

طراحی یک سیستم شناسایی چهره برخط با استفاده از پردازنده‌های گرافیکی

محسن بیگلری^۱ فائزه میرزائی^۱ حسین ابراهیم‌پور کومله^۲

^۱دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شاهرود، شاهرود، ایران
^۲دانشکده مهندسی کامپیوتر، دانشگاه کاشان، کاشان، ایران

چکیده

شناسایی چهره در سه دهه اخیر مورد توجه بسیاری از محققان قرار گرفته است. شناسایی چهره در محیط‌های کنترل شده و در تصاویر ثابت، پیشرفت چشمگیری داشته است. هر چند هنوز در کاربردهای برخط و در محیط‌های کنترل نشده، دارای چالش‌های بسیاری است.

در این مقاله، روشی جدید برای شناسایی چهره ارائه گردیده که تا حدود زیادی مقاوم به تغییرات حالت چهره و روشنایی در تصاویر است. از الگوریتم الگوی دودویی محلی (LBP) برای استخراج ویژگی پایدار به تغییرات یکنواخت روشنایی و حالت چهره و از الگوریتم SQI برای خنثی‌سازی تغییرات نور بهره برده شده است. برای حصول کارایی بالاتر، هر دوی این الگوریتم‌ها توسعه داده شدند. علاوه بر این، برای افزایش هر چه بیشتر سرعت سیستم، از پردازنده‌های گرافیکی (GPU) استفاده گردیده است؛ هر جا که استفاده از GPU مقرون به صرفه نبوده، موزایی‌سازی با CPU صورت گرفته است.

نتایج آزمایشات بر روی دو پایگاه داده YaleB و CMU-PIE نشان از برتری دقت روش پیشنهادی نسبت به سایر روش‌های مورد مقایسه دارد. از این جمله می‌توان به روش‌های SQI، QI، QIR، MQI، DMQI، DSFQI، PCA و LDA اشاره کرد. موزایی‌سازی موجب افزایش تقریبی ۱۹ برابری سرعت سیستم گشته است؛ به شکلی که پردازش یک تصویر با ابعاد 325×205 در زمان ۲۱۳ میلی ثانیه انجام می‌شود.

کلمات کلیدی: تغییرات حالت چهره، تغییرات روشنایی، GPU، الگوریتم موزایی، LBP، SQI.

۱- مقدمه

چالش‌هایی همچون تغییرات روشنایی، حالت و ژست، موجب کاهش کارایی قابل توجه آن‌ها می‌شود.

روش‌های بسیاری برای پشت سر گذاشتن این چالش‌ها ارائه شده است؛ برای نمونه در [۴-۹] چندین الگوریتم برای شناسایی چهره مقاوم به تغییرات روشنایی مورد مطالعه قرار گرفته است؛ روش‌های QI [۶]، SQI [۷] و MQI [۸] از این جمله بوده‌اند. دو روش SQI^۵ و MQI به دلیل موفقیت در نتایج ارائه شده، پایه و اساس بسیاری از روش‌های جدیدتر قرار گرفته‌اند. روش‌های DMQI [۱۰]، DSFQI [۴] نمونه‌هایی از این موارد می‌باشند. مراجع [۱۱، ۱۲] روش‌هایی برای غلبه بر مشکلات حاصل از تغییرات ژست ارائه کرده‌اند. تغییرات ژست از مواردی است که در محیط‌های کنترل نشده بسیار معمول است. به صورت کلی، روش‌های جزئی‌نگر^۶ مانند LBP [۱۳] و EBGm [۱۴] نسبت به روش‌های کلی‌نگر^۷ مانند

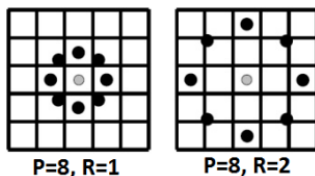
شناسایی چهره در سه دهه اخیر، در مرکز توجه بسیاری از محققان قرار گرفته است. در حال حاضر، سیستم‌های شناسایی چهره در محیط‌های کنترل شده، اغلب عملکرد مناسبی از خود نشان می‌دهند. اغلب این سیستم‌ها با تصاویر ثابت کار می‌کنند. قابلیت پردازش تصاویر متحرک، نیاز به سرعت پردازش برخط بالایی دارد. برخی از چالش‌های اصلی موجود در مسیر سیستم‌های شناسایی چهره عبارتند از: تغییرات حالت^۱ چهره، تغییرات روشنایی^۲، تغییرات ژست^۳ و انسداد^۴ بخشی از چهره.

برای نمونه می‌توان به الگوریتم‌های PCA [۱]، LDA [۲] و ICA [۳] اشاره کرد که در شرایط کنترل شده کارایی مناسبی دارند؛ اما در محیط‌های واقعی،

به دلیل ویژگی‌های ساختاری چهره انسان، عملگر پایه به شکلی توسعه داده شد که بتواند از همسایه‌های دایره‌ای شکل نیز بهره برد [۲۰]. نسخه‌ی توسعه یافته از تعداد نقاط نمونه‌برداری متفاوتی بر روی یک دایره به شعاع دلخواه استفاده می‌کند. نماد $LBP_{P,R}$ برای توصیف این عملگر با تعداد نقاط نمونه‌برداری P بر روی دایره‌ای به شعاع R اشاره دارد. شکل ۲ چند نمونه متفاوت از این عملگر را نشان می‌دهد. فرمول (۱) طرز کار عملگر LBP توسعه یافته را نشان داده است. در فرمول (۱)، $LBP_{P,R}(x,y)$ مقدار دهمی است که برای هر پیکسل (x,y) محاسبه می‌شود. g_c مقدار خاکستری پیکسل مرکزی و g_i مقدار خاکستری پیکسل همسایه است. روش محاسبه $s(x)$ نیز در فرمول (۲) آورده شده است.

$$LBP_{P,R}(x,y) = \sum_{i=0}^{P-1} s(g_i - g_c) 2^i \quad (1)$$

$$s(x) = \begin{cases} 1 & x > 0 \\ 0 & x \leq 0 \end{cases} \quad (2)$$



شکل ۲- عملگر LBP توسعه یافته با P و R متفاوت

توسعه‌ی دیگری که بر روی عملگر LBP صورت گرفته، معرفی الگوهای یکنواخت^{۱۲} بوده است [۲۰]. یک الگوی دودویی، یکنواخت نامیده می‌شود اگر حداکثر دارای دو تغییر بیتی از صفر به یک یا برعکس باشد. عملگر توسعه یافته یکنواخت با نماد $LBP_{P,R}^{u2}$ نشان داده می‌شود. در مرجع [۲۰] به صورت تجربی نشان داده شده که کارایی عملگر $LBP_{8,1}^{u2}$ یکنواخت تقریباً برابر با عملگر $LBP_{8,1}$ است در صورتی که تعداد این الگوهای یکنواخت نسبت به تعداد کل الگوها برابر با $\frac{58}{256}$ است که کاهش ۷۷ درصدی را به همراه دارد. در محاسبه‌ی هیستوگرام الگوهای یکنواخت، معمولاً الگوهای غیریکنواخت همگی در یک ستون واحد قرار داده می‌شوند. در نتیجه در مجموع ۵۹ ستون خواهیم داشت.

