



دانشگاه امیرکبیر

دانشکده مهندسی برق و کامپیوتر

شناسایی چهره دو مرحله ای وزن دار با استفاده از نمایش تنگ نمونه تست

توسط:

محمد مالمیر

میلاذ خادمی نوری

استاد درس:

دکتر ساناز سیدین

قباستان ۹۶

الحمد لله الذي
خلقنا من
الحمم

فهرست مطالب

فصل ۱-مقدمه	۱.....
۱-۱- چکیده	۱.....
۱-۲- پیشگفتار	۱.....
۱-۳- روش های ارائه شده در تشخیص چهره	۲.....
۱-۴- روش ها تشخیص چهره مبتنی بر نمایش تنک	۳.....
فصل ۲-روش دو مرحله ای نمایش تنک نمونه تست (TPTSSR)	۴.....
۲-۱- مرحله ۱	۴.....
۲-۲- مرحله ۲	۵.....
فصل ۳-روش دو مرحله ای نمایش تنک و وزن دار نمونه تست (WTPTSSR)	۶.....
۳-۱- حل مساله بهینه سازی وزن دار در روش WTPTSSR	۶.....
۳-۲- مراحل اصلی روش WTPTSSR	۷.....
فصل ۴-نتایج	۸.....
۴-۱- انتخاب پارامترها	۸.....
۴-۲- شناسایی چهره	۹.....
۴-۳- تست بر روی دیتابیس چهره ORL	۹.....
۴-۴- تست بر روی دیتابیس چهره AR	۱۲.....
۴-۵- تست با استفاده از دیتابیس چهره Yale	۱۴.....
۴-۶- نتایج تست بر روی دیتابیس FERET	۱۶.....

فصل ۵-نتیجه گیری.....۱۸

فصل ۶-مراجع.....۱۹

فصل ۷-پیاده سازی نرم افزاری.....۲۱

فهرست شکل‌ها

- شکل ۴-۱: عملکرد الگوریتم WTPTSSR بر حسب تغییرات t ۹
- شکل ۴-۲: نمونه تصویر یک فرد از دیتابیس ORL ۱۰
- شکل ۴-۳: متوسط نرخ شناسایی بر حسب پارامتر M در دیتابیس ORL ۱۱
- شکل ۴-۴: نمونه تصویر یک فرد از دیتابیس AR ۱۲
- شکل ۴-۵: متوسط نرخ شناسایی بر حسب پارامتر M در دیتابیس AR ۱۳
- شکل ۴-۶: نمونه تصویر چهره از دیتابیس Yale. شکل ۴-۷ ۱۴

فهرست جدول‌ها

- جدول ۴-۱: بیشینه مقدار نرخ متوسط شناسایی و انحراف معیار الگوریتم‌های مختلف در دیتابیس ORL در طول ۱۰ بار اجرای الگوریتم در حالتی که ۵ تصویر از هر شخص به صورت تصادفی برای آموزش در نظر گرفته شده است. ۱۱
- جدول ۴-۲: بیشینه مقدار نرخ شناسایی الگوریتم‌های مختلف برای دیتابیس AR ۱۳
- جدول ۴-۳: بیشینه نرخ شناسایی الگوریتم‌های مختلف در دیتابیس Yale ۱۵
- جدول ۴-۴: بیشینه نرخ شناسایی الگوریتم‌های مختلف در دیتابیس FERET ۱۶

فصل ۱- مقدمه

۱-۱- چکیده

در این گزارش به دنبال توصیف روش طبقه بندی شناسایی چهره دو مرحله ای وزن دار با استفاده از نمایش تنک نمونه تست ($WTPTSSR^1$) و مقایسه این روش با روش طبقه بندی شناسایی چهره دو مرحله ای با استفاده از نمایش تنک نمونه تست ($TPTSSR^2$) و حل مساله بهینه سازی مربوطه هستیم. به طور کلی نمایش تنک (SR^3) با استفاده از یک دیکشنری کامل برای سیگنال برای بسیاری از محققین در دهه اخیر بسیار جذاب بوده است. در همین راستا روش $TPTSSR$ در شناسایی چهره از کارایی فوق العاده ای برخوردار است با این حال این روش تمایل به طبقه بندی از طریق اطلاعات کلی دارد و اطلاعات محلی را نادیده می گیرد. به منظور رفع این مشکل روش $WTPTSSR$ توسط نویسندگان مقاله پیشنهاد شده است که از هر دو نوع اطلاعات محلی و سراسری بهره می گیرد. این روش را می توان توسعه ای بر روش $TPTSSR$ در نظر گرفت. در ادامه نشان داده خواهد شد که روش $WTPTSSR$ بسیار کارآمدتر از روش $TPTSSR$ در شناسایی چهره عمل خواهد کرد.

۱-۲- پیشگفتار

علم بیومتریک^۴ اشاره دارد به تکنولوژی برای اندازه گیری و آنالیز مشخصات بدن افراد جهت تشخیص هویت شخص. همه سیستم های بیومتریک دارای معماری ویژه ای برای پردازش نمونه مورد بررسی و احراز هویت می باشند. روش های مختلفی برای تشخیص هویت در بیومتریک وجود دارد که هر یک با توجه به دقت و کارایی مورد استفاده قرار می گیرند. اثر انگشت به دلیل اینکه برای هر فرد منحصر به فرد است و با گذشت زمان هیچ گونه تغییری نمی کند، در میان سیستم های بیومتریک بیشتر مورد استفاده قرار می گیرد. البته سیستم های دیگر مانند: عنبیه چشم، شبکیه چشم و نمودار حرارتی چهره هم از فردی به فرد دیگر متفاوت هستند. برای افزایش کارایی و امنیت و دقت سیستم می توانیم از ترکیبات بیومتریک استفاده کنیم. شناسایی چهره یکی از شاخه های بسیار

^۱ Weighted Two phase Test Sample Sparse Representation

^۲ Two phase Test Sample Sparse Representation

^۳ Sparse Representation

^۴ Biometric

فعال بیومتریک محسوب می شود. یک امتیاز تکنولوژی شناسایی چهره غیر تماسی بودن آن است. این نوع شناسایی همچنین خللی به زندگی روزمره افراد هم وارد نمی کند. روش های زیادی به منظور شناسایی چهره پیشنهاد شده اند و تا کنون این روش ها در بخش های مختلفی مانند فرودگاه ها، کنترل های مرزی و کنترل دسترسی به کار گرفته شده اند.

۱-۳- روش های ارائه شده در تشخیص چهره

یک سیستم شناسایی شناسایی چهره می بایست به منظور تکمیل فرآیند شناسایی مراحل زیر را انجام دهد: گرفتن تصویر چهره، استخراج ویژگی های تصویر چهره و اعمال طبقه بندی چهره. تاکنون مقالات مختلفی روش های متنوعی برای تشخیص چهره پیشنهاد کرده اند. برای مثال روش های کلاسیک مبتنی بر کاهش بعد شامل تحلیل اجزای اصلی (PCA^1) [۵] تجزیه و تحلیل محرک تانسور عمومی^۲ [۶]، منهتن NMF^3 [۷]، تحلیل اجزای اصلی قوی ($RPCA^4$) [۸]، روش $NeNMF^5$ [۹]. تکنیک های شناسایی چهره مبتنی بر نمایش تنک [۱۰-۱۹] در سال های اخیر بسیار جلب توجه کرده اند. این روش ها را به دو دسته کلی می توان تقسیم کرد: روش های نمایش مبتنی بر نرم L_1 ($1NBRM^6$) و روش های مبتنی بر نرم L_2 ($2NBRM$). روش های ارائه شده در [۱۰-۱۴] نمونه هایی از به کارگیری روش $1NBRM$ هستند. طبقه بندی کننده SRC^7 در [۱۱] مساله شناسایی را به عنوان یکی از مدل های بازسازی خطی چندگانه در نظر می گیرد و توجیه می کند که چطور این تئوری جدید در نمایش تنک سیگنال می تواند پاسخ مناسبی به این مساله بدهد. SRC در برابر مانع^۸، تابش و نویز مقاوم است و یک نتیجه عالی به دست می دهد. Gao در [۱۲] یک نسخه مبتنی بر کرنل از SRC ارائه می دهد. Yang در [۱۳] یک روش کدینگ تنک مقاوم از طریق مدل سازی کدینگ تنک به عنوان یک مساله رگرسیون مقاوم محدود به تنک بودن ارائه می کند به طوری که روش او به نویز مقاوم است. Lai در [۱۴] یک روش روش پروجکشن^۹

