واژه نامه را به ترتیب حروف الفبا مرتب کنیم. حال از هر خوشه یک یا چند پایه را به عنوان نماینده برمی‌گزینیم و در واژه نامه قرار می دهیم. این کار علاوه بر آن که باعث غنای واژه‌نامه خواهد شد، تنک بودن اضافی ضرایب را نیز تضمین می کند. به عبارت دیگر با حذف بعضی اتم‌های غیر ضرور، ضرایب مربوطه را بطور دستی صفر کرده‌ایم. از طرف دیگر می‌دانیم تصویر چهره نیز مانند سایر تصاویر طبیعی شامل تعداد زیادی ساختارهای تکراری است. بنابراین با حذف این اتم ها، اطلاعات زیادی را از دست نخواهیم داد.

از طرف دیگر کاملا منطقی است که فرض کنیم همه‌ی نمونه‌های آموزشی اثر یکسانی روی نمونه آزمون ندارند بلکه کلاس‌هایی که به نمونه‌ی تست نزدیک‌ترند اثر بیشتری روی آن دارند. بنابراین در مرحله دوم ابتدا کلاس‌هایی که از نمونه‌ی آزمون خیلی دورند را حذف کرده و تنها کلاس‌های باقیمانده را در تصمیم گیری در مورد طبقه بندی دخیل می کنیم.

در مرحله‌ی نهایی تنها $M$ کلاس برنده را در تشکیل واژه نامه بکار گرفتیم، آن هم به این نحو که برای هر کدام از کلاس‌ها یک واژه‌نامه مجزا ساختیم. این کار درست مانند آن است که برای یافتن معنای واژه جدید پس از مراجعه به یک واژه‌نامه، صفحه‌ی مربوط به حرف اول واژه ی جدید را یافته و در این صفحات به دنبال کلمات هم معنی با واژه جدید بگردیم. در مورد شناسایی چهره نیز این کار کاملا منطقی است. زیرا پس از آنکه کلاس‌های دور از نمونه تست را حذف کردیم، حال در هر کدام از این کلاس‌ها به دنبال نمونه برنده خواهیم گشت. از سوی دیگر چون در مرحله دوم بسیاری از کلاس‌های دور از نمونه‌ی آزمون را حذف کرده ایم، کار راحت‌تری برای معرفی کلاس برنده داریم.

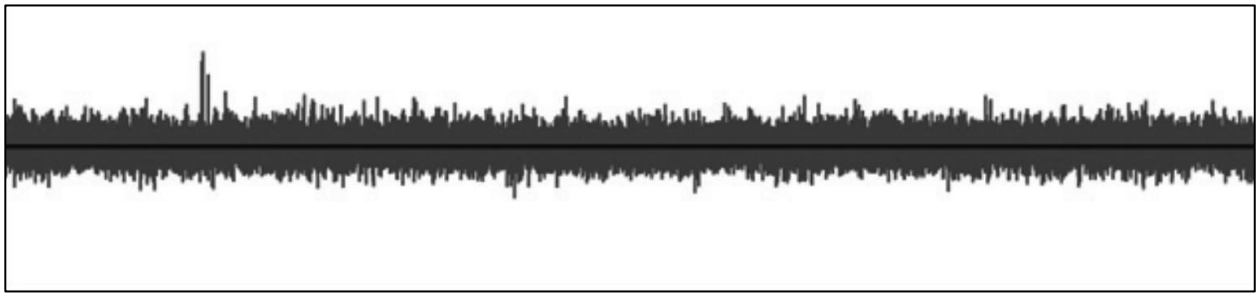
### ۳-۲-۱-۳- ارتباط روش پیشنهادی با روش نمایش تنک :

این روش را می توان روش نمایش تنک نظارتی بهبود یافته خواند. همان طور که دیدیم در این روش از مفهوم تنکی استفاده کردیم. تنک بودن در اینجا به این معنی است که ضرایب بعضی از نمونه‌های آموزش صفر یا نزدیک به صفر است. این ضرایب همان ضرایب تنکی است. و تنکی را در اینجا به معنی صفر شدن بعضی ضرایب در مرحله اول و مرحله‌ی دوم گرفتیم. توضیح بیشتر اینکه در روش‌های رایج نمایش تنک، تنک شدن نمونه‌ها از طریق روش‌های کدینگ حاصل می شود، ولی در اینجا، تنکی از طریق روش گفته شده در مرحله دوم بدست آمد. حذف برخی نمونه‌های آموزش در مرحله‌ی اول نیز به این کار کمک کرد. به عبارت دیگر، ما به دنبال یک واژه‌نامه به اندازه کافی تنک بودیم؛ بنابراین ماتریس واژگان را مجبور کردیم، فقط $M$ کلاس را برای ما نگاه دارد؛ به علاوه اینکه در بسیاری از روش‌های متداول نمایش تنک، ضرایب دقیقا صفر نیستند، بلکه نزدیک به صفراند. اما در این روش، مطمئنیم که ضرایب صفر واقعی خواهند شد.

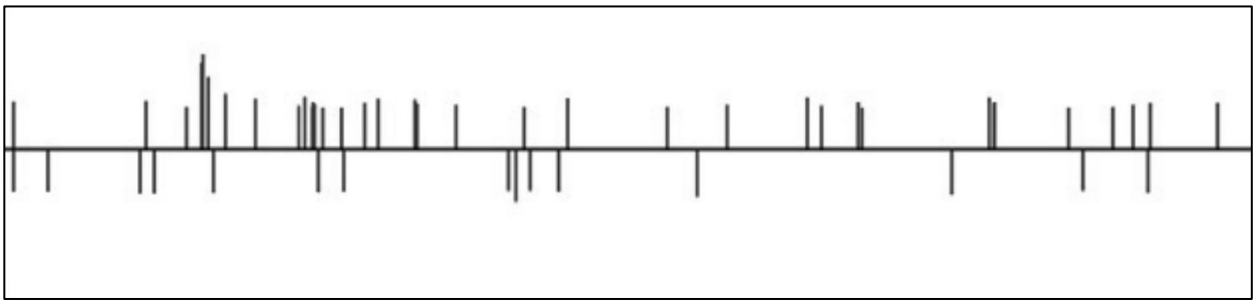
در بخش نتایج تجربی خواهیم دید که اگرچه در این مقاله از ساده‌ترین روش یافتن ضرایب یعنی روش ماتریس شبه معکوس استفاده شده، اما نرخ تشخیص از روش‌های مشابه بهتر است. این روش را می‌توان روش نمایش تنک نظارتی بهبود یافته خواند. لفظ نظارتی به این معنی است که می دانیم چه تعداد و کدام یک از ضرایب صفر هستند. عبارت "بهبود یافته" نیز به این خاطر استفاده شده که از بقیه روش‌های نظارتی بهتر عمل می‌کند.

### ۳-۲-۲- نسخه‌ی سریع روش MSSRC :

یکی از مشکلات روش بالا زمان اجرای الگوریتم است. بنابراین باید به دنبال روشی برای کاهش زمان اجرای الگوریتم باشیم. همانطور که اشاره شد در اینجا از مفهوم روش نمایش تنک استفاده کردیم و نه مستقیما از خود روش نمایش تنک معمول. به همین دلیل بردار ضرایب در مرحله‌ی اول الگوریتم چگال است. در روش نمایش تنک همانطور که در فصل قبل گفته شد، در مرحله‌ی کدینگ، بردار ضرایب را تنک می‌کنیم؛ ولی در اینجا پس از مرحله‌ی دوم است که ضرایب تنک می‌شوند این کار انجام می‌شود. اما همان طور که در شکل‌های پایین نشان می‌دهیم اگرچه با این روش ضرایب را تنک نمی‌کنیم ولی مقادیر بیشینه و کمینه بردار ضرایب، در روش نمایش تنک معمول و همچنین روش پیشنهادی ما در یکجا متمرکز است.



(الف)



(ب)

شکل ۳-۲- مقایسه تنکی ضرایب در مرحله دوم روش MSSRC (شکل الف) با روش SRC (شکل ب).

در این دو شکل، محور افقی زمان، و محور عمودی اندازه را نشان می دهد.