۳- تصویر خود خارج قسمت

الگوریتم خود خارج قسمت یا SQI برای محاسبه تصویر مستقل از تغییرات روشنایی با استفاده از تنها خود تصویر ارائه شده است [۷]. فرمول (۳) روش کار این الگوریتم را نشان می‌دهد. $I(x,y)$ تصویر ورودی و $S(x,y)$ نسخه‌ی یکنواخت^{۱۴} شده‌ی آن می‌باشد.

$$Q(x,y) = \frac{I(x,y)}{S(x,y)} = \frac{I(x,y)}{F(x,y) * I(x,y)} \quad (3)$$

تابع F هسته‌ی یکنواخت کننده و * عملگر کانولوشن^{۱۵} است. معمولاً از فیلتر گوسین وزن دار برای یکنواخت‌سازی تصویر استفاده می‌شود. $Q(x,y)$ نسخه‌ی مستقل از روشنایی تصویر ورودی است؛ خروجی این روش بر روی یک تصویر نمونه در شکل ۳ نمایش داده شده است.

PCA و LDA عملکرد بهتری دارند [۱۵]. تغییرات حالت چهره، اهمیت ویژه‌ای در یک سیستم شناسایی چهره دارند؛ از این رو، تحقیقات گسترده‌ای در این زمینه نیز صورت گرفته است [۱۶-۱۸]. با وجود همه‌ی این مطالعات، شناسایی چهره در محیط‌های واقعی به دلیل وجود چالش‌های مورد اشاره، هنوز یک موضوع تحقیقاتی باز به شمار می‌رود.

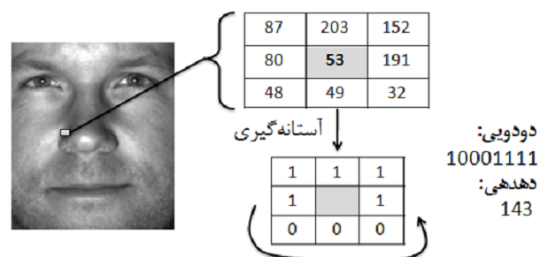
در این مقاله، روشی جدید برای شناسایی چهره پایدار به تغییرات حالت چهره و روشنایی ارائه شده است. این روش بر پایه‌ی دو الگوریتم SQI و LBP^A بنا نهاده شده است. از SQI برای از بین بردن تاثیر تغییرات روشنایی و از LBP برای استخراج ویژگی مقاوم به تغییرات حالت چهره و روشنایی یکنواخت خاکستری استفاده گردیده است. ساختار این دو الگوریتم به شکلی تغییر داده شده که بیشترین کارایی را برای سیستم پیشنهادی داشته باشند. استفاده از چند الگوریتم به صورت ترکیبی، هرچند موجب افزایش کارایی می‌شود ولی کاهش سرعت محسوسی را در پی خواهد داشت. به همین دلیل، با استفاده از پردازنده‌های گرافیکی^{۱۶} و CPU، سیستم پیشنهادی به صورت موازی پیاده‌سازی شده و ساختار مورد استفاده و افزایش سرعت حاصل از آن نیز گزارش گردیده است.

در پایان، دقت سیستم بر روی دو پایگاه داده شناخته شده‌ی YaleB و CMU-PIE مورد آزمایش قرار گرفته و کارایی آن با روش‌هایی چون SQI، QIR [۱۹]، MQI، DMQI، DSFQI، PCA و LDA مقایسه شده است. در ادامه، در بخش دوم و سوم، آشنایی مختصری با الگوریتم‌های LBP و SQI ایجاد شده است. در بخش چهارم، سیستم پیشنهادی ارائه گردیده است. موازی‌سازی سیستم پیشنهادی در بخش پنجم به تفصیل توضیح داده شده است؛ در ابتدای همین بخش، توضیح مختصری در مورد GPU و کودا داده شده است. نتیجه‌ی آزمایشات و نتیجه‌گیری نیز در دو بخش آخر ارائه گشته است.

۲- الگوی دودویی محلی

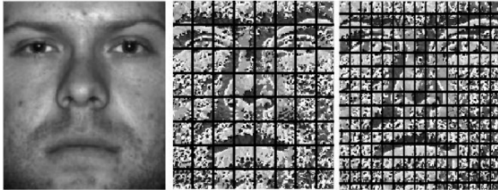
عملگر الگوی دودویی محلی یا LBP توسط أجالا^{۱۱} و سایرین برای توصیف بافت^{۱۱} ارائه شد [۱۳]. توصیف کننده‌های محلی چهره انسان به دلیل پایداریشان نسبت به تغییرات ژست و حالت، توجه زیادی را به خود جلب کرده‌اند. عملگر LBP یکی از بهترین توصیف‌گرهای محلی است. علاوه بر پایداری نسبت به تغییرات ژست و حالت، این عملگر نسبت به تغییرات یکنواخت خاکستری^{۱۲} نیز مقاوم است.

عملگر LBP پایه از یک پنجره 3×3 تشکیل شده است؛ مرکز این پنجره بر روی هر یک از پیکسل‌های تصویر قرار داده شده و اختلاف مقادیر پیکسل‌ها تا مرکز، برای هر یک از همسایه‌ها محاسبه می‌شود. نتیجه به شکل یک رشته‌ی دودویی در آورده شده و سپس به معادل دهمی آن تبدیل می‌شود. در نهایت، یک هیستوگرام به عنوان بردار ویژگی تولید خواهد شد؛ که در آن، هر یک از این مقادیر دهمی، یک ستون را تشکیل می‌دهند. شکل ۱ جزئیات بیشتری را نمایش داده است.