^۱ Principle Component Analysis

^۲ General tensor discriminant analysis

^۳ Manhattan Non-negative Matrix Factorization

^۴ Robust Principle Component Analysis

^۵ Non-Negative Matrix Factorization

^۶ Norm Based Representation Method

^۷ Sparse Representation Classifier

^۸ Occlusion

^۹ Projection

مجزای محلی تنک که به پروجکشن مجزای محلی محدودیت تنک بودن را تحمیل می کند. در این روش می بایست نمونه تست به صورت تنک با استفاده از نمونه آموزشی توصیف شود. روش های ارائه شده در [۱۵-۱۹] نمونه مثال هایی از روش های مبتنی بر ۲NBRM هستند. Xu در [۱۷] یک روش نمایش تنک دو مرحله ای TPTSR که در واقع یک طبقه بندی کلی تا دقیق بر روی نمونه های چهره انجام می دهد و نتیجه مطلوبی به دست می آورد که در مقایسه با ۱NBRM، ۲NBRM امتیازات قابل توجهی دارد. اولاً ۲NBRM نیازی به محاسبات تکراری ندارد و ثانیاً پاسخ از طریق حل یک دستگاه خطی شامل معادلات خطی به دست می آید که این معمولاً از لحاظ محاسباتی به صرفه است. سوماً استفاده کردن از نرم L_1 تنها روش برای رسیدن به تنک بودن نیست در واقع یک استفاده درست از نرم L_2 نیز می تواند منجر به نمایش تنک شود (SR) [۱۷]. علاوه بر این استفاده از ۲NBRM ممکن است به دقت بالاتری از ۱NBRM منجر شود برای مثال Zhang در [۱۵] و Shi در [۱۶] نشان دادند که ۲NBRM بهتر از روش مشهور ۱NBRM ارائه شده در [۱۰، ۱۱] در دقت طبقه بندی عمل می کنند.

۱-۴- روش ها تشخیص چهره مبتنی بر نمایش تنک

در شناسایی الگو نشان داده می شود که ساختارهای محلی داده ها بسیار اهمیت هستند [۲۰-۲۲]. نزدیک ترین همسایه (NN^1) از ساختارهای محلی داده استفاده می کند با این حال NN تصاویر تست را تنها با استفاده از NN مربوط به داده های آموزشی طبقه بندی می کند که در نتیجه می تواند متاثر از نویز شود. روش SR از ساختار های خطی در داده استفاده می کند. Yu در [۲۳، ۲۴] اشاره می کند که تحت شرایط خاص محلی بودن ضروری تر از تنک بودن است. ولی نمایش تنک نمونه تست دو مرحله ای (TPTSSR) محلی بودن را تضمین نمی کند. یعنی TPTSSR یک نمونه تست را تنها با استفاده از نمونه های آموزشی بازسازی می کند و این نمونه های آموزشی ممکن است شامل نمونه هایی باشند که از نمونه تست بسیار دور هستند و در نتیجه نتایج غیرقابل اعتمادی را طبقه بندی ایجاد کنند. بنابراین TPTSSR تمایل دارد که اطلاعات محلی را از دست بدهد به منظور غلبه بر این مشکل یک نسخه توسعه یافته از TPTSSR با نام نمایش دو مرحله ای تنک وزن دار (WTPTSSR) توسط نویسندگان مقاله ارائه می شود. نکته کلیدی در خصوص این روش آن است که در آن هم ساختار های محلی و هم تنک بودن مورد توجه قرار خواهد گرفت.

¹ Nearest Neighbor

فصل ۲- روش دو مرحله ای نمایش تنگ نمونه تست (TPTSSR)

روش دو مرحله ای نمایش نمونه تست (TPTSSR) [۱۹] به طور کلی در دو مرحله توصیف می شود. در ابتدا روش TPTSSR نمونه تست را به عنوان یک ترکیب خطی از همه نمونه های آموزشی در نظر می گیرد و تعداد M نزدیک ترین همسایه به نمونه تست را از روی نمونه های آموزشی به نحوی که بهترین نمایش را از نمونه تست ارائه دهند انتخاب می کند. سپس TPTSSR نمونه تست را بر اساس یک ترکیب خطی همه M نزدیک ترین همسایه نمایش می دهد و از اطلاعات نمایش به منظور طبقه بندی استفاده می کند. فرض کنید که ما N نمونه آموزشی متعلق به C کلاس داریم به نحوی که هر کلاس شامل n نمونه می باشد، $A_k = [a_{k,1}, \dots, a_{k,n}] \in R^{m \times n}$ به طوری که $1 \leq k \leq C$ و m بعد نمونه ها می باشد.

۲-۱- مرحله ۱

y را نمونه تست در نظر بگیرید. از $y = \frac{y}{\|y\|}$ و $a_{k,i} = \frac{a_{k,i}}{\|a_{k,i}\|}$ ($1 \leq i \leq n$) به منظور تبدیل نمونه تست و نمونه آموزشی به بردار واحد استفاده کنید. در ابتدا فرض می کنیم که نمونه تست y به صورت تقریبی از ترکیب خطی همه نمونه های آموزشی تقریب زده شود.

$$y = AX = a_{1,1}x_{1,1} + a_{1,2}x_{1,2} + \dots + a_{c,n}x_{c,n} \quad (1)$$

به طوری که $A = [A_1, A_2, \dots, A_c] = [a_{1,1}, a_{1,2}, \dots, a_{1,c}]$ ماتریس متشکل از همه N نمونه آزمایشی مربوط به C کلاس می باشد و $x_{i,j}$ ($i=1, \dots, C, j=1, \dots, n$) ضرایب محسوب می شوند. هدف کمینه کردن $\|y - AX\|_2$ است. از آنجا که این مساله یک مساله کمترین مربعات است دارای جواب تحلیلی به صورت زیر است:

$$\hat{X} = (A^T A + \sigma I)^{-1} A^T y \quad (2)$$

در این رابطه عبارت σI باعث اضافه شدن یک مقدار مثبت کوچک به قطر اصلی $A^T A$ می شود تا مطمئن باشیم که قطر اصلی غیرصفر بوده و معکوس وجود دارد. در رابطه (۱) دیده می شود که هر یک از نمونه های آموزشی در شکل گیری نمونه تست سهم خودش را دارد. سهمی که نمونه آموزشی $a_{i,j}$ ایجاد می کند برابر $a_{i,j}x_{i,j}$ است ما به منظور تخمین و ارزیابی میزان سهم و تاثیر این نمونه آموزشی از رابطه (۳) استفاده می کنیم.

$$e_{i,j} = \|y - a_{i,j}x_{i,j}\|_2 \quad (3)$$

یک مقدار کوچک برای $e_{i,j}$ به معنای داشتن سهم بیشتر در بازسازی نمونه تست است. بر این اساس تعداد M نمونه آموزشی که بیشترین سهم را دارند انتخاب می شوند و به صورت $(1 \leq M \leq N)$ $a_1 \dots a_M$ در نظر گرفته می شوند. ما این M نمونه آموزشی را M نزدیک ترین همسایه نمونه تست در نظر می گیریم.

