تشخیص کالاهای فروشگاهی به کمک شبکه عصبی عمیق و پیاده سازی آن بر روی پردازنده‌های گرافیکی

مقدمه

فروشگاه‌های سنتی نیاز به یک صندوق‌دار برای تشخیص کالاهای خریداری شده‌ی مشتریان دارند. این صندوق‌دار معمولا کد مربوط به محصولات را به صورت دستی یا با استفاده از دستگاه بارکدخوان وارد می‌کند. سیستم (در صورت وجود) جمع قیمت کالاها را محاسبه کرده و رسید را برای مشتری چاپ می‌کند.

در این گونه فروشگاه‌ها معمولا شاهد صف‌های طولانی مشتریان منتظر نوبت برای تکمیل خرید هستیم که نارضایتی به‌دنبال دارد. پس به منظور سرویس‌دهی بهتر به مشتریان، فروشگاه‌های بزرگ مجبور به به‌کارگیری شمار بسیاری نیروی صندوق‌دار هستند که هزینه‌های جاری فروشگاه‌ها (چه مالی و چه انسانی) را بالا می‌برد.

در سال‌های اخیر تغییر بزرگی در صنعت خرده‌فروشی مخصوصا فروشگاه‌ها به‌سوی سیستم‌های خودوارس(Self-checkout) رخ داده است. در این گونه سیستم‌ها که خود مشتری وظیفه‌ی استفاده از دستگاه بارکدخوان یا وزن کردن میوه‌ها و سبزیجات را دارد، خرده‌فروش‌ها می‌توانند نیروها را در خدمات دیگر مانند جایگزینی کالاهای برداشته شده، تمیز کردن مغازه و بسته‌بندی کردن کالاها و ... به‌کار گیرند. این روش افزون‌بر کم‌کردن هزینه‌ی نیروی انسانی، نوعی احترام به مشتری و افزایش رضایت برخی از مشتریان را به‌دنبال دارد. یک نمونه از احترام به حریم خصوصی مشتری این است که مشتری از خرید مقدار زیادی محصول بهداشتی خجالت‌زده نمی‌شود و هم‌چنین احساس کنترل در مشتری ایجاد می‌شود و این دغدغه را ندارد که نکند صندوق‌دار محصولی را اضافه یا کم وارد کرده است.

در استفاده از روش سیستم خودوارس، فروشگاه می‌تواند دو رویکرد را در پیش بگیرد. یا شماری دستگاه برای استفاده‌ی مشتریان در فروشگاه قرار داده شود یا هر سبد خرید مجهز به یک دستگاه بارکدخوان باشد.

در رویکرد اول به‌دلیل سرعت عمل پایین‌تر مشتریان نسبت به صندوق‌دار ممکن است مسئله‌ی صف طولانی حل نشود. روش دوم نیز هزینه‌ی زیادی برای تهیه و نگهداری شمار زیادی دستگاه بارکدخوان به دنبال دارد.