چگال بودن بردار ضرایب پس از مرحله‌ی دوم در روش MSSRC، روند یافتن باقیمانده را طولانی می-کند؛ اما گفتیم که درایه های بیشینه و کمینه‌ی بردارضرایب در هر دو روش تغییر مکان نمی دهند، پس بجای یافتن باقیمانده، ضرایب مربوط به هر کلاس را یافته و سهم هر کلاس در بردار ضرایب را با جمع همه‌ی درایه های مربوط به یک کلاس حساب می‌کنیم. کلاسی برنده است که سهم بیشتری داشته باشد. به عبارت دیگر در گام سوم از الگوریتم پیشنهادی برای یافتن $M$ کلاس برتر، بجای استفاده از کمترین باقیمانده‌ها، بیشترین سهم کلاس‌ها از ضرایب را می‌یابیم. بدین ترتیب حجم محاسبات کاهش یافته و سرعت اجرای الگوریتم افزایش می‌یابد.

### ۳-۲-۳- روش MBP[۳۳] :

بخاطر هزینه بالای محاسباتی در هنگام کار با پایگاه های داده‌ی بزرگ، عملاً نمی‌توان از روش نرم L1 استفاده کرد. به همین علت باید روشی را برای پایین آوردن محاسبات و در نتیجه استفاده از مزایای نرم L1 بکار بست.

در این روش بجای آنکه مستقیماً از بهینه سازی نرم L1 استفاده کنیم، ابتدا در دو مرحله ماتریس واژه-نامه را تا جای ممکن کوچک کرده و سپس از نرم L1 استفاده می‌کنیم. تفصیل بیشتر اینکه در مرحله‌ی اول تمام داده‌های آزمون را به M بخش مجزا تقسیم می‌کنیم. در اینجا برای هر کدام از این M بخش، یک ماتریس واژگان تشکیل می دهیم. به عبارت دیگر به جای آنکه یک ماتریس واژه‌نامه را با همه‌ی نمونه‌های آموزشی بسازیم، به ازای هرکدام از بخش‌ها یک ماتریس واژه‌نامه می سازیم؛ سپس نمونه آزمون را به صورت ترکیب خطی هر کدام از این ماتریس‌های واژه نامه بیان می کنیم.

$$y_1 \approx a_1 t_1 + \ldots + a_n t_n \qquad (۱۰-۳)$$

در اینجا y نشان‌دهنده نمونه آزمون، t نمایانگر نمونه‌های آموزشی، a نشانگر ضرایب نمونه‌های آموزش و n تعداد نمونه‌های هر بخش است. اگر معادله‌ی تقریبی (۳-۱۰) را به صورت معادله (۳-۱۱) بازنویسی کنیم:

$$Y = GA \qquad (۱۱-۳)$$

[33] Modified Basis Pursuit

که در اینجا $A^T=[a_1,...,a_r]$ و $G=[t_1,...,t_r]$ است. برچسب r به معنی تعداد نمونه های آموزش در هر بخش است. همان طور که گفته شد در مرحله ی اول، برای هر کدام از بخش‌ها، یک ماتریس واژه‌نامه می‌سازیم. بنابراین بهتر است معادله ی (۳-۱۱) را به صورت دسته معادلات (۳-۱۲) بیان کنیم:

$$\begin{aligned} Y &= G_1 A_1 \\ Y &= G_2 A_2 \\ &.... \\ Y &= G_M A_M \end{aligned} \qquad (۱۲-۳)$$

M در اینجا تعداد بخش‌هاست. آنگاه می‌توان بردارهای A را توسط معادلات (۳-۱۳) بدست آورد :

$$\begin{aligned} \tilde{A_1} &= (G_1{}^T G_1 + \gamma I)^{-1} G_1{}^T y \\ \tilde{A_2} &= (G_2{}^T G_2 + \gamma I)^{-1} G_2{}^T y \\ &.... \\ \tilde{A_M} &= (G_M{}^T G_M + \gamma I)^{-1} G_M{}^T y \end{aligned} \qquad (۱۳-۳)$$

در رابطه (۳-۱۳) متغیر $\gamma$ یک ثابت مثبت کوچک، و I ماتریس واحد است. حال در هر کدام از بردارهای ضرایب، K ضریبی که دارای بالاترین مقدار باشد را اخذ خواهیم کرد.

در مرحله‌ی دوم تمام نمونه‌های مرتبط با K ضریب برنده در هر کدام از بردارهای A را داخل یک ماتریس واژگان جدید جایگذاری خواهیم کرد و طبق معادله‌ی (۳-۱۴) سعی می‌کنیم یک ترکیب خطی از تمام نمونه‌های آموزشی برنده را برای هر کدام از نمونه‌های آزمون تعیین کنیم.

$$y \approx d_1 x_1 + d_2 x_2 + ... + d_b x_b \qquad (۱۴-۳)$$

در هر صورت نمی توان نمونه‌ی آزمون را بطور دقیق با نمونه‌های آموزش ساخت. برای همین معادله (۳-۱۴) را بصورت تقریبی نوشتیم. در اینجا y نشان‌دهنده نمونه آزمون، d نمایانگر نمونه‌های آموزشی، x نشانگر ضرایب نمونه‌های آموزش و b تعداد کل نمونه‌های برنده است. اگر معادله‌ی تقریبی (۳-۱۴) را به صورت معادله (۳-۱۵) بیان کنیم :

$$Y = DX \qquad (۱۵-۳)$$

که در این معادله $X^T = [x_1, x_2, x_3, ... x_b]$ و $D = [d_1, d_2, d_3, ..., d_b]$ است؛ و هرکدام از ستون‌های واژه-نامه D برابر با یکی از تصاویر نمونه‌های آموزش است. برچسب b نیز تعداد همه ی نمونه های آموزش برنده در مرحله‌ی اول است. آنگاه می‌توان بردار ضرایب X را به صورت معادله (۳-۱۶) بدست آورد :

$$\tilde{X} = (D^T X + \mu I)^{-1} D^T y \qquad (۱۶-۳)$$

در رابطه (۳-۱۶) متغیر $\mu$ یک ثابت مثبت کوچک و I ماتریس واحد است. عبارت $\mu I$ را برای جلوگیری از بد حالت شدن ماتریس معکوس اضافه کرده‌ایم.

از این به بعد مرحله‌ی سوم الگوریتم آغاز خواهد شد. در این مرحله با استفاده از بردار ضرایب بدست آمده در مرحله‌ی دوم و همچنین ماتریس واژگان کوچکتر شده طبق معادله‌ی (۳-۱۷)، مسئله نرم L1 را برای بدست آوردن بردار ضرایب بهینه حل خواهیم کرد.

$$\hat{X} = \arg\min \| y - DX \|_2 + \lambda \| \alpha \|_1 \qquad (۱۷-۳)$$

حال طبق معادله‌ی (۳-۱۸) باقیمانده را برای هر کلاس محاسبه خواهیم کرد.

$$\begin{aligned} \mathrm{Re}\, sidual_1 &= y - \sum x_{1,i}\, d_{1,i} \\ \mathrm{Re}\, sidual_2 &= y - \sum x_{2,i}\, d_{2,i} \\ &.... \\ \mathrm{Re}\, sidual_M &= y - \sum x_{M,i}\, d_{M,i} \end{aligned} \qquad (۱۸-۳)$$

کلاسی برنده است که کمترین باقیمانده را داشته باشد. در ادامه خلاصه‌ای از روش پیشنهادی و همچنین در شکل (۳-۲) یک بلوک دیاگرام از این روش را آورده‌ایم.