شکل ۱- عملگر LBP پایه

ویژگی‌های استخراج شده دو چندان خواهد شد. بنابراین، دو سطح ویژگی از تصویر ورودی استخراج گشته و سپس به شکل بردار ویژگی‌های نهایی ادغام می‌گردند (شکل ۴).



شکل ۴- استخراج ویژگی با استفاده از عملگر LBP به صورت چندبخشی 8×8 و 15×15



شکل ۳- تصویر مستقل از روشنایی بدست آمده توسط روش SQI

روش MQI [۸] که بعد از این روش ارائه گردیده، از عملگرهای مورفولوژیکی^{۱۶} برای محاسبه تصویر مستقل از روشنایی بهره می‌برد. این روش سرعت بالاتری دارد ولی دقت آن کمی پایین‌تر از روش SQI است.

۳-۳- ساخت بردار ویژگی نهایی و طبقه‌بندی وزن‌دار

برای تشکیل بردار ویژگی‌های نهایی، دو مرحله‌ی زیر به ترتیب انجام می‌شوند:

۱. هیستوگرام‌های استخراج شده از هر بخش، در کنار هم قرار داده می‌شوند.
۲. سپس هیستوگرام‌های بدست آمده از عملگر ۸ بخشی و ۱۵ بخشی (مرحله ۱)، در هم ادغام می‌شوند.

با داشتن ویژگی‌های مناسب، نکته‌ی حائز اهمیت بعدی، یک روش مناسب برای مقایسه‌ی ویژگی‌ها در مرحله‌ی طبقه‌بندی^{۱۷} است. اعمال وزن در مرحله‌ی مقایسه‌ی بردارهای ویژگی صورت می‌گیرد. برای مقایسه‌ی ویژگی‌ها از نسخه‌ی وزن‌دار روش مربع چای^{۱۸} (فرمول (۴)) استفاده شده است.

$$\chi_w^2(H1, H2) = \sum_{i,j} w_j \frac{(H1_{i,j} - H2_{i,j})^2}{(H1_{i,j} + H2_{i,j})} \quad (4)$$

$H1$ و $H2$ هیستوگرام ورودی و هیستوگرام ثبت شده در سیستم هستند. اندیس‌های i و j به ستون i ام از بردار ویژگی مربوط به بخش j اشاره دارند و w_j وزن بخش j ام می‌باشد. بخش‌های مختلف چهره، تاثیر یکسانی در شناسایی چهره ندارند؛ برای پررنگ‌تر کردن تاثیر بخش‌هایی چون بینی، چشم‌ها، دهان و ابروها، از وزن‌های نشان داده شده در شکل ۵ استفاده شده است. در این تصویر، وزن‌های مربوط به هر یک از $8 \times 8 = 64$ بخش که به صورت تجربی بدست آمده‌اند، نمایش داده شده است [۲۱].



شکل ۵- وزن‌های استفاده شده در مقایسه بردارهای ویژگی. رنگ تیره‌تر روشن‌تر به معنی وزن بیشتر است [۲۱]

۵- مقدمه‌ای بر پردازنده‌های گرافیکی و کودا

استفاده از پردازنده‌های چند هسته‌ای تلاشی در جهت رفع مشکل محدودیت زمانی در نرم‌افزارها بوده است [۲۲]؛ اما این پردازنده‌ها قیمت بالایی دارند و حداکثر افزایش راندمان توسط آن‌ها محدود به تعداد هسته‌های آن‌ها است. GPUها که اخیراً توجه بسیاری از محققان را به خود جلب کرده‌اند، ابزاری بسیار

۴- سیستم پیشنهادی

سیستم پیشنهادی از سه لایه تشکیل شده است. در لایه‌ی اول، با استفاده از SQI تاثیر روشنایی بر تصویر خنثی می‌شود. در لایه‌ی دوم، عملگر LBP یکنواخت به صورت چند بخشی به تصویر اعمال شده و ویژگی‌های مورد انتظار را از هر یک از بخش‌ها استخراج می‌کند. در لایه‌ی پایانی، ویژگی‌ها یا در واقع هیستوگرام‌های استخراج شده در کنار هم قرار داده شده و بردار ویژگی نهایی را تشکیل می‌دهند. در ادامه هر یک از لایه‌ها به اختصار توضیح داده شده‌اند.

۴-۱- پیش پردازش

دو فاکتور مهم در روش SQI اندازه پنجره و مقدار انحراف معیار (σ) مربوط به فیلتر گوسین مورد استفاده در آن است. بر اساس آزمایشات صورت گرفته در [۲۱]، اندازه پنجره 15×15 و مقدار انحراف معیار متغیر برای آموزش و آزمایش سیستم انتخاب گردید. در مرحله‌ی آموزش از مقدار انحراف معیار 0.7 و در مرحله‌ی آزمایش از مقدار 0.3 استفاده شد. مقادیر کوچک‌تر انحراف معیار با توجه به فرمول (۳)، تصویر مستقل از روشنایی یکنواخت‌تری را بدست می‌دهند. در نتیجه با استفاده از انحراف معیار بزرگ‌تر در مرحله‌ی آموزش، تصویری با جزئیات بیشتر در سیستم ثبت می‌شود؛ استفاده از تصویر یکنواخت‌تر در مرحله‌ی آزمایش، کارایی را افزایش خواهد داد.

۴-۲- استخراج ویژگی

برای استخراج ویژگی، از عملگر LBP توسعه یافته‌ی یکنواخت به صورت چندبخشی استفاده شده است ($LBP_{g,2}^{m,2}$). اعمال این روش به صورت چندبخشی برای شناسایی چهره الزامی است؛ به این ترتیب، ویژگی‌های محلی از هر بخش استخراج می‌گردد. از طرفی تجمیع ویژگی‌های محلی در بردار ویژگی‌های نهایی، ویژگی‌های عمومی را تشکیل می‌دهد. یکی از مزایای دیگر استخراج ویژگی به صورت چندبخشی، امکان وزن‌دهی به بخش‌های مهم‌تر از قبیل چشم‌ها، دهان، بینی و ابروها است.