۲-۲-۲ مرحله ۲

فرض کنید که نمونه تست y را بتوان به صورت ترکیب خطی M نمونه آموزشی با بیشترین سهم در نمایش نمونه تست تقریب زد:

$$y = \tilde{A}\tilde{X} = a_1x_1 + a_2x_2 + \dots a_Mx_M \quad (۴)$$

به طوری که $\tilde{A} = [a_1 \dots a_M]$ و $\tilde{X} = [x_1 \dots x_M]$ چنانچه مجدداً روش مساله نرم ۲ را برای این حالت در نظر بگیریم به جواب بسته رابطه (۵) خواهیم رسید.

$$\tilde{X} = (\tilde{A}^T \tilde{A} + \gamma I)^{-1} \tilde{A}^T y \quad (۵)$$

به طور مشابه در این رابطه γ یک مقدار مثبت کوچک و I یک ماتریس واحد است.

به علت آنکه همسایه ها ممکن است از کلاس های متفاوتی باشند ما مجموع سهم نمونه های آموزشی در هر کلاس را در نمایش نمونه تست محاسبه می کنیم و حاصل این جمع را به منظور طبقه بندی نمونه تست استفاده می کنیم. فرض کنید t_i معرف کلاس i ام است که اعضای آن $a_1^{t_i}, \dots a_{t_i}^{t_i}$ می باشد و ضرایب متناظر آن ها هم $x_1^{t_i}, \dots x_{t_i}^{t_i}$ باشند. به منظور ارزیابی میزان شراکت هر کلاس در بازسازی نمونه تست ما رابطه (۶) را از نمونه های آموزشی کلاس i ام در نمایش نمونه تست تشکیل می دهیم.

$$con_i = \left\| y - \sum_{j=1}^{t_i} a_j^{t_i} x_j^{t_i} \right\|_2 \quad (۶)$$

یک مقدار کوچکتر برای عبارت con_i به معنای سهم بیشتر در نمایش نمونه تست است. اگر برای نمونه تست رابطه $h = \arg \min_i con_i$ را حل کنیم آنگاه نمونه تست i ام را به کلاس h ام نسبت خواهیم داد.

فصل ۳- روش دو مرحله ای نمایش تنگ و وزن دار نمونه تست WTPTSSR

در روش $TPTSSR$ یک نمونه تست با استفاده از همه نمونه های آموزشی بازسازی می شود. در این حالت نمونه های آموزشی ممکن بود شامل نمونه هایی بسیار دور از نمونه تست باشند و در نتیجه یک طبقه بندی غیرمطمئن انجام می شود. علاوه بر این $TPTSSR$ از ساختار خطی داده استفاده می کرد و در نتیجه تمایل به از دست دادن اطلاعات محلی داشت. در بسیاری از موارد همین اطلاعات محلی می توانند بسیار ضروری باشند و نرخ شناسایی بالایی به دست دهند [۲۲]. به منظور غلبه بر این مشکل یک روش جدید با عنوان $WTPTSSR$ توسط نویسندگان مقاله ارائه شد که در آن هم ساختارهای محلی و هم تنگ بودن در نظر گرفته می شوند.

۳-۱- حل مساله بهینه سازی وزن دار در روش WTPTSSR

مشابه $TPTSSR$ نمونه تست با استفاده از نمونه های آموزشی نمایش داده می شود. همچنین به طور همزمان ما محلی بودن را بر اساس نرم ۲ اعمال می کنیم. $WTPTSSR$ در واقع مساله کمینه سازی وزن دار نرم ۲ زیر را حل می کند:

$$\hat{X} = \operatorname{argmin} \|WX\|_2 \text{ subject to } y = AX \quad (7)$$

در این رابطه W یک ماتریس قطری بوده و یک تطبیق دهنده محلی است که فاصله بین y و هر کدام از داده های آموزشی را جریمه می کند. این ماتریس می تواند به صورت رابطه (۸) نمایش داده شود.

$$W = \operatorname{diag}([dist(y, a_{1,1}), dist(y, a_{1,2}), \dots, dist(y, a_{c,1}), \dots, dist(y, a_{c,n})]) \quad (8)$$

در این رابطه $dist(y, a_{i,j}) = \|y - a_{i,j}\|^k$ و k پارامتر تطبیق محلی است. وقتی $k=0$ روش $WTPTSSR$ تبدیل به روش $TPTSSR$ می شود. یک مقدار بیشتر برای $dist(y, a_{i,j})$ یک فاصله بیشتر بین y و $a_{i,j}$ را نشان می دهد. این فاکتور به خوبی می تواند میزان شباهت بین نمونه تست و نمونه های آموزشی را مشخص کند. بنابراین ضرایب کدگذاری $WTPTSSR$ تمایل دارند در نمایش خطی بیشتر محلی باشند (از کلاس مشترک با نمونه تست باشند). به منظور حل با استفاده از روش لاگرانژ رابطه (۷) را به صورت زیر بازنویسی می کنیم:

$$\begin{aligned} \min(\|AX - y\|_2^2 + t * \|WX\|_2^2) &= (X^T A^T - y^T)(AX - y) + t(X^T W^T)WX \quad (9) \\ &= X^T A^T AX - X^T A^T y - y^T AX + y^T y + tX^T W^T WX \\ &= X^T (A^T A + tW^T W)X - 2X^T A^T y + y^T y \end{aligned}$$

در این رابطه پارامتر t ($t > 0$) یک پارامتر رگولاریزاسیون است. این رابطه یک مساله رگولاریزاسیون تیکونوف^۱ است که دارای جواب تحلیلی می باشد که در رابطه (۱۰) نشان داده شده است:

$$X = (A^T * A + t * W^T * W)^{-1} * A^T * y \quad (۱۰)$$

۳-۲- مراحل اصلی روش WTPTSSR

مراحل اصلی روش WTPTSSR را می توان به صورت زیر نام برد:

ورودی: ماتریس نمونه های آموزشی $A \in R^{m \times N}$ و نمونه تست $y \in R^m$.

به منظور داشتن نرم ۲ واحد، ستون های A و y را نرمالایز کنید.

با استفاده از رابطه (۱۰) M نزدیک ترین همسایگی اول نمونه تست را مشخص کنید.

از M نزدیک ترین همسایگی نمونه تست برای تشکیل رابطه (۴) و حل آن استفاده کنید.

از رابطه (۶) برای محاسبه مقدار انحراف از کلاس ام استفاده کنید و نمونه تست را در کلاسی که کمترین مقدار con_i دارد طبقه بندی کنید.

WTPTSSR هر دوی اطلاعات خطی و محلی را برای ارتقای کیفیت فرایند شناسایی ترکیب می کند و در نتیجه می تواند کارایی بهتری ارائه دهد. در حوزه خطی روش WTPTSSR در واقع به طور مستقیم از توسعه روش TPTSSR به دست می آید.