### ۳-۲-۳-۱- خلاصه ای از روش پیشنهادی :

گام اول – تقسیم داده های آزمون به M بخش

گام دوم – حل معادلات (۳-۱۳)

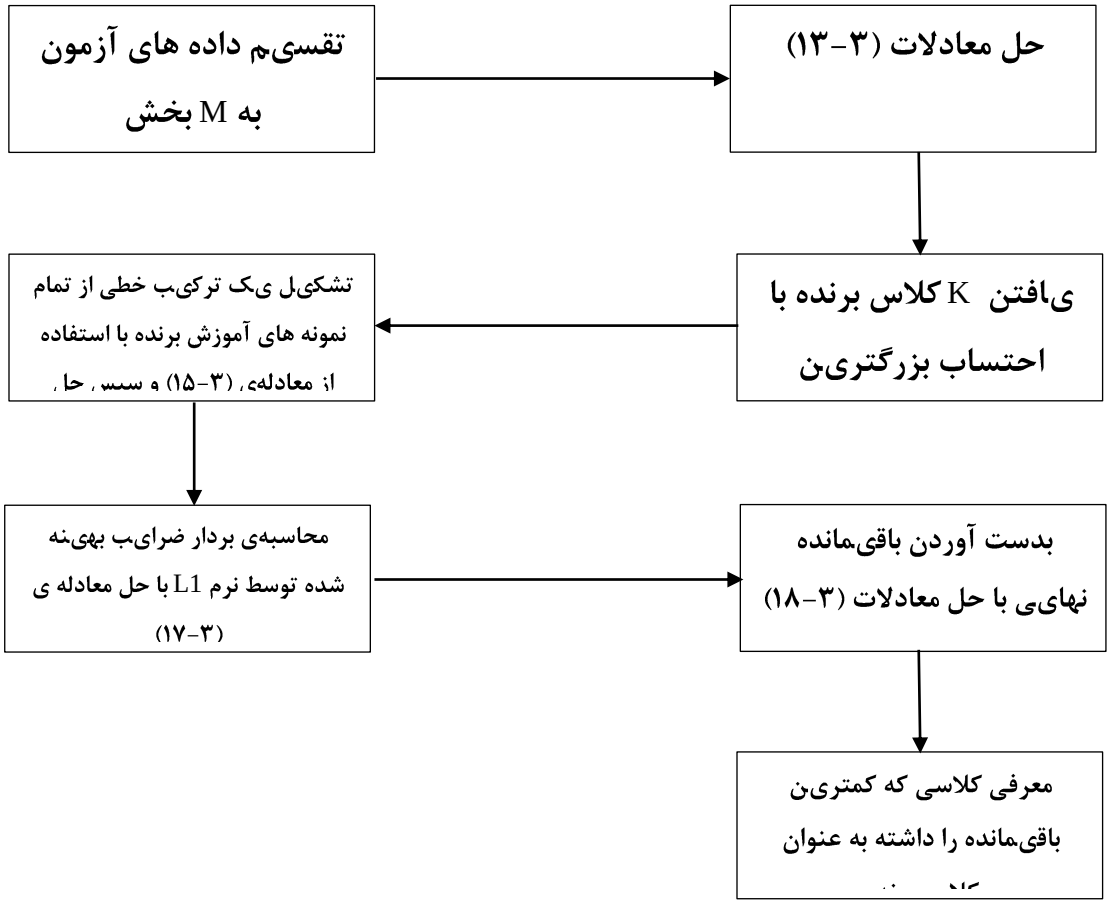
گام سوم – یافتن K کلاس برنده با احتساب بزرگترین ضرایب هر بخش

گام چهارم – تشکیل یک ترکیب خطی از تمام نمونه های آموزش برنده با استفاده از معادله‌ی (۳-۱۵) و سپس حل معادله ی (۳-۱۶)

گام پنجم – محاسبه‌ی بردار ضرایب بهینه شده توسط نرم L1 با حل معادله ی (۳-۱۷)

گام ششم – بدست آوردن باقیمانده نهایی با حل معادلات (۳-۱۸)

گام هفتم – معرفی کلاسی که کمترین باقیمانده را داشته به عنوان کلاس برنده.




شکل ۳-۳- بلوک دیاگرام روش MBP

## ۳-۳- معرفی ویژگی‌های مورد استفاده:

همان طور که گفته شد فهم صحیح از ذات ویژگی و همچنین درک درست از الگوریتم مورد نظر منجر به جواب دلخواه خواهد شد. ما در این بخش سعی خواهیم کرد، ویژگی‌های متنوعی را معرفی کنیم و از بطن آنها راهی برای حل بهتر مسئله بیابیم.

برخی مقالات و مراجعی که از نمایش تنک به عنوان طبقه‌بند استفاده کرده بودند[۳۲]، پیکسل های خام را به عنوان ویژگی مورد استفاده قرار داده بودند. البته بعضی اوقات پیکسل‌های خام نتایج بهتری را به همراه دارد و نیز همان‌طور که در[۳۲] نشان داده شده یکی از مزایای روش نمایش تنک این است که ویژگی‌های مختلف تاثیر زیادی روی جواب نهایی نخواهد داشت؛ اما همان‌طور که در فصل بعد خواهیم دید حداقل در مورد موضوع شناسایی چهره استفاده از ویژگی‌های مختلف تاثیر زیادی روی جواب نهایی خواهد داشت. مسئله‌ی دیگر این است که استفاده از پیکسل‌های خام زمان‌بر است و در نتیجه روند حل مسئله را کند خواهد کرد. بنابراین هنگامی که زمان برای ما اولویت دارد بهتر است از پیکسل‌های خام استفاده نکنیم. ما در این فصل از ویژگی‌های تبدیل ویولت و گرادیان هیستوگرام استفاده کردیم.

### ۳-۳-۱- تبدیل موجک

در روش MSSRC از پیکسل‌های خام به عنوان ویژگی استفاده کردیم. اما گفتیم، در مسائلی که زمان برای ما مهم است، نباید از پیکسل‌های خام استفاده کرد. از سوی دیگر یکی از مسائل چالش برانگیز در دنیای علم این بوده است که واقعا مغز انسان چه چیز را می بیند؟ گفتیم روش نمایش تنک نیز با الهام گرفتن از این موضوع که تعداد کمی از نرون‌های مغزی در هنگام دیدن فعال می‌شوند، پدید آمد. اما یکی دیگر از دیدگاه‌ها که در بسیاری از منابع از جمله در [۳۳] آمده است، این است که مغز به دید فرکانسی به جهان نگاه می‌کند. تفصیل این موضوع خارج از حوصله‌ی این بحث است. ما به همین مقدار کفایت می‌کنیم که بر اساس بسیاری از مشاهدات، ادعا شده است که آن چه مغز درک می‌کند، نه آن چیزی است که چشم می‌بیند. بلکه چشم در حوزه زمان و مغز در حوزه فرکانس کار می‌کند. نرون‌های مبدل، وظیفه‌ی تبدیل این داده‌های مختلف را بر عهده دارند.

با توجه به این موضوع بر آن شدیم که از ضرایب تبدیل موجک به عنوان ویژگی استفاده کنیم. استخراج ضرایب تبدیل موجک از یک طرف باعث کاهش ویژگی و درنتیجه افزایش سرعت الگوریتم خواهد شد. همچنین با توجه به قرابتی که بین ضرایب تبدیل موجک و ویژگی های استخراج شده توسط مغز وجود دارد، دقت نیز افزایش خواهد یافت.

چند روش مختلف را برای استفاده از این ویژگی استفاده کرده‌ایم. مبنای کلی روش‌های استفاده شده بر دو اصل استوار بوده است:

۱- مغز برای دیدن و تشخیص چهره بیشتر از اطلاعات فرکانس پایین استفاده خواهد کرد.

۲- از آنجا که مغز، عمل تشخیص چهره را با حداقل نرون ها انجام می دهد، در اینجا نیز از حداقل ویژگی ها استفاده شود.

در روش اول برای استخراج ویژگی از تصاویر چهره، ضرایب تبدیل موجک را گرفته و ماتریس‌های حاصل را نرمال می کنیم. حال دو ماتریس جزئیات افقی و عمودی را در دو ماتریس جداگانه جای می‌دهیم. جزئیات قطری را در فرآیند استخراج ویژگی دخیل نکردیم؛ چون با سعی و خطا مشاهده شد که تاثیر زیادی روی نرخ تشخیص ندارد. از آنجا که قرار است کمترین و در عین حال مهم ترین ویژگی‌ها را نگه داریم، ماتریس‌های با جزئیات افقی و عمودی را با هم مقایسه کرده و آرایه‌های با شدت بیشتر را در ماتریس جدیدی حفظ می‌کنیم و بقیه‌ی آرایه‌ها را حذف می‌کنیم. حال این ماتریس حاوی فرکانس‌های بالا را به ماتریس حاوی فرکانس‌های پایین الحاق می کنیم. مشاهده شد که با استخراج این ویژگی‌ها، تنها مرحله‌ی سوم الگوریتم MSSRC برای گرفتن یک جواب خوب کافیست. یعنی با استحصال ویژگی‌ها بدین صورت، نیازی به اجرای هر سه مرحله‌ی الگوریتم نیست؛ و فقط با اجرای مرحله‌ی سوم الگوریتم گفته شده به جواب خوبی خواهیم رسید.