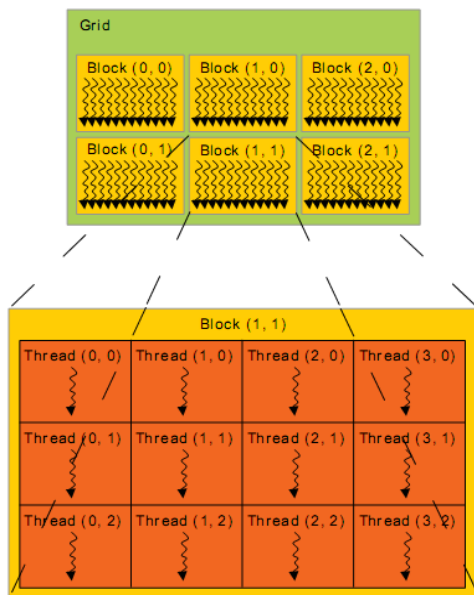
با توجه به تصاویر دو پایگاه داده‌ی مورد آزمایش، تعداد بخش‌های 15×15 انتخاب گردید. کارایی این تعداد بخش در [۲۱] به صورت تجربی اثبات شده است. استفاده از تنها همین تعداد بخش، بیشتر بر روی ویژگی‌های محلی تمرکز می‌کند. برای استخراج ویژگی‌های سطح بالاتر، از تعداد بخش‌های 8×8 نیز به صورت موازی بهره برده‌ایم. به دلیل بخش‌پذیر نبودن این دو مقدار (۱۵ و ۸)، کارایی

OS:	Windows 7
Processor:	Intel Core2 Quad CPU Q9550@2.83 GHz
RAM:	8 GB
Graphics:	GeForce GT 430
CUDA ComputeCapability:	2.1

زمان کل سیستم در حالت ترتیبی برای ابعاد اشاره شده، تقریباً برابر با ۴ ثانیه است؛ در این بخش، با ارائه‌ی ساختاری موازی و پیاده‌سازی آن، این زمان را کاهش داده و سیستم را برای کاربردهای برخط ایده‌آل ساخته‌ایم. در سکوی کودا، تعداد زیادی نخ^{۲۲} اجرایی در قالب مجموعه‌ای از بلوک‌ها در اختیار توسعه دهنده قرار می‌گیرد. برای اجرای یک تابع نوشته شده در کودا، می‌توان تعداد بلوک‌ها و تعداد نخ‌های موجود در هر بلوک را مشخص ساخت. شکل ۸ یک نمونه چیدمان بلوک‌ها و نخ‌های موجود در آن را در دو بعد نشان می‌دهد.

جدول ۱- بخش‌های مختلف سیستم پیشنهادی و زمان اجرای هر یک

مرحله	روش	مدت زمان (s)
۱-۱	پیش‌پردازش (تبدیل رنگی به خاکستری)	0.001
۲-۱	پیش‌پردازش (روش SQR)	3.800
۱	پیش‌پردازش (کل)	3.801
۱-۲	استخراج ویژگی‌ها (۸ LBP بخشی)	0.038
۲-۲	استخراج ویژگی‌ها (۱۵ LBP بخشی)	0.086
۲	استخراج ویژگی‌ها (کل)	0.124
۳	مقایسه ویژگی‌ها	0.016
مجموع	کل سیستم	3.951



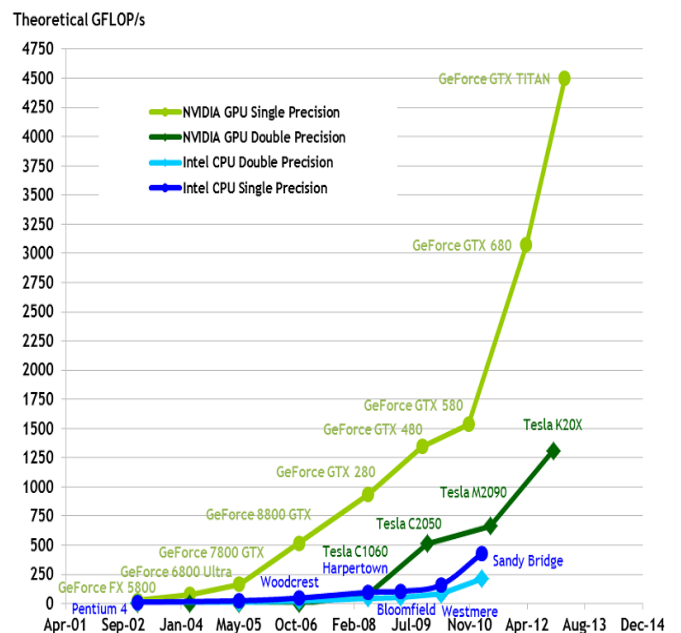
شکل ۸- یک نمونه چیدمان بلوک‌ها و نخ‌های GPU در دو بعد [۲۳]

برای افزایش سرعت سیستم، بخش‌های ۱ و ۳ از جدول ۱ با استفاده از GPU و بخش ۲ با استفاده از CPU موازی‌سازی شد. برای اجرای یک برنامه با استفاده از GPU باید ابتدا داده‌های موردنیاز را به حافظه‌ی GPU منتقل کرد و بعد از اتمام اجرا، داده‌های تولید شده را به حافظه‌ی CPU بازگرداند. بنابراین، در صورتی که زمان جابه‌جایی داده‌ها از سود حاصل از موازی‌سازی برنامه بیشتر باشد، موازی‌سازی به صرفه نخواهد بود. برای مثال موازی‌سازی LBP با GPU به دلیل ساده بودن عملیات انجام شده توسط آن نسبت به حجم داده‌های انتقالی، به صرفه نیست.

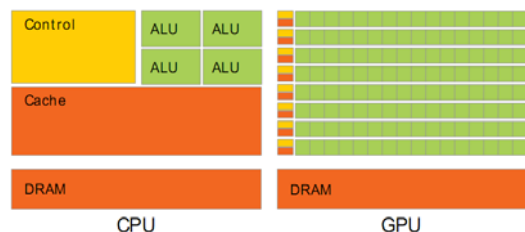
مناسب برای پیاده‌سازی الگوریتم‌های موازی می‌باشند. مزیت GPUها، کارایی بسیار بالا، قیمت مناسب و در دسترس بودن آنها است.

در شکل ۶ مقایسه‌ای بین مدل‌های مختلف CPU و GPU برای انجام عملیات ممیز شناور صورت گرفته است. همان‌طور که در این شکل قابل ملاحظه است، اختلاف عمده‌ای در توانایی انجام عملیات ممیز شناور بین CPU و GPU وجود دارد. علت برتری محسوس GPU، طراحی اختصاصی آن برای انجام عملیات محاسباتی به صورت موازی است؛ به شکلی که تعداد واحدهای پردازشی در GPU نسبت به واحدهای کش^{۱۹} و کنترل جریان^{۲۰} بسیار بیشتر هستند؛ در صورتی که در CPU چنین نیست. شکل ۷ گویای این مطلب است.