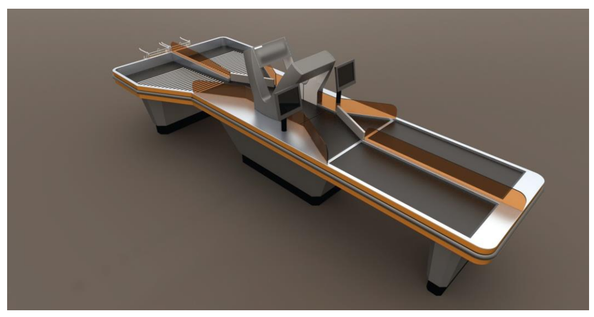
این روش معمولا مشکلات دیگری را از جمله چالش درستی‌سنجی کالاها، شناسایی، انتظار برای رسیدگی به مشکلات احتمالی و یادگیری مشتریان برای کار درست با سیستم نیز دارد. روش‌های درستی‌سنجی کالاهای وارد شده معمولا از طریق محاسبه‌ی وزن محصولات برداشته‌ی شده‌ی مشتری صورت می‌گیرد که افزون‌بر این که درجه‌ی اطمینان روش سنتی را ندارد، به خاطر خطاهای احتمالی نارضایتی‌هایی را برای مشتری ایجاد می‌کند. برای مثال بسیار رخ می‌دهد که مشتری با خطای «یک کالای نامشخص در سبد خرید شما وجود دارد» یا «یک محصول از سبد خرید شما برداشته شد، لطفا آن را در جای خود قرار دهید» روبه‌رو می‌شود. برخی زمان‌ها نیز یک محصول توسط دستگاه به نادرستی شناسایی می‌شود که نیاز به کمک نیروهای پشتیبانی فروشگاه دارد. نبود نیروی دائم مانند صندوق‌دار در کنار دستگاه موجب می‌شود تا مشتریانی که در کار با سیستم به مشکل خورده‌اند برای دریافت پشتیبانی منتظر شوند و به‌دنبال نیروی پشتیبان بگردند که ناگزیر نارضایتی آن‌ها در پی دارد. کاربرد هوش مصنوعی در این سیستم‌ها می‌تواند با تشخیص دقیق‌تر و رخداد خطای کم‌تر نیاز کاربر به نیروی پشتیبان را کم‌تر کند. اما بنا بر چالش‌های پیش‌روی یک سیستم خودوارس، برخی فروشگاه‌ها به سوی راه‌حل‌های وارسی‌تمام‌خودکار(Automatic Checkout) یا بی‌وارسی (No Checkout) رفته‌اند. در راه‌حل‌های وارسی تمام‌خودکار کاربر نیازی به درگیر شدن در فرآیند وارسی ندارد و تنها بایستی محصولات مورد نیاز خود را بردارد و به روی ریل وارسی قرار دهد. در راه‌حل‌های بی‌وارسی مشتری حتی نیاز به قراردادن کالاهای مورد نظرش به روی ریل وارسی هم ندارد و تنها با برداشتن آن‌ها در رسید خرید او نوشته می‌شود.

نخستین سیستم‌های خودوارس (خودِ مشتری) در سال ۱۹۹۰ طراحی و ساخته‌شدند اما در سال ۲۰۰۰ مورد اقبال قرار گرفتند . در این گونه سیستم‌ها معمولا یک وزن‌کننده‌یِ وارسِ نهایی، یک بارکدخوان لیزری و یک جایگاه کارت‌خوان برای تسویه‌ی حساب وجود داشت.

به مرور زمان سیستم‌هایی برای رفع قابلیت اطمینان کم این سیستم‌ها طراحی و ساخته شدند. خدماتی از جمله تشخیص خودداری مشتری از اسکن کالا ، تشخیص کالای اضافه در سبد خرید و تشخیص خطای اجرایی به ساختارهای موجود خودوارس ارائه می‌دادند. برای نمونه محصول ScanItAll Checkout Vision Suite شرکت stoplift از نمونه‌های سرویس‌دهنده چنین خدماتی است. این سیستم‌های مکمل از تخلفات مشتری در خرید پیشگیری می‌کنند ولی هنوز نیاز به تعامل زیاد مشتری دارند و زمان‌بر هستند. از این رو رغبت بازار در جهت سیستم‌های سریع بی‌وارسی و وارسی تمام‌خودکار است.

برای طراحی سیستمِ وارسی تمام‌خودکارِ محصولات فروشگاه، بنابر جستجوهای انجام شده در محصولات مشابهِ به بهره‌برداری رسیده، چند روش گوناگون به‌کار گرفته می‌شود.

۱- تصویربرداری از تمامِ سطح‌هایِ کالا و پیداکردن بارکد روی جسم و تشخیصِ آن.



۲- کاربرد سنسورهای مختلف و الگوریتم‌های دسته‌بندیِ گوناگون کالا برای بازشناخت از ویژگی‌های تصویری و غیرتصوری (مانند وزن). مانندِ محصول EasyFlow از شرکت ITAB

۳- استفاده از شبکه‌های عصبی عمیق برای بازشناختِ کالا

۴- استفاده از برچسب‌های RFIDیِ ارزان‌قیمت

هدف در این جستار، جست‌وجو و امکان‌سنجی رویکردهای ممکن و بررسی چالش‌های پیش روی هر کدام از روش‌ها برای پیاده‌سازی صنعتی و پیاده‌سازی بخش‌هایی از این سیستم است.