شرح بیشتر این روش بدین گونه است که پس از استخراج ویژگی به روش توضیح داده شده، کافیست برای هرکدام از کلاس‌ها یک واژه‌نامه منحصر به فرد شامل کلیه نمونه‌های آموزش از هر کلاس تشکیل داده و ضرایب را با استفاده از این واژه‌نامه ها بدست آوریم حال باقیمانده‌ی حاصل از تفریق نمونه‌ی آزمون و واژه‌نامه وزن‌دار شده، کلاس برنده را مشخص می‌سازد.

در روش دوم پیشنهادیِ استخراج ویژگی توسط ضرایب تبدیل موجک، تاثیر ماتریس‌های با فرکانس بالا و پایین را با هم مقایسه خواهیم کرد. روش کار بدین گونه است که پس از استخراج ضرایب تبدیل موجک از تصاویر و نرمال کردن آنها، مثل روش استخراج ویژگی در بالا، ماتریس های جزئیات عمودی و جزئیات افقی را مقایسه کرده، آرایه های با شدت بیشتر از دو ماتریس را حفظ می‌کنیم. سپس این آرایه ها را در یک ماتریس جدید جایگذاری می کنیم. حال دو ماتریس ویژگی داریم. اول ماتریس حاوی فرکانس های پایین و دوم ماتریس حاوی شدیدترین آرایه ها از دو ماتریس با جزئیات عمودی و جزئیات افقی.

حال بطور مستقل، هر کدام از این ماتریس‌ها را به عنوان ویژگی به روش MSSRC پیشنهادی اعمال خواهیم کرد. هر کدام از ورودی ها که باقیمانده کمتری داشته باشد، نشان دهنده‌ی کلاس مورد نظر است.

### ۳-۳-۲- گرادیان هیستوگرام

این ویژگی در [۳۴] توسط دلال و دیگران معرفی شده است. برای استخراج این ویژگی، هر تصویر ورودی را به $W*W$ بلوک مساوی تقسیم می‌کنیم. این بلوک‌ها را سلول می نامیم. در هر سلول یک هیستوگرام گرادیان با $n$ بین محاسبه می شود. برای محاسبه‌ی هیستوگرام گرادیان، بصورت زیر عمل می کنیم:
ابتدا تصویر با استفاده از کرنل های سوبل در جهت $x$ و $y$ فیلتر می‌شود تا گرادیان تصویر در راستای $x$ و $y$ بدست آید.

$$G_x = I * D_x \qquad (۱۹-۳)$$

$$G_y = I * D_y \qquad (۲۰-۳)$$

در معادلات بالا، I تصویر اصلی، $D_x$ و $D_y$ کرنل های سوبل، $G_x$ و $G_y$ گرادیان تصویر در راستای x و y را نشان می‌دهند. سپس اندازه و جهت گرادیان در هر پیکسل به صورت زیر بدست می آید:

$$|G(i,j)| = \sqrt{(G_x(i,j))^2 + (G_y(i,j))^2} \qquad \theta_G(i,j) = \tan^{-1}\left(\frac{G_y(i,j)}{G_x(i,j)}\right) \qquad (۲۱-۳)$$

حال برای محاسبه‌ی هیستوگرام گرادیان در هر سلول، فاصله‌ی بین ۰-۳۶۰ درجه را به n فاصله‌ی مساوی تقسیم می کنیم، که n تعداد جهت های گرادیان یا همان بین های هیستوگرام را نشان می دهد و هر کدام از این فاصله ها یک کانال هیستوگرام را تشکیل می دهد. برای محاسبه‌ی هیستوگرام، هر پیکسل داخل سلول بر مبنای زاویه‌ی گرادیانش به یکی از کانال های هیستوگرام رای می‌دهد. این رای‌ها بر اساس اندازه گرادیان در هر پیکسل وزن‌دار می شود. پس از محاسبه‌ی هیستوگرام گرادیان در هر سلول، این بردارها را در کنار هم و تحت یک بردار مستقل می‌چینیم[۳۵]. این بردار نهایی را بردار هیستوگرام گرادیان تصویر می‌گوییم. از این ویژگی هم در این پایان‌نامه استفاده شده که نتایج آن را در فصل بعد مشاهده خواهیم کرد.

### ۳-۴- تشخیص چهره بدون استفاده از نمایش تنک

این روش ترکیبی از روش‌های مبتنی بر مدل و مبتنی بر ظاهر است. از ابتدا سعی ما بر این بود که هم از مزایای روش‌های مبتنی بر ظاهر و هم از حسنات روش‌های مبتنی بر مدل استفاده کنیم. قبل از آنکه به الگوریتم نهایی برسیم، الگوریتم‌های مختلفی را هم آزمایش کردیم که بعضی از آنها با شکست مواجه شد و از بعضی دیگر ایده گرفتیم و در الگوریتم نهایی از آنها استفاده کردیم. یکی از الگوریتم هایی که مورد بررسی قرار دادیم، این بود که ابتدا نواحی مهم چهره را استحصال کرده، سپس از آنها استخراج ویژگی کردیم. پس از آنکه نواحی چشم، بینی و لب را معین کردیم، آنگاه روی آن‌ها گرادیان هیستوگرام را اعمال کردیم. برداری که نهایتا بدست آمد، حاوی سه بردار حاصل از اعمال گرادیان هیستوگرام روی این نواحی بود که پشت سر هم گذاشته شدند. اما این الگوریتم جواب دلخواه‌مان را به همراه نداشت. نهایتاً و پس از بررسی چند الگوریتم دیگر به روشی که در ادامه خواهد آمد، رسیدیم.

روش پیشنهادی ما در واقع تلفیقی از روش‌های مبتنی بر مدل و مبتنی بر ظاهر است. به عبارت دیگر برای آنکه اطلاعات کلی چهره را از دست ندهیم از ویژگی‌های میانگین بلوکی و هیستوگرام گرادیان روی کل چهره استفاده کردیم و از دیگر سو برای آنکه از اطلاعات نواحی مهم چهره که ویژگی‌های مهمی از چهره در آنها انباشته شده غفلت نکرده باشیم، از ویژگی‌های SIFT و روش ASM بهره بردیم. مشروح این الگوریتم را در ادامه از نظر خواهیم گذراند.

در مرحله اول، ویژگی‌های کلی چهره را استخراج می‌کنیم. برای این کار از روش میانگین بلوکی و روش گرادیان هیستوگرام استفاده کردیم. ویژگی مهم دیگری که البته در بعضی از پایگاه‌های داده عملکرد مناسبی دارد، ویژگی‌های هندسی صورت است. طبق [۱۶] و [۳۶] و بر اساس تجربیات، ویژگی‌های هندسی ذیل را برای این پروژه استفاده کردیم:

الف - نسبت طول به عرض بینی

ب – نسبت طول به عرض سر

ج - فاصله دو نقطه واقع در گوشه های بیرونی دو چشم به دو نقطه درونی چشم‌ها

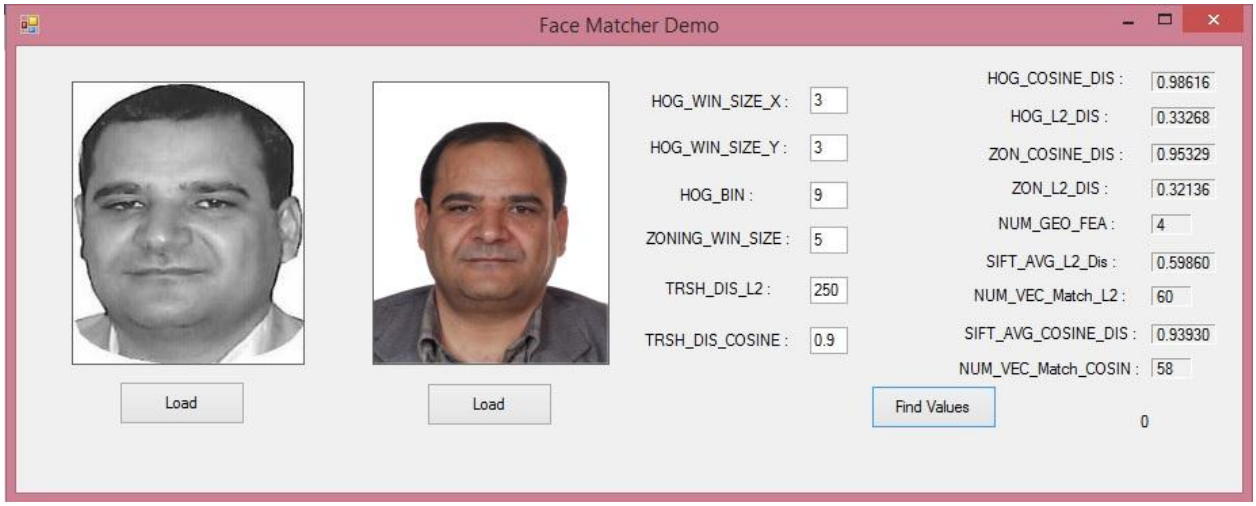
د - نسبت فاصله‌ی پایین‌ترین نقطه صورت روی فک تا نقطه وسط چشم به فاصله‌ی پایین‌ترین نقطه صورت روی فک تا نقطه پایینی روی لب

ه - نسبت طول به عرض لب

سپس با استفاده از کتابخانه IntraFace که از روش ASM برای یافتن نقاط مهم چهره استفاده می-کند، ۷۶ نقطه کلیدی چهره را بدست می‌آوریم. ۱۶ نقطه از این نقاط، نقاط اطراف صورت هستند که به

تجربه ثابت شد، بهتر است از این نقاط صرف نظر شود. وجود نویزی که در اطراف صورت (بدلیل تاثیر محیط) وجود دارد، استفاده از این نقاط مهم را محدود می‌کند.