^۱ Tikhonov

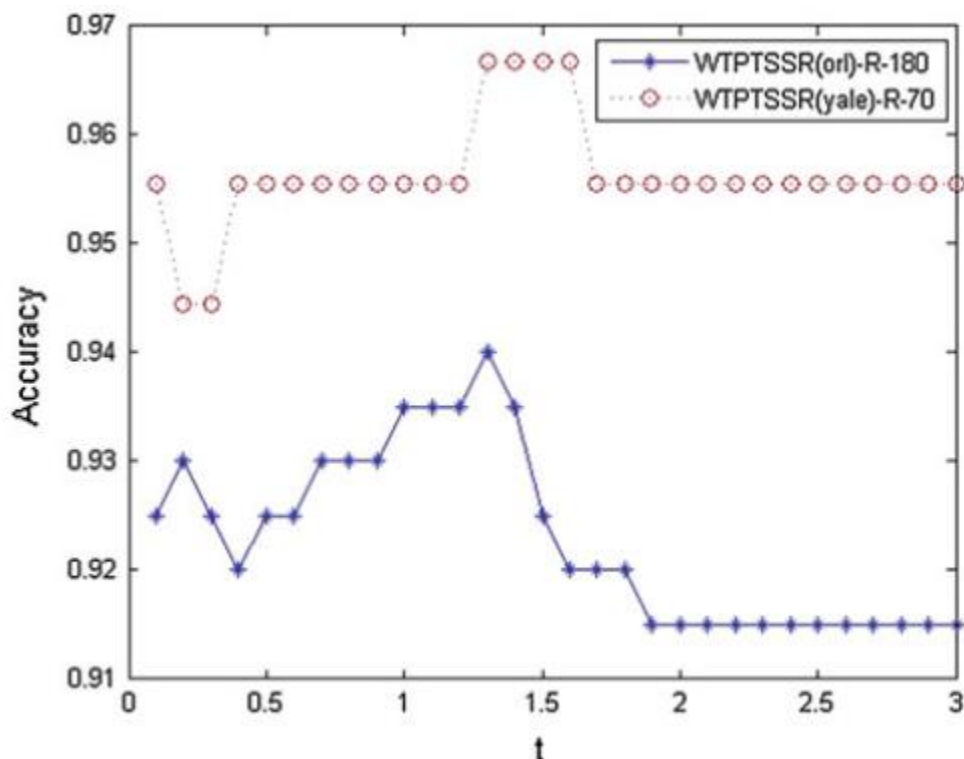
فصل ۴- نتایج

در این بخش به دنبال مقایسه عملکرد الگوریتم WTPTSSR با الگوریتم های دیگر و بخصوص TPTSSR و ارزیابی کارایی آن هستیم. در ابتدا می بایست پارامترهای فرایند را تعیین کنیم. پارامترهای δ و γ در الگوریتم TPTSSR برابر ۰,۰۱ در نظر گرفته می شوند و در مورد پارامتر M (تعداد MN نمونه آموزشی برای نمونه تست) در الگوریتم های WTPTSSR و TPTSSR مقدار بهینه به صورت تجربی استخراج می شود.

۴-۱- انتخاب پارامترها

هنوز یک پارامتر رگولایزاسیون t وجود دارد که می بایست برای الگوریتم WTPTSSR تعیین شود. به منظور تخمین این پارامتر ما از دیتابیس چهره ^۱ ORL و دیتابیس Yale برای تست استفاده کردیم. ما به طور تصادفی پنج تصویر از افراد مختلف را برای تشکیل مجموعه آموزشی انتخاب کردیم و باقیمانده تصاویر هر فرد را به عنوان مجموعه تست در نظر گرفتیم. الگوریتم ده بار بر روی هر کدام از دیتابیس ها اعمال شد. در نتیجه آزمایش انجام شده حداقل ابعاد برای دیتابیس ORL، برابر ۱۸۰ و برای دیتابیس Yale برابر ۸۰ به در نظر گرفته شد. مقادیر مختلف پارامتر t نیز از ۰,۱ تا ۳ بررسی شد که شکل ۱ میانگین دقت شناسایی الگوریتم WTPTSSR برای مقادیر مختلف این پارامتر را نشان می دهد.

^۱ Olivetti Research Laboratory



شکل ۴-۱: عملکرد الگوریتم WTPTSSR بر حسب تغییرات t.

همان طور که در شکل ۴-۱ دیده می شود نرخ شناسایی الگوریتم WTPTSSR با تغییر t تغییر می کند. نرخ شناسایی متوسط در الگوریتم WTPTSSR در هر دو دیتابیس به ازای t برابر ۱,۳ به حداکثر مقدار خود می رسد. بنابراین به طور کلی ما t را برابر ۱,۳ برای استفاده در مراحل بعدی در نظر می گیریم.

۴-۲- شناسایی چهره

در این بخش ما از چهار دیتابیس معروف و عمومی استفاده می کنیم و همچنین کارایی الگوریتم ارائه شده را با الگوریتم های PCA[۵]، SRC[۱۱]، WSRC[۲۲] و TPTSSR مقایسه می کنیم.

۴-۳- تست بر روی دیتابیس چهره ORL

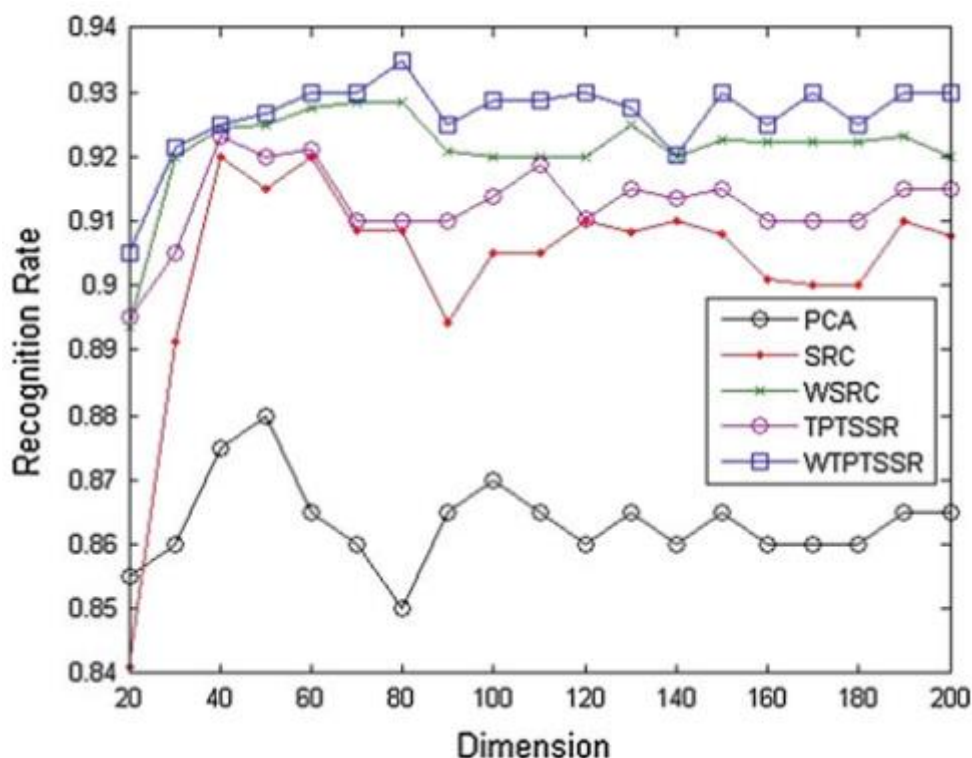
دیتابیس چهره ORL[۲۵] شامل ۴۰۰ تصویر از شخص مختلف (۱۰ تصویر از هر شخص) می باشد. تصاویر در زمان های مختلف گرفته شده اند و شامل حالت ها مختلفی از چهره فرد مانند چشمان باز و بسته، در حال خنده

و یا اخم و جزئیات دیگر چهره مانند با عینک و بدون عینک می باشد. همچنین این تصاویر در یک محدوده چرخش و پیچش زاویه صورت تا ۲۰ درجه گرفته شده اند. اندازه هر تصویر برابر ۱۱۲ در ۹۲ پیکسل می باشد. شکل ۴-۲ یک تصویر نمونه از یک شخص را نشان می دهد.



شکل ۴-۲: نمونه تصویر یک فرد از دیتابیس ORL.