چالش‌هایِ پیشرویِ رویکردهای موجود

برای طراحی یک سیستمِ خودکارِ تشخیصِ محصول برپایه‌ی بارکد، بایستی جایگاه و زوایای دوربین‌های نوری و آیینه‌ها (در صورت وجود) محاسبه و مشخص شود، زاویه‌ی هر دوربین با سطح عکس‌گرفته شده باید با تبدیل‌های هندسی خنثی شود و با الگوریتم‌های تشخیص جایگاه بارکد، کاهش noise و تشخیص بارکد در زوایای مختلف، درصد موفقیت در تشخیص کالا بالا برود. در این جستار تلاش شده راه‌حل‌هایی برای هر یک از این چالش‌ها ارائه شود.

برای طراحی یک سیستم غیرمبتنی بر بارکد و برپایه‌ی ویژگی‌های تصویری و غیرتصویری، نیاز به شناسایی ویژگی‌های مناسب برای استخراج از تصویر ، طراحی و پیاده‌سازی الگوریتم‌های مناسب برای برگیری این ویژگی‌ها و طراحی دسته‌بند های مناسب و ترکیب آن‌ها می‌باشد. به نظر می‌آید طیف رنگی، شکل سه‌بعدی و حجم از ویژگی‌هایی باشند که برگیری آن‌ها مفید باشد. همچنین وزن جسم می‌تواند به دسته‌بندی بهتر محصولات کمک کند.

طراحی چنین سیستمی بر پایه‌ی شبکه‌های عصبی عمیق نیز با چالش‌هایی روبه‌روست. طراحی شبکه‌ی مناسب، جمع‌آوری داده‌های کافی و مناسب چالش بزرگی‌ست که در راه به‌کارگیری این شبکه‌ها وجود دارد.

استفاده از RFIDدر حالِ حاضر در کشور ایران به علت سربار مالی آن برای محصولات کم‌قیمت که همراه با عقب‌ماندگی کارخانه‌ها در ایران از فن‌آوری روز، موجب عدم استقبال کارخانه‌ها در به‌کارگیری آن‌ها شده، مقدور نیست.

برتری‌سنجیِ نخستینِ رویکردها

از برتری‌های به‌کارگیری الگوریتم‌های غیرمبتنی بر بارکد در برابر الگوریتم‌های برپایه‌ی بارکد، بازشناخت میوه‌ها، سبزیجات و کالاهای بی‌بارکد است بدون نیاز به ساختن بارکد برای آن‌ها به‌وسیله‌ی ترازوی جدا. همچین امکانِ مخدوشی بارکد یا فریب‌کاری ضریب اطمینان سیستم‌های برپایه‌ی بارکد را پایین می‌آورند.

گزینشِ دستیِ ویژگی‌ها برای بازشناختِ کالاها، ممکن است اطلاعات مفیدی از تصویر را از دیدِ دسته‌بندها پنهان کند و بازدهی تشخیص درست سیستم را پایین بیاورد این در حالی است که اگر این گزینش همراه داده‌های بسیار به رایانه سپرده شود، احتمالا نتیجه‌ی بهتری حاصل می‌شود.

افزودن یک کالایِ نو (از پیش دیده نشده) به لیستِ کالاهایِ فروشگاه، می‌تواند چالشی برای به‌روزرسانیِ سیستمی که برپایه‌ی شبکه‌ی عصبی عمیق است ایجاد کند. بایستی داده‌های کافی از کالای نوافزوده گردآوری شود و دوباره بازتنظیم شود.

printed by MegaPrint Inc. www.postersession.com

امیرمسعود زارع بیدکی و پویا رضایی جعفری
دکتر سعید صفری

دانشکده مهندسی برق و کامپیوتر، دانشگاه تهران

روش پیشنهادی

با توجه به این که هرکدام از روش‌ها بر روی بخش خاصی از داده‌ها خوب عمل نمی‌کنند. ترکیبِ روش‌های گوناگون احتمالا می‌تواند درصد تشخیص سیستم را بالاتر ببرد. برای نمونه ممکن است یک کالا یا بارکد نداشته باشد یا بارکدِ آن مخدوش باشد، یا ویژگی‌های برگزیده برای استخراج بین دو کالا بسیار شبیه باشند (مانند دوگونه پفک هم‌اندازه و هم‌حجم با رنگِ بسته‌بندی شبیه به هم). همچنین مشکل آموزش شبکه‌ی عصبی عمیق می‌تواند با سنجش خروجیِ آن با پاسخِ دو الگوریتم دیگر حل شود و به مرور آن دو روش داده‌های آموزشی جدیدی با رویکرد یادگیری عمیق تقویتی برای شبکه‌ی عصبی عمیق مهیا کنند.