حال از ۶۰ نقطه‌ی مهم باقیمانده، ویژگی SIFT را استخراج می‌کنیم. برای این کار کدی نوشتیم که عملگر SIFT مختصات نقطه مورد نظر را از کتابخانه Intraface گرفته و آنگاه روی این نقاط عملگر SIFT را بیابد. این ویژگی به ازای هر نقطه مهم، ۱۲۸ ویژگی ارائه می‌دهد. توضیحات بیشتری در مورد این عملگر در ادامه خوهد آمد. نمایی از این برنامه نرم افزاری را در شکل ۳-۴ آورده‌ایم.




شکل ۳-۴- نمایی از برنامه نرم افزاری

لازم به ذکر است که کد این برنامه به زبان C++ و با استفاده از کتابخانه OpenCV نوشته شده است. در ادامه و به عنوان آخرین بخش از این فصل معرفی مختصری از SIFT را ارائه می‌دهیم.

## ۳-۵- ویژگی SIFT

توصیف‌گر SIFT امروزه به عنوان یکی از بهترین و قدرتمندترین ابزارها برای استخراج نقاط کلیدی غیر حساس به شرایط مختلف مانند چرخش، بزرگنمایی، تغییر نمای دید، نویز، نورپردازی و تبدیل کشیدگی است[۳۷]. به طور کلی مراحل استفاده از این توصیفگر را می توان به ۳ قسمت اصلی زیر تقسیم نمود:

### ۳-۵-۱- یافتن نقاط کلیدی:

مرحله‌ی اول در تمامی روش‌هایی که روی نقاط خاصی از تصویر کار می کنند، یافتن آنها است. در این روش برای یافتن نقاط کلیدی در تصویر از تفاوت‌های گاوسین (DoG) استفاده شده است. فرآیند یافتن این نقاط، با ساخت یک هرم از تصاویر و کانولوشن تصویر $I(x,y)$ با فیلتر گاوسین $G(x,y,\sigma)$ شروع می‌شود. بنابراین فضای مقیاسی بصورت زیر نمایش داده می شود:

$$L(x,y,\sigma) = I(x,y) * G(x,y,\sigma) \qquad (۲۲-۳)$$

G را بصورت زیر تعریف می‌کنیم:

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \qquad (۲۳-۳)$$

میزان بلور بودن با پارامتر انحراف استاندارد $\sigma$ در تابع گاوسین کنترل می شود. فضای مقیاسی DoG با تفریق سطوح مقیاسی مجاور حاصل می شود:

$$D(x,y,\sigma) = [G(x,y,k,\sigma) - G(x,y,\sigma)] * I(x,y) \qquad (۲۴-۳)$$

حال داریم :

$$D(x,y,\sigma) = L(x,y,k,\sigma) - L(x,y,\sigma) \qquad (۲۵-۳)$$

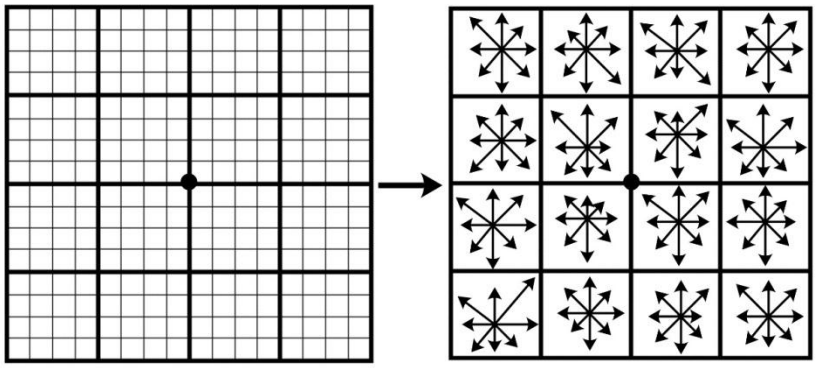
در شکل ۳-۵ مراحل ساخت DoG را آورده‌ایم.

شکل ۳-۵- مراحل ساخت DoG

حال باید نقاط ماکسیمم یا مینیمم در هر اکتاو را پیدا کنیم. این کار با مقایسه هر پیکسل با همسایه‌های ۲۶گانه در ناحیه ۳*۳ تمامی سطوح DoG مجاور در همان اکتاو انجام می‌گیرد. اگر نقطه‌ی مورد نظر بزرگتر یا کوچکتر از تمامی همسایگانش بود به عنوان نقطه‌ی مورد نظر انتخاب می شود.

### ۳-۵-۲- نمایش توصیفگر نقاط کلیدی:

در این مرحله بردار ویژگی اصلی ایجاد خواهد شد. در ابتدا دامنه گرادیان و جهت در اطراف نقطه کلیدی نمونه برداری می شود. اگر از آرایه ۴*۴ با ۸ جهت در هر هیستوگرام استفاده کنیم، طول بردار ویژگی، ۱۲۸ عنصر برای هر نقطه ویژگی خواهد بود.



بردار ۱۲۸ بعدی  پنجره ۱۶*۱۶

شکل ۳-۶- نمونه‌ای از نقاط استخراج شده توسط SIFT [۳۸]

### ۳-۵-۳- تطبیق بردارهای ویژگی

فاز تطبیق در مرحله تشخیص، با مقایسه هر یک از نقاط کلیدی استخراج شده از تصویر تست با مجموعه نقاط کلیدی مربوط به تصویر آموزشی انجام می گیرد. بهترین نقاط کاندید برای تطبیق، از طریق تشخیص نزدیکترین همسایه در مجموعه نقاط کلیدی تصویر آموزشی یافت می شوند. نزدیکترین همسایه دارای کمترین فاصله با نقطه مطابقش است.

## ۳-۶- جمع‌بندی

در این فصل روش‌های پیشنهادی مختلفی را بررسی کردیم. ضمن این الگوریتم‌ها، ویژگی‌های متنوعی را نیز برشمردیم. گفتیم ادغام برخی الگوریتم‌ها و روش‌ها منجر به نتایج خوبی خواهد شد و بر همین اساس ادغام روش‌های MSSRC و MBP را با ویژگی‌هایی از قبیل تبدیل ویولت و گرادیان هیستوگرام بررسی کردیم. سپس گزارشی از یک الگوریتم تشخیص چهره بدون استفاده از نمایش تنک را ذکر کردیم؛ و نهایتا برای تکمیل بحث توصیف‌گر SIFT را به اختصار توضیح دادیم.

در فصل بعد نشان خواهیم داد روش‌های پیشنهادی در عمل هم نتایج خوبی به همراه داشته و در آخر پیشنهاداتی را برای کارهای آینده ارائه می‌دهیم.