به منظور تست، پنج تصویر از هر شخص به صورت تصادفی برای آموزش و پنج تصویر باقیمانده به منظور تست انتخاب می شوند. تست بر روی تصاویر ۱۰ بار انجام می شود. پارامتر M در الگوریتم های TPTSSR و WTPTSSR برابر ۳۵ در نظر گرفته می شود. شکل ۴-۳ نرخ شناسایی متوسط را نسبت به ابعاد نشان می دهد.



شکل ۴-۳: متوسط نرخ شناسایی بر حسب پارامتر M در دیتابیس ORL.

جدول ۴-۱ مقادیر نرخ شناسایی متوسط بیشینه و انحراف معیار هر روش را در طول ۱۰ با اجرا نشان می دهد. در شکل ۳ دو نکته مهم دیده می شود، اول آنکه الگوریتم WTPTSSR مستقل از تغییر ابعاد عملکرد بهتری در مقایسه با چهار الگوریتم دیگر به دست آورده است و دوم آنکه WTPTSSR همیشه بهتر از TPTSSR عمل کرده است. از جدول ۱ هم دیده می شود که الگوریتم WTPTSSR به ماکزیمم نرخ متوسط شناسایی ۹۳٫۵ درصد رسیده است.

جدول ۴-۱: بیشینه مقدار نرخ متوسط شناسایی و انحراف معیار الگوریتم های مختلف در دیتابیس ORL در طول ۱۰ بار اجرای الگوریتم در حالتی که ۵ تصویر از هر شخص به صورت تصادفی برای آموزش در نظر گرفته شده است.

Method	PCA	SRC	WSRC	TPTSSR	WTPTSSR
Recognition rate	88 % \pm 3.0	92 % \pm 1.0	93 % \pm 2.0	92.5 % \pm 1.5	93.5 % \pm 2.0
Dimension	50	40	70	40	80

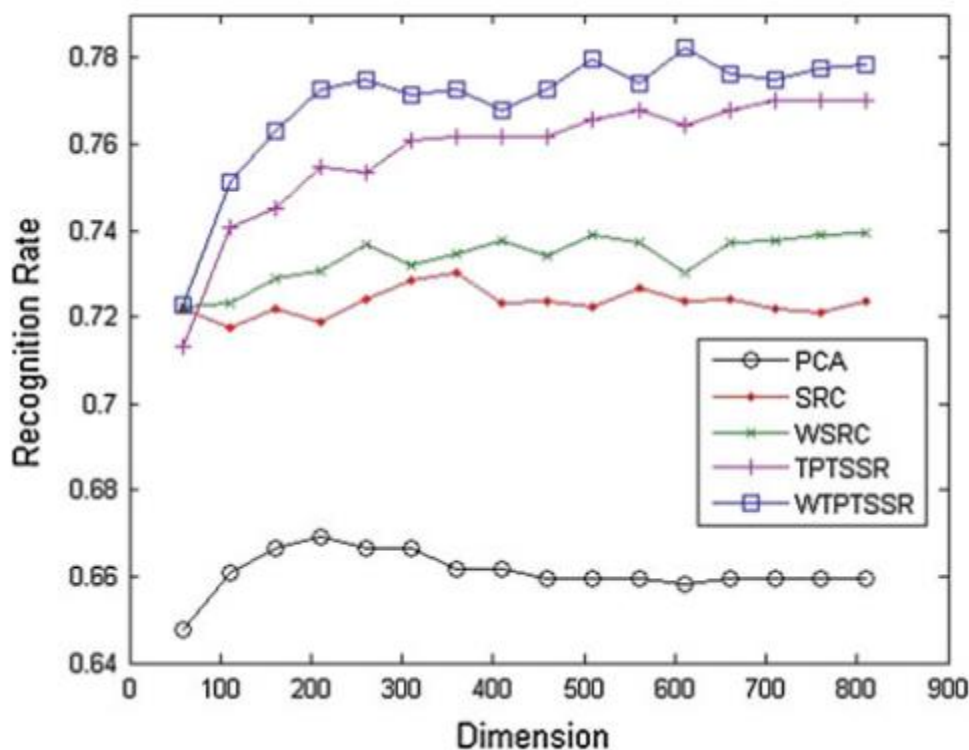
۴-۴- تست بر روی دیتابیس چهره AR

تست بر روی دیتابیس چهره [۲۶]AR، این دیتابیس شامل ۴۰۰۰ تصویر چهره رنگی از ۱۲۶ نفر می باشد که شامل ۲۶ تصویر از روبروی چهره با حالت های چهره مختلف و شرایط نوری مختلف می باشد. تصاویر ۱۲۰ نفر در دو بازه زمانی (با ۱۴ روز فاصله) و در هر بازه شامل ۱۳ تصویر رنگی می باشد. ۱۴ تصویر چهره (شامل ۷ تصویر از هر بازه زمانی) از این ۱۲۰ نفر برای تست در نظر گرفته شدند و سپس تصاویر رنگی به تصاویر خاکستری تبدیل شدند. شکل ۴-۴ یک نمونه تصویر از یک شخص را از این دیتابیس را نشان می دهد.



شکل ۴-۴: نمونه تصویر یک فرد از دیتابیس AR.

در این آزمایش ۱۲۰ در ۷ تصویر چهره از بازه زمانی اول برای آموزش و ۱۲۰ در ۷ تصویر چهره از بازه زمانی دوم به منظور تست انتخاب می شود. پارامتر M در TPTSSR و WPTSSR برابر ۱۵۰ در نظر گرفته می شود. شکل ۴-۵ نرخ شناسایی را نسبت به تغییر ابعاد نشان می دهد.



شکل ۴-۵: متوسط نرخ شناسایی بر حسب پارامتر M در دیتابیس AR.

جدول ۴-۲: بیشینه مقدار نرخ شناسایی الگوریتم های مختلف برای دیتابیس AR.

Method	PCA	SRC	WSRC	TPTSSR	WTPTSSR
Recognition rate	66.9 %	73 %	73.92 % \pm 2.0	77.02 %	78.21 %
Dimension	210	360	260	710	610

جدول ۴-۲ نیز نرخ شناسایی ماکزیمم و همچنین بعد متناظر با هر یک از روش های طبقه بندی را نشان می دهد. از این جدول همچنین دیده می شود که ماکزیمم نرخ شناسایی الگوریتم های WTPTSSR و TPTSSR نزدیک هم هستند. از شکل ۵ هم می توان دید که روش WTPTSSR نسبت به چهار الگوریتم دیگر مستقل از تغییر ابعاد عملکرد بهتری داشته است.