شرکت NVIDIA در سال ۲۰۰۶، سکوی کودا^{۲۱} را برای انجام محاسبات موازی حجیم و کارا بر روی GPUهای ساخت خود که در کارت‌های گرافیک این شرکت قرار داشتند، ارائه کرد. با استفاده از کودا، توسعه دهندگان قادر خواهند بود که برنامه‌ی خود را به زبان C نوشته و توسط GPUها اجرا کنند. برای توضیحات بیشتر در مورد این سکوی می‌توانید به [۲۳] مراجعه



شکل ۶- مقایسه‌ی توانایی انجام عملیات ممیز شناور در مدل‌های مختلف CPU و GPU [۲۳]



شکل ۷- مقایسه ساختار داخلی CPU و GPU [۲۳]

۶- موازی‌سازی سیستم پیشنهادی

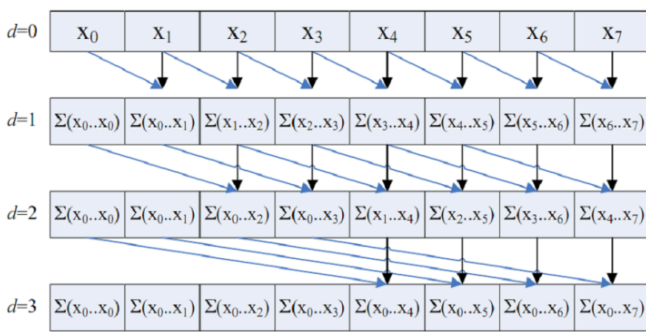
جدول ۱ بخش‌های تشکیل دهنده سیستم شناسایی چهره پیشنهادی و زمان اجرای هر یک را برای تصویری با ابعاد 325×205 نمایش می‌دهد. مشخصات سخت‌افزاری رایانه‌ی مورد استفاده به قرار زیر بوده است:

نگرانی برطرف می‌شود. شکل ۱۱ اجرای موازی این الگوریتم را برای آرایه‌ای به طول ۸ نمایش داده است.

```

for d = 1 to log n
  for all k in parallel
    if k ≥ 2k then
      x[out][k] = x[in][k - 2d-1] + x[in][k];
    else
      x[out][k] = x[in][k];
  swap(in, out)
    
```

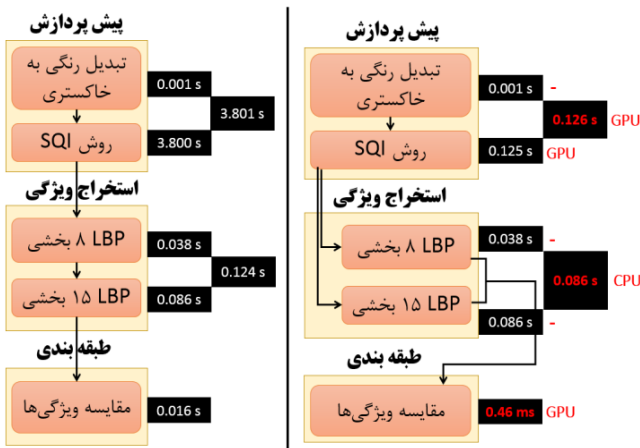
شکل ۱۰- شبه‌کد الگوریتم موازی دوباره برای محاسبه مجموع عناصر بردار



شکل ۱۱- الگوریتم دوباره برای محاسبه مجموع عناصر آرایه‌ای به طول ۸

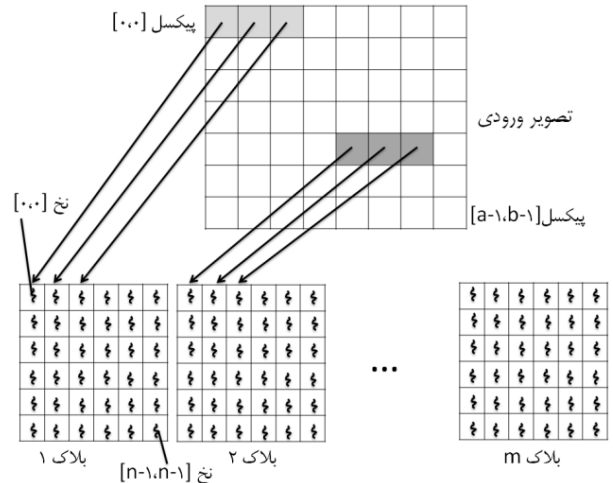
جدول ۲- زمان اجرای بخش‌های تشکیل دهنده سیستم موازی شده

مرحله	روش	مدت زمان (ms)
۱-۱	پیش‌پردازش (تبدیل رنگی به خاکستری)	1
۲-۱	پیش‌پردازش (روش SQI)	125
۱	پیش‌پردازش (کل)	126
۱-۲	استخراج ویژگی‌ها (۸ بخشی)	38
۲-۲	استخراج ویژگی‌ها (۱۵ بخشی)	86
۲	استخراج ویژگی (کل)	86
۳	مقایسه ویژگی‌ها	0.462
مجموع	کل سیستم	213.462



شکل ۱۲- خلاصه‌ای از مراحل موازی‌سازی صورت گرفته بر روی بخش‌های مختلف سیستم پیشنهادی

الگوریتم SQI برای محاسبه تصویر خارج قسمت، فیلتر گوسین را به تصویر اعمال می‌کند. برای این منظور، پنجره‌ای با اندازه‌ی 5×5 را بر روی تک‌تک پیکسل‌های تصویر با ابعاد $n \times n$ قرار می‌دهد. در نتیجه، پیچیدگی زمانی آن برابر با $O(s^2 n^2) \in O(n^2)$ خواهد بود. برای موازی‌سازی این الگوریتم، هر نخ را مسئول اعمال پنجره به یک پیکسل قرار می‌دهیم؛ در نتیجه پیچیدگی زمانی به $O(s^2) \in O(1)$ کاهش پیدا می‌کند. برای سپردن هر پیکسل به یک نخ، باید ابتدا نخ‌های تمامی بلوک‌های GPU و تمامی پیکسل‌های تصویر ورودی را به صورت خطی کنار هم قرار داده و سپس یک نگاشت ساده یک به یک بین آن‌ها برقرار کنیم؛ که البته این امر نیاز به تغییر چیدمان به صورت واقعی ندارد. شکل ۹ یک شمای کلی از توضیحات داده شده را نشان می‌دهد.