# فصل چهارم :

# نتا یج  پیاده  سازی

## ۴-۱- مقدمه

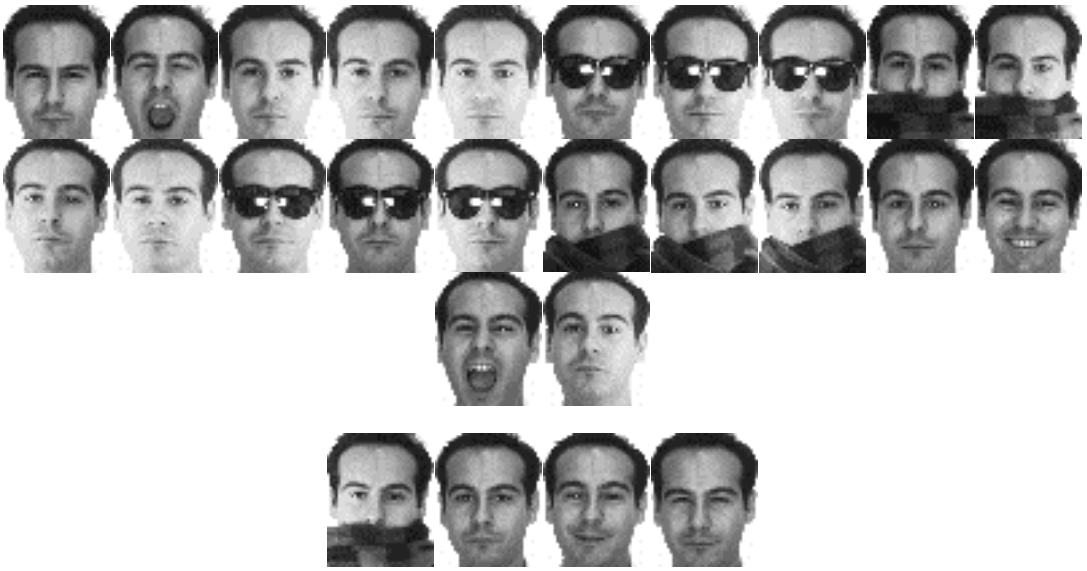
در این فصل نتایج شبیه سازی الگوریتم‌های پیشنهادی روی چند پایگاه داده را می‌آوریم. ابتدا معرفی مختصری از پایگاه‌های داده‌ی مورد استفاده را آورده و سپس نتایج شبیه سازی‌ها را در حالات مختلف نشان خواهیم داد.

## ۴-۲- معرفی پایگاه داده :

در این بخش دو پایگاه داده AR و ORL را معرفی می‌کنیم. این دو پایگاه داده در اغلب مقالاتی که روی موضوع تشخیص چهره کار کرده‌اند، مورد استفاده قرار گرفته است.

### ۴-۲-۱- پایگاه داده AR :

پایگاه داده AR یک پایگاه داده با شرایط کنترل شده است[۳۹]. تصاویر این پایگاه داده در دو دوره‌ی مختلف با فاصله زمانی دو هفته گرفته شده است. در دوره‌ی اول ۱۳۵ شخص (شامل ۷۶ مرد و ۵۹ زن) و در دوره دوم، از ۱۳۵ شخص حاضر در دوره اول، ۱۲۰ شخص (شامل ۶۵ مرد و ۵۵ زن) حضور دارند. در هر دوره ۱۳ تصویر از هر شخص و تحت شرایط کنترل شده گرفته شده است. تصاویر بصورت تمام رخ و شامل تغییرات روشنایی، تغییرات حالت چهره و موانعی مانند عینک آفتابی و شال گردن هستند. تعداد کل تصاویر موجود در پایگاه داده، ۳۳۱۵ تصویر و با اندازه ۵۷۶ X ۷۶۸ پیکسل است. این تصاویر به صورت رنگی گرفته شده است. در شکل ۴-۱ نمونه ای از تصاویر این پایگاه داده برای یک شخص خاص نشان داده شده است.



شکل ۴-۱- نمونه‌ای از تصاویر پایگاه داده AR

**۴-۲-۲- پایگاه داده ORL :**

پایگاه داده ORL شامل تصاویری از چهره‌های مختلف است که در فاصله‌ی بین سال‌های ۱۹۹۲ تا ۱۹۹۴ و توسط آزمایشگاه کامپیوتر دانشگاه کمبریج گرفته شده است. این مجموعه شامل ۴۰۰ تصویر چهره از نمای جلو از ۴۰ شخص، و در زوایای مختلف است. تمام تصاویر در ابعاد ۹۲×۱۱۲ گرفته شده است. برای برخی افراد تصاویر با فاصله‌ی زمانی و تغییرات در حالت عینک گرفته شده است. کلیه تصاویر با یک پس زمینه‌ی تیره و یکدست تصویر برداری شده اند. در شکل ۴-۲ نمونه ای از تصاویر این پایگاه داده برای یک شخص خاص نشان داده شده است[۴۰].

شکل۴-۲- نمونه‌ای از تصاویر پایگاه داده ORL

### ۴-۳- نتایج شبیه سازی :

در این بخش ابتدا نتایج شبیه سازی روش MSSRC را روی پایگاه داده AR می‌آوریم. سپس نتایج این روش را روی پایگاه داده ORL بررسی کرده و در انتها نتایج شبیه سازی روش MBP را بررسی می-کنیم.

### ۴-۳-۱- روش MSSRC :

ابتدا نتایج شبیه سازی روش MSSRC را روی پایگاه داده AR می‌آوریم. در اولین آزمایش از پیکسل-های خام تصویر به عنوان ویژگی استفاده می‌کنیم. نتایج این شبیه سازی در جدول ۴-۱ آمده است. این آزمایش را با سه مقدار مختلف طول بردار ویژگی شبیه سازی کرده‌ایم. از کل تصاویر مجموعه AR، حدود ۳۵٪ یعنی ۹ نمونه به عنوان آموزش و حدود ۶۵٪ یعنی ۱۷ نمونه به عنوان آزمایش انتخاب شده است. در آزمایش دوم از ویژگی گرادیان هیستوگرام و در آخرین آزمایش از تبدیل موجک استفاده کرده ایم. در جداول ۴-۲ و ۴-۳ نتایج سایر شبیه سازی‌ها را آورده‌ایم.

جدول ۴-۱- نتایج شبیه سازی روش MSSRC روی پایگاه AR با استفاده از ویژگی پیکسل‌های خام

| | | | |
|---|---|---|---|
| طول بردار ویژگی | ۱۳۰ | ۵۰۰ | ۲۰۰۰ |
| درصد بازشناسی | ٪ ۹۲/۷۹ | ٪ ۹۶/۳۱ | **٪ ۹۷/۰۶** |

جدول ۴-۲- نتایج شبیه سازی روش MSSRC روی پایگاه AR با استفاده از ویژگی گرادیان هیستوگرام

| | | | | |
|---|---|---|---|---|
| طول بردار ویژگی | ۱۶۲ | ۴۵۰ | ۱۷۶۴ | ۲۹۱۶ |
| درصد بازشناسی | ٪ ۸۵/۴۴ | ٪ ۹۴/۷۵ | ٪ ۹۸/۳۳ | **٪ ۹۹/۲۲** |

جدول ۴-۳- نتایج شبیه سازی روش MSSRC روی پایگاه AR با استفاده از ویژگی تبدیل موجک

| | | | | |
|---|---|---|---|---|
| طول بردار ویژگی | ۵۰۰ | ۶۴۴ | ۸۶۴ | ۲۶۴۶ |
| درصد بازشناسی | ٪ ۹۷/۵۰ | **٪ ۹۸/۵۳** | ٪ ۹۸/۴۸ | ٪ ۹۷/۹۹ |

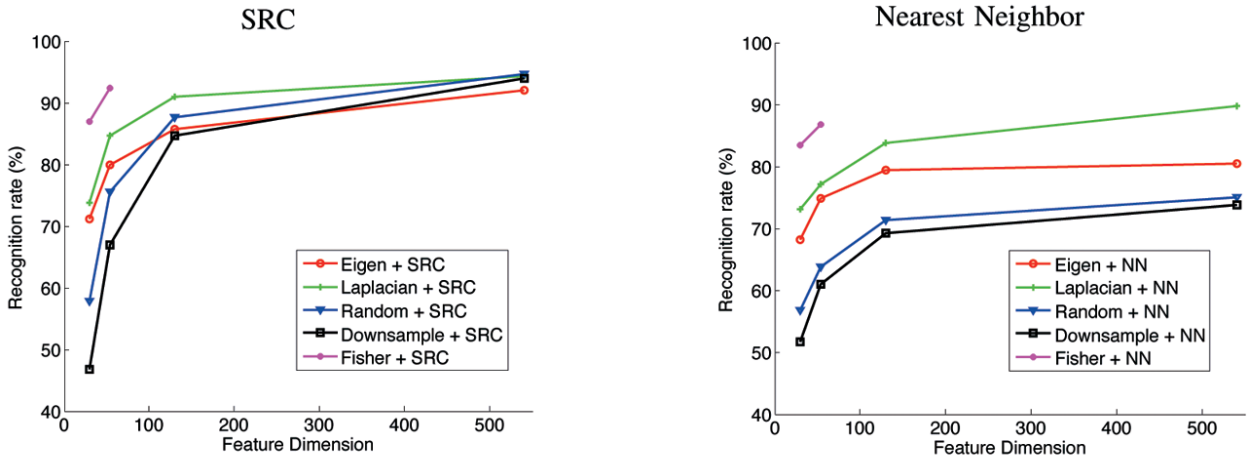
همان طور که از جداول مشخص است، بهترین نتیجه به ازای ویژگی گرادیان هیستوگرام حاصل شده است. این ویژگی به دلیل عدم حساسیت به تغییرات روشنایی و حساسیت به لبه های محلی که مهمترین اطلاعات تصاویر را در خود دارند از سایر ویژگیها پاسخ بهتری می‌دهد.