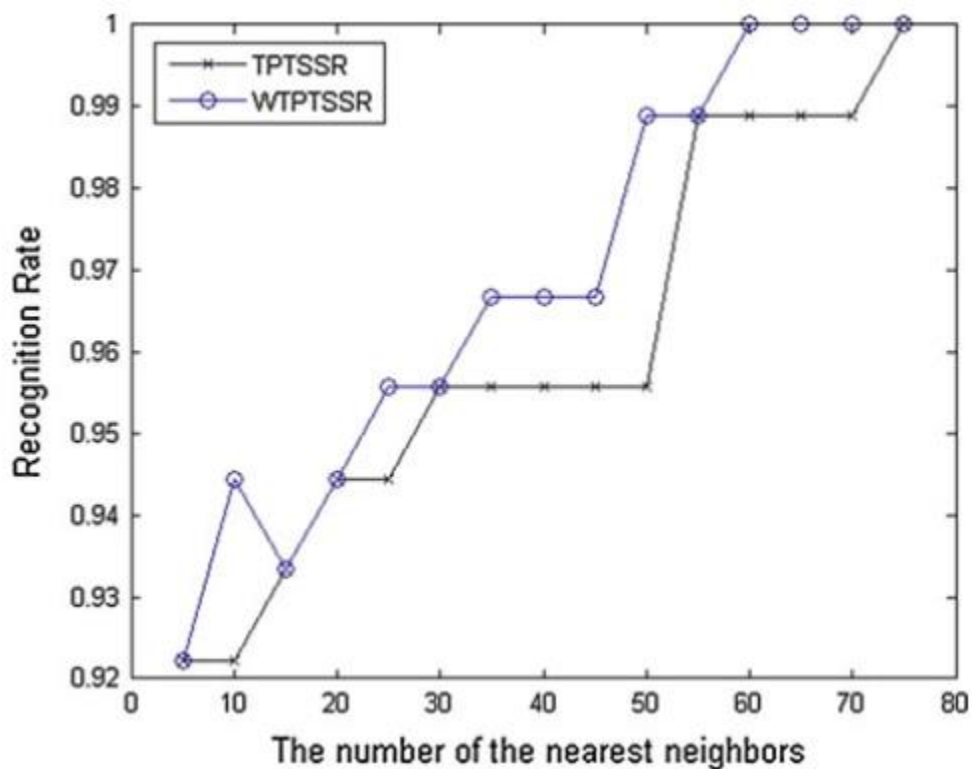
۴-۵- تست با استفاده از دیتابیس چهره Yale

دیتابیس چهره Yale شامل ۱۶۵ تصویر از ۱۵ فرد مختلف می باشد که از هر فرد ۱۱ تصویر با حالت های مختلف چهره و شرایط روشنایی مختلف وجود دارد. در این تست هر تصویر به صورت دستی برش داده شده و به ابعاد ۱۰۰ در ۸۰ پیکسل تغییر سایز داده می شود. شکل ۴-۶ یک نمونه تصاویر یک شخص را از این دیتابیس نشان می دهد.



شکل ۴-۶: نمونه تصویر چهره از دیتابیس Yale. شکل ۷-۴

در این تست پنج تصویر از هر شخص به صورت تصادفی به عنوان نمونه های آموزشی و شش تصویر باقیمانده به عنوان نمونه های تست در نظر گرفته می شوند و تست به طور مشابه ۱۰ بار تکرار می شود. شکل ۴-۷ نرخ شناسایی هر روش را نسبت به تعداد نزدیک ترین همسایگی ها نشان می دهد. جدول ۴-۳ ماکزیمم نرخ شناسایی هر روش و نرخ متناظر و با تعداد نزدیک ترین همایگی ها را نشان می دهد. همان زور که از شکل ۴-۷ دیده می شود هر دو الگوریتم WTPTSSR و TPTSSR نرخ شناسایی قابل توجهی دارند. از جدول ۴-۳ هم می توان دید که ماکزیمم نرخ شناسایی روش های WTPTSSR و TPTSSR وقتی تعداد نزدیک ترین همسایگی ها به ترتیب به ۷۵ و ۶۰ می رسد، به ۱۰۰ درصد خواهد رسید.



شکل ۴-۷: نرخ شناسایی TPTSSR و WTPTSSR بر حسب تغییرات تعداد نزدیک ترین همسایگی در دیتابیس Yale.

جدول ۴-۳: بیشینه نرخ شناسایی الگوریتم های مختلف در دیتابیس Yale.

Method	PCA	TPTSSR	WTPTSSR
Recognition rate	87.78 % \pm 2.5	100 % \pm 2	100 % \pm 1.5
Dimension or the number of the neighbors	41(D)	75(N)	60(N)

۴-۶- نتایج تست بر روی دیتابیس FERET

دیتابیس [۲۷] FERET به طور کل شامل ۱۳۵۳۹ تصویر چهره از ۱۵۶۵ نفر می باشد. تصاویر از نظر اندازه، حالت، روشنایی، ژست و سن با یکدیگر فرق می کنند. در این تست ۱۴۰۰ تصویر از ۲۰۰ شخص (هر شخص هفت تصویر دارد) انتخاب و به ۸۰ در ۸۰ پیکسل برش داده شد. شکل ۴-۸ تصاویر مربوط به یک فرد را در این دیتابیس نشان می دهد.

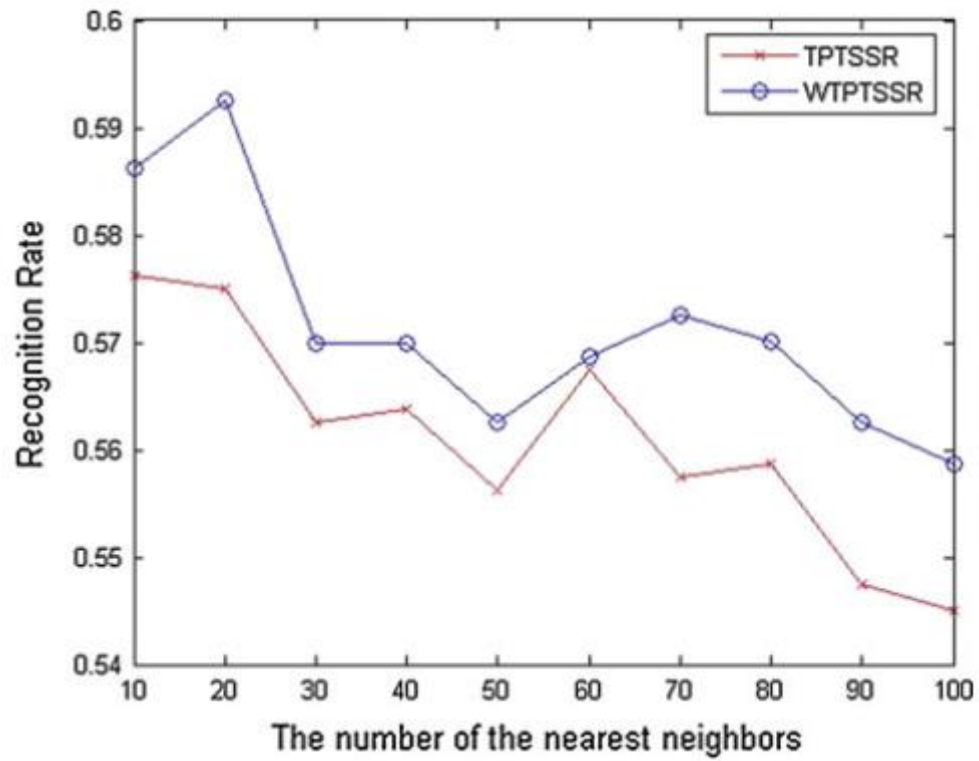


شکل ۴-۸: نمونه تصویر چهره یک فرد از دیتابیس FERET.

در تست انجام شده، سه تصویر اول هر شخص به عنوان نمونه های آموزشی و چهار تصویر باقیمانده به عنوان نمونه های تست در نظر گرفته می شود. در این تست از الگوریتم PCA به منظور استخراج ویژگی نیز استفاده شده است. نتایج تست شناسایی چهره مربوطه در شکل ۴-۹ نشان داده شده است. جدول ۴-۴ نیز ماکزیمم نرخ شناسایی هر روش و بعد یا تعداد نزدیک ترین همسایه متناظر با هر روش را نشان می دهد. همان طور که دیده می شود WTPPTSSR همواره عملکرد بهتری نسبت به الگوریتم TPTSSR مستقل از ابعاد ارائه کرده است. جدول ۴-۴ نیز نشان می دهد که WTPPTSSR بهترین عملکرد را داشته است.

جدول ۴-۴: پیشینه نرخ شناسایی الگوریتم های مختلف در دیتابیس FERET.