شکل ۹- نحوه‌ی توزیع کار بین نخ‌ها و بلوک‌های GPU

عملگر LBP به دو شکل ۸ بخشی و ۱۵ بخشی برای استخراج ویژگی به کار گرفته شد. از آنجا که این دو عملگر کاملاً نسبت به هم مستقل هستند، با استفاده از CPU آن‌ها به صورت موازی نسبت به هم پیاده‌سازی شد. در نتیجه زمان اجرای بخش ۲ از جدول ۱ به زمان عملگر ۱۵ بخشی (عملگر زمانبرتر) کاهش یافت.

مقایسه‌ی بردارهای ویژگی توسط روش مربع چای صورت می‌گیرد. واضح است که پیچیدگی زمانی این روش برابر با $O(n)$ است که n طول بردار ویژگی است.

این الگوریتم از دو بخش زیر تشکیل می‌شود:

۱. محاسبه‌ی جمع، تفریق و تقسیم بردارهای ویژگی
۲. جمع نتایج حاصل

در بخش اول، خانه‌های معادل هم از دو بردار ویژگی به صورت ترتیبی مورد پردازش قرار می‌گیرند؛ در صورتی که پردازش هر دو خانه‌ی معادل را به یک نخ در GPU بسپاریم، پیچیدگی زمانی را از $O(n)$ به $O(1)$ کاهش داده‌ایم. البته در این حالت، خروجی الگوریتم یک بردار خواهد بود نه یک مقدار! در نتیجه، محاسبه‌ی مجموع عناصر بردار حاصل، ضروری است. برای محاسبه‌ی مجموع عناصر بردار از الگوریتم موازی دوباره^{۲۳} بهره برده‌ایم. شبه‌کد این روش که دارای پیچیدگی $O(\log n)$ می‌باشد در شکل ۱۰ نمایش داده شده است.

کودا دارای چند نوع حافظه است؛ حافظه‌ی سراسری بزرگترین و در عین حال کندترین آن‌هاست. حافظه‌ی اشتراکی دارای حجم محدود (بسته به نسخه کارت گرافیک) ولی سرعتی ۱۰۰ تا ۱۵۰ برابر حافظه‌ی سراسری است. از آنجا که دو بافر مورد استفاده در این الگوریتم به تعداد $O(n \log n)$ مرتبه خوانده می‌شوند، سرعت خواندن از آن‌ها بسیار با اهمیت است؛ با استفاده از حافظه‌ی اشتراکی، این

است؛ روش‌هایی مانند DSFQI که دقت نزدیکی به سیستم پیشنهادی کسب کرده‌اند، دارای سرعت مناسبی نیستند. در صورتی که سیستم پیشنهادی در کسری از ثانیه عمل می‌کند.



شکل ۱۴- چند نمونه از تصاویر CMU-PIE با روشنایی متفاوت

جدول ۳- مقایسه و آزمون سیستم پیشنهادی بر روی دو پایگاه داده YaleB و CMU-PIE

روش‌ها	دقت شناسایی (%)						
	CMU	YaleB					مجموع
		گروه ۱	گروه ۲	گروه ۳	گروه ۴	گروه ۵	
QI	84	100	98.3	62.5	34.2	23.6	63.72
HE	-	100	95	88.3	50.7	46.8	76.16
PCA	54	-	-	-	-	-	50
LDA	-	-	-	-	-	-	79
QIR	-	100	100	100	90	82.1	94.42
SQI	98.38	100	97.5	100	96.4	97.8	98.34
MQI	98.75	100	98.3	98.3	98.5	97.3	98.48
DMQI	-	100	100	98.3	98.5	97.8	98.92
DSFQI	-	100	99.1	99.1	98.5	98.4	99.02
روش پیشنهادی ۱	98.60	100	98.3	97.5	99.2	98.4	98.68
روش پیشنهادی ۲	99.11	100	100	97.5	100	98.9	99.28

۸- نتیجه‌گیری و کارهای آینده

در این مقاله، سیستمی برخط برای شناسایی چهره پایدار به تغییرات حالت چهره و روشنایی ارائه شد. این سیستم دارای سه بخش پیش‌پردازش، استخراج ویژگی و طبقه‌بندی بود که هر یک به تفصیل توضیح داده شدند. در بخش پیش‌پردازش، با استفاده از الگوریتم SQI تغییرات روشنایی در تصویر خنثی گردید. با استفاده از مقادیر متفاوت انحراف معیار در مرحله آموزش و آزمایش، کارایی سیستم افزایش پیدا کرد. سپس نسخه‌ی توسعه یافته‌ی عملگر LBP به صورت چندبخشی و در تعداد بخش‌های متفاوت برای استخراج ویژگی به تصویر اعمال گردید. در مرحله‌ی مقایسه، روش مربع چای وزن‌دار به کار گرفته شد؛ وزن‌های به کار گرفته شده، به صورت تجربی محاسبه شدند و همان‌طور که در نتایج مشخص است، تاثیر عمده‌ای در بهبود کارایی سیستم داشتند.

جدول ۲ زمان اجرای موازی بخش‌های تشکیل دهنده سیستم را برای یک تصویر با ابعاد 325×205 نمایش می‌دهد. افزایش سرعت حاصل از موازی‌سازی سیستم ۱۸.۵ برابر است که میزان قابل توجهی است. با استفاده از CPU به تنهایی، رسیدن به چنین سرعتی به سادگی امکان پذیر نیست. شکل ۱۲ مراحل موازی‌سازی انجام شده را خلاصه‌وار نمایش می‌دهد.

۷- نتیجه آزمایشات

برای آزمون سیستم پیشنهادی از دو پایگاه داده‌ی شناخته شده‌ی YaleB [۲۴] و CMU-PIE [۲۵] استفاده شده است. روش‌های QI، HE، SQI، QIR، MQI، DSFQI، DMQI، PCA و LDA با روش پیشنهادی، مورد مقایسه قرار گرفته‌اند. پایگاه داده YaleB شامل 10 شخص است که از هر یک ۶۴ تصویر با روشنایی متفاوت موجود می‌باشد. تصاویر به صورت دستی بریده شده و به اندازه‌ی 168×192 در آورده شده‌اند. مجموعه تصاویر بر اساس زاویه‌ی بین منبع نور و دوربین که منجر به سطح روشنایی متفاوتی گشته، به پنج دسته تقسیم شده‌اند (شکل ۱۳). در آزمایشات، یک تصویر از هر شخص که دارای روشنایی معمولی است، به عنوان تصویر آموزشی در نظر گرفته شده و سایر تصاویر مورد آزمون قرار گرفته‌اند.