در شکل ۴- ۳ نمونه ای از تصاویری که با هیچ کدام از ویژگی‌های گفته شده درست تشخیص داده نشده-اند را آورده‌ایم. همان‌طور که مشخص است، این دو تصویر شباهت زیادی به هم دارند، و تشخیص اشتباه این دو تصویر توسط هر الگوریتمی دور از ذهن نیست.

شکل ۴-۳- دو نمونه از تصاویری که اشتباه تشخیص داده شده‌اند.

شکل‌های سمت راست نمونه‌های آزمایش هستند.

در شکل ۴-۵ نتایج روش‌های NN و SRC با استفاده از روش های ویژگی مختلف روی پایگاه داده AR نشان داده شده است. همان طور که مشهود است، در روش SRC و با بردار ویژگی به طول ۵۴۰، نرخ تشخیص ٪ ۹۴/۷ و در روش NN درصد بازشناسی ٪ ۸۹/۷ بدست آمده است[۴۱]. همان‌طور که در جداول بالا مشهود است، روش پیشنهادی نتیجه‌ی بهتری را ارائه داده است.




شکل ۴-۵- درصد بازشناسی روی پایگاه داده AR با استفاده از طبقه بندها و ویژگی های مختلف

حال نتایج شبیه سازی روش MSSRC را روی پایگاه داده ORL بررسی خواهیم کرد. از کل تصاویر مجموعه ORL، ۶۰٪ یعنی ۶ نمونه به عنوان آموزش و ۴۰٪ یعنی ۴ نمونه به عنوان آزمایش انتخاب شده است در جدول زیر از پیکسل‌های تصویر به عنوان ویژگی استفاده شده است.

جدول ۴-۴- نتایج شبیه سازی روش MSSRC روی پایگاه ORL با استفاده از ویژگی پیکسل‌های خام

| | | | |
|---|---|---|---|
| طول بردار ویژگی | ۱۶۸ | ۶۴۴ | ۲۶۹۴ |
| درصد بازشناسی | ٪ ۹۵/۶۳ | **٪ ۹۶/۲۵** | ٪ ۹۵/۹۱ |

جدول ۴-۵- نتایج شبیه سازی روش MSSRC روی پایگاه ORL با استفاده از ویژگی گرادیان هیستوگرام

| | | | | |
|---|---|---|---|---|
| طول بردار ویژگی | ۱۶۲ | ۴۵۰ | ۱۷۶۴ | ۲۹۱۶ |
| درصد بازشناسی | ٪ ۹۶/۸۸ | **٪ ۹۸/۱۳** | ٪ ۹۶/۰۷ | ٪ ۹۵/۰۱ |

جدول ۴-۶- نتایج شبیه سازی روش MSSRC روی پایگاه ORL با استفاده از ویژگی تبدیل موجک

| | | | |
|---|---|---|---|
| طول بردار ویژگی | ۶۴۴ | ۱۱۱۶ | ۲۶۹۴ |
| درصد بازشناسی | ٪ ۹۶/۸۸ | ٪ ۹۶/۲۹ | ٪ ۹۵/۹۱ |

در شکل ۴- ۴ دو نمونه از تصاویری که با هیچ کدام از ویژگی‌های گفته شده درست تشخیص داده نشده-اند را آورده‌ایم.

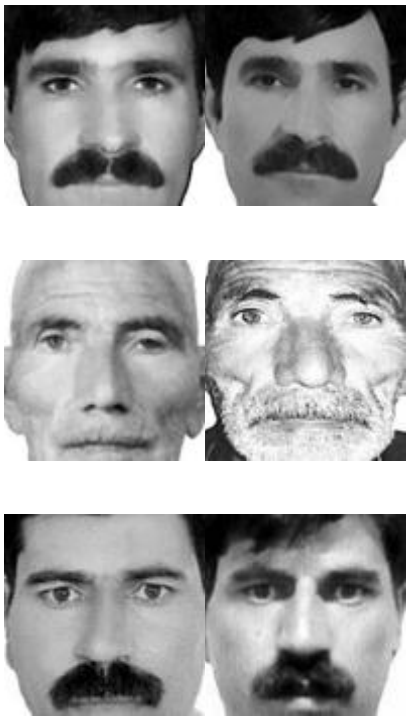
شکل ۴-۵- دو نمونه از تصاویری که اشتباه تشخیص داده شده‌اند.

شکل‌های سمت راست نمونه‌های آزمایش هستند.

در مرجع [۴۲] نتیجه روش NN روی پایگاه داده ORL، ٪ ۹۳/۰۵ آورده شده است. لازم به ذکر است در این مرجع طول بردار ویژگی قید نشده بود. همان‌طور که در جداول بالا مشهود است روش پیشنهادی نتیجه‌ی بهتری را ارائه داده است.

### ۴-۳-۲- روش MBP و روش تشخیص بدون استفاده از نمایش تنک:

از آنجا که نرم L1 بخاطر وجود تابع کوواریانس فقط روی پایگاه‌های داده بسیار بزرگ کار می‌کند، بنابراین نمی‌توان این روش را روی پایگاه های معرفی شده در بالا آزمایش کرد. ما این روش را برای کار بر روی پروژه‌ای صنعتی پیشنهاد دادیم. پایگاه داده این پروژه شامل ۷۵۰۰۰ تصویر از افراد مختلف است که نمونه‌ای از تصاویر این پایگاه داده را در شکل ۵-۵ آورده‌ایم.



شکل ۴-۵- در اشکال بالا ۶ تصویر از ۳ کلاس مختلف آورده شده است.

روش معرفی شده توسط ما نرخ تشخیص ۴۲/۱ ٪ را داشت و نرخ تشخیص SDK شرکت CogniTec نرخ تشخیص ۴۰/۸۳ ٪ را ارائه کرد.

در مورد روش تشخیص بدون استفاده از نمایش تنک این که نرم افزار تولید ما از نرم افزارهای مشابهImage Compraror و FaceSnap بهتر عمل می‌کند و در حال حاضر در شرکت ثبت احوال تهران به عنوان یکی از ماژول های تشخیص مورد استفاده قرار می گیرد.

**۴-۴- کارهای آینده :**

اگرچه در روش SRC ادعا شده است که ویژگی های استخراج شده تاثیری روی جواب نهایی نخواهد داشت اما ظاهرا این موضوع روی بعضی از پایگاه های داده صادق نیست. به همین منظور یکی از کارهایی که می توان در آینده انجام داد، انتخاب ویژگی های بهتری است، مثلا ویژگی های LBP ، HOG و موجک گابور و سپس اعمال PCA روی آنها احتمالا نتایج بهتری را به همراه داشته باشد. از طرف دیگر ترکیب روش نمایش تنک با روش های تشخیص چهره مبتنی بر مدل روشی است که باید در آینده مورد بررسی قرار گیرد.

# مراجع


[1] Turk M. and Pentland A., (1991), “Eigenfaces for recognition”, *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86.

[2] Belhumeur P., Hespanha J. and Kriegman D., (1997) “Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection,*”IEEE Trans. Pattern Analysis and Machine Intelligence*”, vol. 19, pp. 711-720.

[3] Liu C. and Wechsler H., (2000) “Evolutionary Pursuit and Its Application to Face Recognition,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, pp. 570-582.

[4] Wiskott L., Fellous J., and Malsburg C., (1997), “Face Recognition by Elastic Bunch Graph Matching,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, pp. 775-779,.