Method	PCA	TPTSSR	WTPPTSSR
Recognition rate	44.87 %	57.62 %	59.25 %
Dimension or the number of the neighbors	67(D)	10(N)	20(N)



شکل ۴-۹: نرخ شناسایی روش های TPTSSR و WTPTSSR بر حسب پارامتر M در دیتابیس FERET.

فصل ۵- نتیجه گیری

در مقاله ارائه شده نویسنده ابتدا به بررسی روش نمایش دو مرحله ای تنک نمونه تست (TPTSSR) می پردازد. این روش در یک طراحی دقیق استفاده می شود تا نمونه های آموزشی که بهترین انتخاب برای نمایش نمونه تست هستند به دست آیند. سپس جمع وزن دار نمونه های آموزشی انتخاب شده به منظور طبقه بندی نمونه های آموزشی مورد استفاده قرار می گیرد. TPTSSR از اطلاعات کلی تصویر بهره می گیرد و چنندات به اطلاعات محلی تصویر نمی پردازد. از آنجا که تاثیر قابل توجه اهمیت بالای اطلاعات محلی در طبقه بندی تصویر ثابت شده است، نویسنده با یک توسعه مستقیم از TPTSSR روش WTPTSSR را ارائه کرده است که امکان بهره گیری از اطلاعات محلی را به روش TPTSSR اضافه می کند. نتایج تست بر روی دیتابیس های چهره ORL، Yale، FERET و AR بهبود عملکرد روش ارائه شده را نسبت به سایر روش ها مشابه اثبات کرد.

فصل ٦-مراجع

١. Tian X, Tao D, Rui Y (٢٠١٢) Sparse transfer learning for interactive video search reranking. ACM Trans Multimed Comput Commun Appl ٨:٢٦.
doi:١٠.١١٤٥/٢٢٤٠١٣٦,٢٢٤٠١٣٩
٢. Gao X, Wang N, Tao D, Li X (٢٠١٢) Face sketch-phono synthesis and retrieval using sparse representation. IEEE Trans Circuits Syst Video Technol ٢٢(٨):١٢١٣-١٢١٦
٣. Tao D et al (٢٠٠٨) Bayesian tensor approach for ٣-D face modelling. IEEE Trans Circuits Syst Video Technol ١٨(١٠):١٣٩٧-١٤١٠
٤. Xu Y, Zhong A, Yang J, Zhang D (٢٠١٠) LPP solution schemes for use with face recognition. Pattern Recognit ٤٣:٤١٦٥-٤١٧٦
٥. Turk MA, Pentland AP (١٩٩١) Face recognition using eigenfaces. In: IEEE computer society conference on computer vision and, pattern recognition, ١٩٩١, pp. ٥٨٦-٥٩١
٦. Tao D, Li X, Maybank SJ (٢٠٠٧) General tensor discriminant analysis and gabor features for gait recognition. IEEE Trans Pattern Anal Mach Intell ٢٩(١٠):١٧٠٠-١٧١٥
٧. Guan N, Tao D, Luo Z, Shawe-Taylor J (٢٠١٢) Manhattann non-negative matrix factorization. J Mach Learn Res. <http://arxiv.org/abs/١٢٠٧.٣٤٣٨>
٨. Candès EJ, Li X, Ma Y, Wright J (٢٠١١) Robust principal component analysis? J ACM ٥٨(٣):١-٣٧
٩. Guan N, Tao D, Luo Z, Yuan B (٢٠١٢) NeNMF: an optimal gradient method for nonnegative matrix factorization. IEEE Trans Signal Process ٦٠(٦):٢٨٨٢-٢٨٩٨
١٠. Wright J, Ma Y, Mairal J et al. (٢٠٠٩) Sparse representation for computer vision and pattern recognition. In: Proceedings of IEEE, ٢٠٠٩, pp. ١-٨
١١. Wright J, Yang AY, Ganesh A et al (٢٠٠٩) Robust face recognition via sparse representation. IEEE Trans Pattern Anal Mach Intell ٣١:٢١٠-٢٢٧
١٢. Gao S, Tsang I, Chia L-T (٢٠١٠) Kernel sparse representation for image classification and face recognition. In: Daniilidis K, Maragos P, Paragios N (eds) Computer vision ECCV ٢٠١٠. Springer, Berlin, Heidelberg, pp ١-١٤
١٣. Yang M, Zhang L, Yang J, Zhang D (٢٠١١) Robust sparse coding for face recognition. In: IEEE international conference computer vision and pattern recognition, ٢٠١١, pp. ٦٢٥-٦٣٢
١٤. Lai Z, Jin Z, Yang J, Wong WK (٢٠١٠) Sparse local discriminant projections for feature extraction. In: Proceedings of ICPR, ٢٠١٠, pp. ٩٢٦-٩٢٩
١٥. Zhang L et al. (٢٠١١) Sparse representation or collaborative representation: which helps face recognition? In: ICCV ٢٠١١
١٦. Shi Q, Eriksson A, Hengel A, Shen C (٢٠١١) Is face recognition really a compressive sensing problem? In: CVPR ٢٠١١

17. Xu Y, Zhang D, Yang J, Yang J-Y (2011) A two-phase test sample sparse representation method for use with face recognition. *IEEE Trans Circuits Syst Video Technol* 21(9):1200-1212
18. Xu Y, Zuo W, Fan Z (2011) Supervised sparse presentation method with a heuristic strategy and face recognition experiments. *Neurocomputing* 74:120-131
19. Xu Y, Zhu Q (2013) A simple and fast representation-based face recognition method. *Neural Comput Appl*. doi:10.1007/s00521-012-0833-0
20. Tao D, Li X, Wu X, Maybank SJ (2009) Geometric mean for subspace selection. *IEEE Trans Pattern Anal Mach Intell* 31(2):260-274
21. Wang J, Yang J, Yu K, Lv F, Huang T, Gong Y (2010) Locality-constrained linear coding for image classification. In: *IEEE computer society conference on computer vision and, pattern recognition*, 2010, pp. 3360-3367
22. Lu C-Y, Min H, Gui J, Zhu L, Lei Y-K (2010) Face recognition via weighted sparse representation. *J Vis Commun Image Represent* 21:111
23. Yu K, Zhang T, Gong Y (2009) Nonlinear learning using local coordinate coding. *Adv Neural Inf Process Syst* 22:2223-2231
24. Yu K, Zhang T (2010) Improved local coordinate coding using local tangents. In: *International conference on machine learning*, 2010, pp. 1210-1222
25. Samaria F, Harter A (1994) Parameterisation of a stochastic model for human face identification. In: *Second IEEE workshop on applications of computer vision*, Sarasota
26. Martinez A, Benavente R (1998) The AR face database. In: *CVC technical report no.* 24
27. Phillips PJ, Wechsler H et al (1998) The FERET database and evaluation procedure for face-recognition algorithms. *Image Vis Comput* 16(5):290-306

فصل ۷- پیاده سازی نرم افزار

در این بخش به دنبال پیاده سازی مقاله ارائه شده با نرم افزار متلب هستیم. ما پیاده سازی را بر روی دیتابیس ORL و Yale اعمال می کنیم اگرچه هر یک از دیتابیس های دیگر ارائه شده نیز می توانند انتخاب شوند. به منظور سادگی از تصاویر تغییرساز شده ۳۲ در ۳۲ پیکسل برای دیتابیس ORL و ۶۴ در ۶۴ پیکسل برای دیتابیس Yale استفاده می کنیم. برای هر دیتابیس یک ماتریس برای ذخیره سازی تصاویر و یک بردار برای برچسب کلاس ها در نظر گرفته شده است. دیتابیس ORL شامل ۴۰۰ تصویر از ۴۰ فرد مختلف یعنی به ازای هر فرد ۱۰ تصویر می باشد. با پشت سر هم قرار دادن ستون های هر تصویر یک سطر از ماتریس تصویر این دیتابیس ساخته می شود. بنابراین ماتریس تصاویر دیتابیس ORL یک ماتریس ۴۰۰ در ۱۰۲۴ است که در آن هر سطر معادل یک تصویر و هر ستون معادل یک پیکسل از تصاویر مختلف می باشد. بدیهی است که بردار برچسب کلاس ها نیز یک بردار ۴۰۰ در ۱ است. مقادیر این بردار اعداد صحیح ۱ تا ۴۰ می باشند. به طور مشابه برای دیتابیس Yale ماتریس تصاویر یک ماتریس ۱۶۵ در ۱۰۲۴ و بردار برچسب کلاس ها یک بردار ۱۶۵ تایی می باشد.