شکل ۱۳- نمونه تصاویری از پنج گروه با سطح روشنایی متفاوت از YaleB

پایگاه داده CMU-PIE شامل ۶۴ شخص است که از هر شخص تصاویری با تغییرات در ژست، حالت و روشنایی وجود دارد. از آنجا که تمرکز سیستم پیشنهادی بر روی تغییرات روشنایی بوده است، تنها ۲۱ تصویر متعلق به گروه C27 برای هر شخص انتخاب شده‌اند. تصاویر به صورت دستی بریده شده و به اندازه‌ی 150×150 در آورده شده‌اند (شکل ۱۴). در آزمایشات، یک تصویر از هر شخص با روشنایی تقریباً معمولی (f8) به عنوان تصویر آموزشی در نظر گرفته شده و سایر تصاویر مورد آزمون قرار گرفتند. جدول ۳ نتایج آزمایشات صورت گرفته بر روی سیستم پیشنهادی را نمایش می‌دهد.

روش پیشنهادی ۱ در جدول ۳، نسخه‌ای از سیستم پیشنهادی است که از مقایسه‌ی وزن‌دار بهره نمی‌برد. با مقایسه روش پیشنهادی ۱ و ۲ می‌توان به کارایی مقایسه وزن‌دار به خوبی پی برد.

جدول ۳ به روشنی نشان می‌دهد که روش پیشنهادی از سایر روش‌های مقایسه شده عملکرد بهتری داشته است. گروه ۵ در پایگاه YaleB به دلیل تغییرات روشنایی بالای آن، دشوارترین گروه محسوب می‌شود. علت اصلی برتری روش پیشنهادی، اختلاف دقت کسب شده توسط آن بر روی همین گروه بوده است. نتیجه‌ی دیگری که از نتیجه‌ی آزمایشات می‌توان گرفت، کارایی بهتر روش‌های مبتنی بر تصویر خارج قسمت نسبت به سایر روش‌ها از قبیل LDA و PCA است. نکته‌ی مهمی که در این جدول قابل ملاحظه نیست، فاکتور سرعت

[11] X. Chai, S. Shan, X. Chen, and W. Gao, "Locally Linear Regression for Pose-invariant Face Recognition," *IEEE Trans. Image Processing*, vol. 16, no. 4, pp. 1716-1725, 2007.

[12] M. Wai Lee, and S. Ranganath, "Pose-invariant Face Recognition Using a 3D Deformable Model," *Journal of Pattern Recognition*, vol. 36, no. 6, pp. 1835-1846, 2003.

[13] T. Ojala, M. Pietikäinen, and D. Harwood, "A Comparative Study of Texture Measures with Classification based on Featured Distributions," *Journal of Pattern Recognition*, vol. 29, no. 2, pp. 51-59, 1996.

[14] L. Wiskott, J. M. Fellous, N. Kuiger, and C. von der Malsburg, "Face Recognition by Elastic Bunch Graph Matching," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 1, pp. 775-779, 1997.

[15] X. Zhang, and Y. Gao, "Face Recognition Across Pose: A Review," *Journal of Pattern Recognition*, vol. 42, no. 2, pp. 2876-2896, 2009.

[16] B. Amberg, R. Knothe, and T. Vetter, "Expression Invariant 3D Face Recognition with a Morphable Model," *Proc. IEEE Intl Conf. Automatic Face and Gesture Recognition*, pp. 1-6, 2008.

[17] A. Bronstein, M. Bronstein, and R. Kimmel, "Robust Expression-invariant Face Recognition from Partially Missing Data," *Proc. IEEE Intl Conf. Computer Vision*, pp. 396-408, 2006.

[18] H. S. Lee, and D. Kim, "Expression-invariant Face Recognition by Facial Expression Transformations," *Journal of Pattern Recognition*, vol. 29, no. 2, pp. 1797-1805, 2008.

[19] S. Shan, W. Gao, B. Cao, and D. Zhao, "Illumination Normalization for Robust Face Recognition against Varying Lighting Conditions," *Proc. IEEE Intl Workshop on Analysis and Modeling of Faces and Gestures*, pp. 157-164, 2003.

[20] T. Ojala, M. Pietikäinen, and T. Maenpää, "Multiresolution Gray-scale and Rotation Invariant Texture Classification with Local Binary Patterns," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 971-987, 2002.

[21] M. Biglari, F. Mirzaei, and H. Ebrahimpour, "Illumination Invariant Face Recognition Using SQI and Weighted LBP Histogram," *Journal of Electrical Engineering*, vol. 7, no. 4, pp. 47-54, 2013.

[22] J. D. Owens, M. Houston, D. Luebke, S. Green, J. E. Stone, and J. C. Phillips, "GPU Computing," *Journal of Electrical Engineering*, vol. 96, no. 6, pp. 879-899, 2008.

[23] CUDA C Programming Guide V5.0, <http://docs.nvidia.com/cuda/cuda-c-programming-guide/>, October 2012.

[24] A. S. Georghiadis, P. N. Belhumeur, and D. J. Kriegman, "From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose,"

در پایان، سیستم نهایی با استفاده از CPU و GPU موازی‌سازی شد؛ به شکلی که سرعت آن تقریباً تا ۱۹ برابر افزایش پیدا کرد. زمان نهایی نسخه‌ی موازی سیستم شناسایی چهره برای پردازش یک تصویر با ابعاد 325×205 به ۲۱۳ میلی ثانیه کاهش پیدا کرد. این زمان برای استفاده از سیستم در یک کاربرد برخط، کاملاً ایده‌آل است. با استفاده از چنین زمان پردازشی می‌توان سیستم را برای پردازش تصاویر متحرک نیز به کار گرفت.

دقت بدست آمده از سیستم و نتایج مقایسه‌ی آن با سایر روش‌ها، قدرت و پایداری روش پیشنهادی را نشان می‌دهد. تکمیل این روش به شکلی که نسبت به فاکتورهای چالش برانگیزی چون تغییرات ژست و انسداد بخشی از چهره نیز پایدار شود، می‌تواند ایده‌ی مناسبی برای کارهای آینده باشد.

مراجع

[1] M. Turk, and A. Pentland, "Eigenfaces for Recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 2, pp. 71-86, 1991.

[2] J. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "Face Recognition Using LDA-based Algorithms," *IEEE Trans. Neural Networks*, vol. 14, no. 10, pp. 195-200, 2003.