[5] Cootes T.F. , Edwards G.J. , and Taylor C.V. , (1998) “Active appearance models,” *in Proc. European Conference on Computer Vision, ,* vol. 2, pp. 484–498.

[6] Liu Y., Zhang D., Lu G., and Ma W.Y., (2007) “A survey of content-based image retrieval with high-level semantics,” *Pattern Recogn.*, vol.40, pp.262–282.

[7] Elad M., Figueiredo M. A. T. and Ma Y., (2010), “On the role of sparse and redundant representations in image processing,” *Proceedings of the IEEE*, vol. 98, no. 6, pp. 972–982.

[8] Elad M., (2010), *Sparse and Redundant Representations*, Springer.

[9] Mallat S. and Zhang Z., (1993), “Matching pursuits with time-frequency dictionaries,” *IEEE Trans. On Signal Proc.*, vol. 41, no. 12, pp. 3397–3415.

[10] Lewicki M.S and Sejnowski T., (2000), “Learning overcomplete representations,” *Neural Comp*., vol.12, no. 2, pp. 337–365.

[11] Mairal J., Bach F., Ponce J., Sapiro G., and Zisserman A., (2008) “Discriminative learned dictionaries for local image analysis,” *Computer Vision and Pattern Recognition IEEE Computer Society Conference*, vol.0,pp.1–8.

[12] Zhao W., Chellappa R., Phillips P.J., and Rosenfeld A., (2003) “Face Recognition: A Literature Survey,” *ACM Computing Surveys*, pp. 399-458.

[13] Kirby M. and Sirovich L., (1990) “Application of the Karhunen-Lo´eve procedure for the characterization of human faces,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 12, no. 1, pp. 103–108.

[14] Delac K., Grgic M. and Grgic S., (2005) , “Independent Comparative Study of PCA, ICA, and LDA on the FERET Data Set” , *International Journal of Imaging Systems and Technology*,Vol. 15, Issue 5, pages 252–260.

[15] Yang M., (2002), “Kernel Eigenfaces vs. Kernel Fisherfaces: Face Recognition Using kernel Methods”, *IEEE International Conference on Automatic Face and Gesture Recognition*, pp.215-220.

[16] Brunelli R. and Poggio T. (1992), "Face Recognition through Geometrical Features*", Computer Vision ECCV* , pp 792-800.

[17] Cootes T.F., Taylor C.J., and Lanitis A., (1994) “Active Shape Models: Evaluation of a Multiresolution Method for Improving Image Search”, *5th British Machine Vision Conference*, pages 327-336.

[18] Baldock E.R. and Graham J., (2000), ”Image Processing and Analysis,” Chapter7:Model-Based Methods in Analysis of Biomedical Images, Oxford University Press, pp223-248.

[19] Bourbaki, N. (1987), *Topological vector spaces*, Springer.

[20] Chen S. S., Donoho D. D. and Saunders M. A., (2001) “Atomic decomposition by basis pursuit,” *SIAM Rev.*, vol. 43, pp. 129–159.

[21] Tropp J.A. and Wright S.A., (2010), “Computational methods for sparse solution of linear inverse problems,” *Proceedings of the IEEE*, vol. 98, no. 6, pp. 948–958.

[22] Mallat S. and Zhang Z., (1993), “Matching pursuits with time-frequency dictionaries,” *IEEE Trans. On Signal Proc.*,vol. 41, no. 12, pp. 3397–3415.

[23] Davenport A. et.al, (2012), *Introduction to Compressed Sensing*, Chapter in Compressed Sensing: Theory and Applications, Cambridge University Press.

[24] Pati Y.C., Rezaiifar R., and Krishnaprasad P.S., (1993) “Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition,” *Asilomar Conf. Signal Syst. Comput*.

[25] Gorodnitsky I.F., Rao B.D., (1997), “Sparse signal reconstruction from limited data using FOCUSS: a re-weighted minimum norm algorithm”, *IEEE Trans Signal Process*, vol 45, pp 600–616.

[26] Gonzalez R.G. and Woods R.E., (2007) *Digital Image Processing*, Prentice Hall.

[27] Kovacevic J. and Chebira A., (2007) “Life beyond bases: The advent of frames (Part I),” *IEEE Signal Proc. Magazine*, vol. 24, no. 4, pp. 86–104.

[28] Rubinstein R., Bruckstein A.M. and Elad M., (2010) “Dictionaries for sparse representation modeling,” *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1045–1057.

[29] Gersho A. and Gray R.M., (1992), *Vector Quantization and Signal Compression*, Springer.

[30] Engan K., Aase S.O. and Hakon J., (1999) “Method of optimal directions for frame design,” *in Proceedings of IEEE ICASSP*, vol. 5.

[31] Aharon M., Elad M. and Bruckstein A., (2006) “K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Trans. on Signal Processing*, vol. 54, no. 11, pp.4311–4322.

[32] Wright J., Yang A.Y.,Ganesh A., Sastry S.S., Ma Y., (2009), “Robust face recognition via sparse representation,” *TPAMI,* vol.31, pp. 210–227.

[۳۳] تالبوت م، (۱۳۸۹)، **"جهان هولوگرافیک"**، مهرجویی د، انتشارات هرمس، چاپ شانزدهم، صص ۱۰۵-۱۰۸

[34] Dalal N. and Triggs B., (2005), “Histograms of Oriented Gradients for Human Detection,” *Conference on Computer Vision and Pattern Recognition.*

[۳۵] خسروی ح، کبیر ا، (۱۳۸۵)، "معرفی دو ویژگی سریع و کارآمد برای بازشناسی ارقام دستنویس"، **چهارمین کنفرانس ماشین بینایی و پردازش تصویر ایران**، مشهد

[36] Dhall S., Sethi P., (2000), "Geometric and Appearance Feature Analysis For Facial Expression Recognition," *International Journal of Advanced Engineering Technology.*

[37] Lowe D.G., (2004), "Distinctive image features from scale-invariant keypoints", *International journal of computer vision*, Vol. 60(2), pp. 91-110.

[۳۸] کهنسال ا. ، (۱۳۹۳)، پایان نامه ارشد، "تشخیص چهره در تصاویر ویدئویی درگاه های ورودی"، دانشکده کامپیوتر و هوش مصنوعی، دانشگاه شاهرود.

[39] Martinez A., Benavente R., (1998), "The AR Face Database," *Computer Vision Center (CVC).*

[40] A.T.L. Cambridge, The Database of Faces. http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html.

[41] Ortiz E.G., Becker B.C., (2013), "Face recognition for web-scale datasets," *Comput. Vis. Image Understanding*.

[42] Thakur S., Sing J.K., Basu D.K., Nasipuri M., Kundu M., (2008) , "Face Recognition using Principal Component Analysis and RBF Neural Networks," *Emerging Trends in Engineering and Technology*, pp. 695-700.

**Abstract**

With ever increasing digital images on personal computers and web servers, the image classification is considered more than ever. Accordingly in recent years various features and also classifiers is presented to solve this problem. Combining various features with the classifier based on neural network is one of the suggestions that is presented in this thesis. Classification based on sparse representation is another method that we will discuss here, and according to the novelty of this approach, we will further focus on it. In this method, within a large number of basis signals that are much greater than their dimensions, in general; we select the minimum number of basis to display a signal. Each basis signal is called an "atom" and set of these atoms is called a "dictionary". It's very hard that select the best and lowest atom to represent a signal. But in recent years, researchers optimized both the speed and accuracy of this method. Thus, this method quickly found several applications in signal processing such as compression, image denoising, pattern recognition and authentication. Before solving a problem using sparse representation, we must consider two important issues. The first issue is finding an appropriate dictionary for the data and the second issue is to find an efficient algorithm to obtain a sparse representation of the signal. In this thesis, we first review a number of well-known algorithms in face recognition and then represent the issue of sparse classification, dictionary learning and sparse coding. Next, we propose a method based on neural network and also will offer a method inspired by sparse representation. The simulation results show the promising performance of this methods. For example, One of the methods that are proposed in this thesis present recognition rate 98.13 % on the ORL database.

Keywords:

Face Recognition, Sparse Representation, Dictionary Learning, Sparse Coding


**Electronic Engineering Department**

**Faculty Of Electronic**

# Authentication by Face in a large dataset of face image

**Hossein saeidi**

**Supervisor**

**Dr. Hossein Khosravi**

**Advisor**

**Dr. Vahid Abolghasemi**

**September 2015**