در ادامه پیاده سازی را برای دیتابیس ORL توضیح خواهیم داد که برای دیتابیس Yale هم با اندکی تغییر قابل استفاده است. البته کدهای مربوط به پیاده سازی ها به همراه فایل گزارش قرار داده شده اند. در ابتدا پارامترهای لازم را که در مقاله به آن ها اشاره شده بود تعیین می کنیم. همان طور که در کدهای مربوط به این بخش دیده می شود. پارامتر `num_tests` مربوط به تعداد تکرار تست می باشد که در اینجا برابر ۱۰ قرار داده شده است. ما برای پارامتر `la_p` مقدار ۰,۰۱ را در نظر گرفتیم. این پارامتر در واقع همان پارامتر `k` برای محاسبه فاصله در ماتریس وزن (رابطه ۸) است. مقدار صفر برای این پارامتر الگوریتم را به الگوریتم TPTSSR تبدیل می کند. پارامتر `reg_p` همان ضریب لاگرانژ در رابطه (۹) است و همان طور که در مقاله به آن اشاره شده است مقدار آن برابر ۱,۳ در نظر گرفته شده است. مقدار پارامتر `sigma` یک مقدار مثبت کوچک در رابطه (۵) است و مقدار آن برابر ۰,۰۱ تعیین شده است. همان طور که پیشتر اشاره شد در دیتابیس ORL از هر فرد ۱۰ تصویر داریم که ۵ تصویر را برای آموزش و ۵ تصویر را برای تست در نظر می گیریم بنابراین تعداد نمونه های آموزشی `num_training_samples` برای هر فرد ۵ می باشد. تعداد نزدیک ترین همسایگی برای نمونه تست نیز برابر $M = 60$ تعیین شده است.

```

%% Defining parameters
num_tests = ۱۰;           % number of test
la_p = ۰,۰۱;              % local adaptor parameter
reg_p = ۱,۳;              % regularization parameter
sigma = ۰,۰۱;            % small positive Constant
num_training_samples = ۵; % number of training sample
M = ۱۰;                   % M nearest neighbor

```

کد اصلی برنامه که شامل اعمال الگوریتم بر روی نمونه های تست است در یک حلقه با تعداد تکرار num_tests و اندیس J قرار دارد. در داخل حلقه در ابتدا با استفاده از تابع rnd_selection نیمی از تعداد نمونه ها به صورت تصادفی به عنوان نمونه های آموزشی و سایر نمونه ها به عنوان نمونه های تست در نظر گرفته می شوند.

```

[tr_idx, te_idx]=rnd_selection(Labels, num_training_samples);
test_samples=Images(te_idx,:);test_labels= Labels(te_idx);
train_samples=Images(tr_idx,:);train_labels= Labels(tr_idx);

```

در ادامه محاسبات لازم می بایست بر روی هر یک از نمونه های تست صورت می گیرد. و لذا از یک حلقه به تعداد نمونه های تست و با اندیس i استفاده می کنیم. سپس با استفاده از کد های زیر برای هر نمونه تست γ ماتریس وزن W را تشکیل می دهیم.

```

y =test_samples(:, i);
for k = ۱:num_train
    W(k,k) = sum(abs(y - train_samples(:,k))) ^la_p;

end

```

با محاسبه ماتریس وزن W ، وزن های لازم X برای نمایش نمونه تست γ مطابق رابطه (۱۰) به صورت زیر قابل محاسبه است.

```

X=(train_samples'*train_samples+reg_p*(W'*W))\ (train_samples'*y);

```

حال به منظور ارزیابی میزان شراکت هر کلاس در بازسازی نمونه تست چنان چه پیشتر گفته شد مطابق رابطه (۶) مقدار $CONS$ را به دست می آوریم. یک مقدار کوچکتر برای عبارت CON_i به معنای سهم بیشتر در نمایش نمونه تست است.

```

for k = ۱:num_train
    cons(k) = sqrt( sum((y - X(k)*train_samples(:,k)) .^۲));
end

```

در ادامه M نمونه آموزشی نزدیک ترین همسایه نمونه تست را به دست می آوریم و مجدداً مشابه روابط (۴)، (۵) و (۶) محاسبات انجام شده را این بار برای حالتی که تنها M نمونه آزمایشی برای نمایش نمونه تست در اختیار داریم انجام می دهیم. ابتدا مقادیر \tilde{A} و \tilde{X} را به دست می آوریم و در نهایت به منظور ارزیابی میزان شراکت کلاس های مختلف در بازسازی نمونه تست مجدداً از رابطه (۶) استفاده می کنیم. و در نهایت به منظور طبقه بندی نمونه تست کلاس با کمترین مقدار $cons$ را انتخاب می کنیم.

```
cons = sortrows([cons, (1:num_train)'], 1);
Ahat = train_samples(:, cons(1:M,2));
Xhat = (Ahat' * Ahat + sigma * eye(M)) \ (Ahat' * y);
hat_class = train_labels(cons(1:M,2));
cons = zeros(num_classes, 1);
cons_data = zeros(num_features, num_classes);
for k = 1:M
    cons_data(:,hat_class(k)) =
        cons_data(:,hat_class(k)) + Xhat(k) * Ahat(:,k);
end
for k = 1:num_classes
    cons(k) = sqrt( sum((y - cons_data(:,k)) .^2));
end
[~, predicted_labels(i)] = min(cons);
accuracy = size((find(test_labels==predicted_labels)),1)/size(test_labels,1);
```

در نهایت با بررسی تعداد طبقه بندی های صحیح می توانیم دقت شناسایی نسبی را به دست آوریم و از آنجا انتخاب نمونه های آموزشی و نمونه تست به صورت تصادفی انجام می شود این مقدار می تواند در اجرای بعدی اندکی تغییر کند. برای مثال بعد از اجرای کد برای دیتابیس ORL نتیجه زیر حاصل شد:

accuracy: %۹۳,۸۵ +- %۱,۶۶۷۵

که عدد اول مربوط به دقت و عدد دوم مقدار انحراف از معیار می باشد. چنانچه این مقادیر را با مقادیر محاسبه شده در مقاله در شکل ۳ و جدول ۱ مقایسه کنیم، خواهیم دید اگرچه تصاویر استفاده شده کاهش سایز قابل توجهی داشته اند ولی نتایج به دست آمده کاملاً متناسب با نتایج به دست آمده در مقاله می باشند. حال تست را برای دیتابیس Yale انجام می دهیم. در این تست ما مشابه مقاله از ۱۱ تصویر هر فرد ۵ تصویر را به عنوان نمونه

آموزشی و ۶ تصویر را به عنوان نمونه تست در نظر می گیریم. پس از ۱۰ بار انجام تست بر روی این دیتابیس خواهیم داشت:

accuracy: 88.4444 ± 3.664 %

همان طور که دیده می شود دقت به دست آمده اندکی با مقداری که در مقاله ارائه شده است متفاوت می باشد که ممکن است ناشی از تغییر ساین اعمال شده باشد.