[3] M. S. Bartlett, J. R. Movellan, and T. J. Sejnowski, "Face Recognition by Independent Component Analysis," *IEEE Trans. Neural Networks*, vol. 13, no. 3, pp. 1450-1464, 2002.

[4] Y. Cheng, Z. Jin, and C. Hao, "Illumination Normalization based on Different Smoothing Filters Quotient Image," *Proc. IEEE Intl Conf. Intelligent Networks and Intelligent Systems*, pp. 28-31, 2010.

[5] M. Nishiyama, T. Kozakaya, and O. Yamaguchi, "Illumination Normalization Using Quotient Image-based Techniques," *Proc. IEEE Intl Conf. Recent Advances in Face Recognition*, pp. 97-108, 2008.

[6] A. Shashua, and T. Riklin-Raviv, "The Quotient Image: Class-based Re-rendering and Recognition with Varying Illuminations," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 129-139, 2001.

[7] H. Wang, S. Z. Li, Y. Wang, and J. Zhang, "Self Quotient Image for Face Recognition," *Proc. IEEE Intl Conf. Image Processing*, pp. 1397-1400, 2004.

[8] Y. Zhang, J. Tian, X. He, and X. Yang, "MQI-based Face Recognition under Uneven Illumination," *Proc. IEEE Intl Conf. Advances in Biometrics*, pp. 290-298, 2007.

[9] L. Zhichao, and E. M. Joo, "Face Recognition under Varying Illumination," *Proc. IEEE Intl Conf. New Trends in Technologies: Control, Management, Computational Intelligence and Network Systems*, pp. 470-476, 2010.

[10] X. G. He, J. Tian, L. F. Wu, Y. Y. Zhang, and X. Yang, "Illumination Normalization with Morphological Quotient Image," *Journal of Software*, vol. 18, no. 1, pp. 2318-2325, 2007.

اطلاعات بررسی مقاله:

تاریخ ارسال: ۹۳/۲/۲۹

تاریخ اصلاح: ۹۳/۸/۴

تاریخ قبول شدن: ۹۳/۸/۱۱

نویسنده مرتبط: محسن بیگلری، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شاهرود، شاهرود، ایران.

IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 23, no. 3, pp. 643-660, 2001.[25] T. Sim, S. Baker, and M. Bsat, "The CMU Pose, Illumination, and Expression (PIE) Database," *Proc. IEEE Intl Conf. Automatic Face and Gesture Recognition*, pp. 46-51, 2002.

محسن بیگلری مدرک کارشناسی خود را در گرایش نرم‌افزار از دانشگاه کاشان در سال ۸۹ دریافت کرد. سپس در سال ۹۱ موفق به اخذ مدرک کارشناسی ارشد در گرایش نرم‌افزار از دانشگاه کاشان گردید. در حال حاضر نیز، دانشجوی دکتری دانشگاه صنعتی شاهرود در گرایش هوش مصنوعی است. موضوع پایان‌نامه کارشناسی ارشد او، "تشخیص و شناسایی چهره با استفاده از الگوریتم‌های موازی، توسط کودا" می‌باشد. علایق پژوهشی او برنامه‌نویسی موازی و پردازش تصویر می‌باشند. آدرس پست‌الکترونیکی ایشان عبارت است از:

biglari@shahroodut.ac.ir



فائزه میرزائی مدرک کارشناسی و کارشناسی ارشد خود را در رشته‌ی مهندسی کامپیوتر در گرایش نرم‌افزار از دانشگاه کاشان در سال‌های ۱۳۸۹ و ۱۳۹۱ دریافت کرد. وی هم‌اکنون دانشجوی دکتری هوش مصنوعی در دانشگاه شاهرود می‌باشد. زمینه تحقیقات ایشان پردازش تصویر، بیومتریک و پردازش موازی است. آدرس پست‌الکترونیکی ایشان عبارت است از:

fmirzaei@shahroodut.ac.ir



حسین ابراهیم‌پور کومله دوره کارشناسی خود را در رشته مهندسی کامپیوتر (سخت‌افزار) با رتبه اول در دانشگاه صنعتی اصفهان در سال ۱۳۷۲ و دوره کارشناسی ارشد را با رتبه ممتاز در دانشگاه صنعتی امیرکبیر در گرایش هوش ماشین و رباتیک در سال ۱۳۷۵ به پایان رساند. سپس به عنوان عضو هیات علمی جذب دانشگاه کاشان شد و پس از قبولی در امتحان دکتری بورس و اعزام، از سال ۱۳۷۹ عازم کشور استرالیا گشت. او پروژه دکتری خود را در سال ۲۰۰۴ با عنوان "تکنیک‌های فرکتالی برای شناسایی چهره" در دانشگاه صنعتی کوئینزلند با موفقیت به پایان رساند. دکتر ابراهیم‌پور پس از اتمام دوره پسادکتری در دانشگاه نیوکاسل استرالیا و همکاری با مرکز تحقیقات علمی استرالیا (csiro) به عنوان "Visiting Scientist" در سال ۱۳۸۶ به ایران مراجعت و همکاری خود را با دانشگاه‌های کشور از سر گرفته است. در حال حاضر ایشان عضو هیات علمی و معاون پژوهشی دانشکده مهندسی برق و کامپیوتر دانشگاه کاشان، همچنین عضو هیات تحریریه و مدیر داخلی مجله محاسبات نرم و عضو کمیته علمی و داور چندین ژورنال و کنفرانس معتبر داخلی و خارجی هستند و از ایشان تا کنون بیش از ۵۰ مقاله علمی در کنفرانس‌ها، مجلات و کتب معتبر علمی چاپ گشته است. آدرس پست‌الکترونیکی ایشان عبارت است از:

ebrahimpour@kashanu.ac.ir

- ¹ Expression
- ² Illumination
- ³ Pose
- ⁴ Occlusion
- ⁵ Self Quotient Image
- ⁶ Local Approaches
- ⁷ Holistic Approaches
- ⁸ Local Binary Pattern
- ⁹ Graphics Processing Unit (GPU)
- ¹⁰ Ojala
- ¹¹ Texture
- ¹² Monotonic Gray-scale
- ¹³ Uniform
- ¹⁴ Smooth
- ¹⁵ Convolution
- ¹⁶ Morphological Operators
- ¹⁷ Classification
- ¹⁸ Chi Square
- ¹⁹ Cache
- ²⁰ Flow Control
- ²¹ CUDA
- ²² Thread
- ²³ Double Buffer
- ²⁴ Histogram Equalization