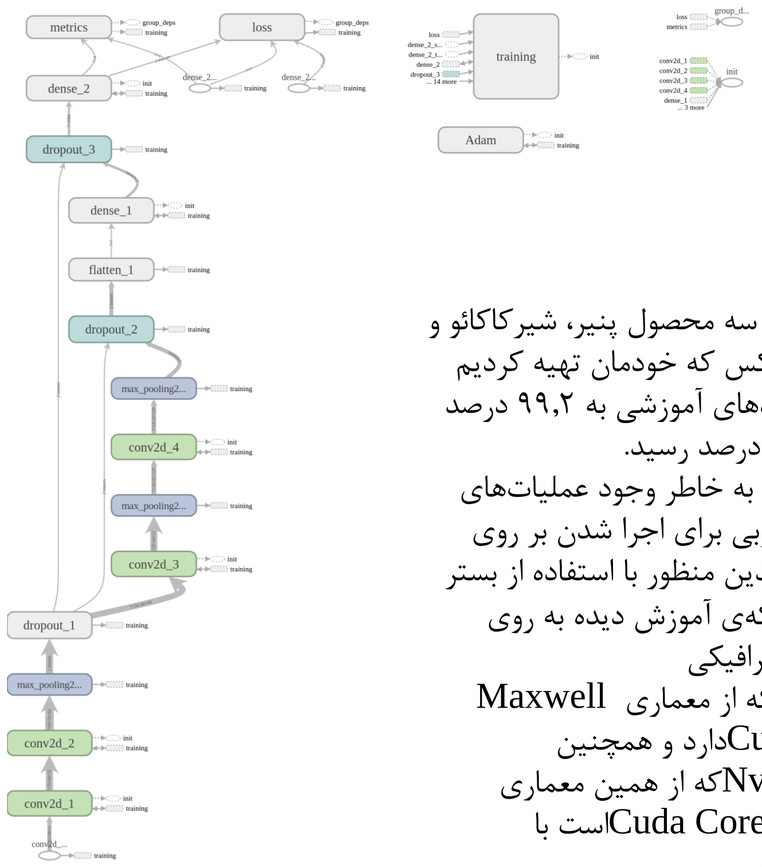
شبکه‌ی عصبی

شبکه‌های عصبی که دارای تعداد کمی لایه‌ی مخفی هستند، توانایی تشخیص ویژگی‌های پیچیده‌تر را ندارند. برای حل این مشکل از شبکه‌های عصبی با تعداد لایه‌های بیشتر استفاده می‌شود. اما با افزایش تعداد لایه‌ها به علت زیاد شدن تعداد پارامترهای مدل، مسئله‌ی فرابرازش پیش می‌آید. در راستای حل این مشکل در مسائل بینایی ماشین و پردازش تصویر، با الهام گرفتن از سیستمِ بینایی انسان، از شبکه‌های کانولوشنی بهره‌می‌گیریم.
ایده‌ی اصلی این شبکه‌ها این است که با اشتراک وزن‌ها شماری فیلتر ساخته‌شود و این فیلترها با پشت‌سر هم قرار گرفتن ویژگی‌های به مرور پیچیده‌تری از تصویر را بدست بیاورند. در نهایت این ویژگی‌های بدست آمده به یک دسته‌بند داده می‌شوند که این‌جا یک MLP سه لایه است که لایه‌ی آخر آن یک لایه‌ی Softmax است. ساختار کلی شبکه به شکل زیر طراحی شد:

نخست تصویرهای HD به تصویرهای کوچک‌شده‌ی ۹۶×۷۲ تبدیل می‌شوند. سپس در لایه‌ی اول ۳۲ فیلتر ۳ ×۳×۳ (چون هر پیکسل سه رنگ RGB دارد) می‌گذاریم. در لایه‌ی بعدی که اندازه‌ی ورودی ۹۴×۷۰×۳۲ دارد شمار ۶۴ فیلتر ۳×۳×۳۲ قرار می‌دهیم. لایه‌ی بعدی یک لایه‌ی MaxPooling است که کاربرد آن کاهش ابعاد ورودی به لایه‌های بعدی و اضافه کردن Translational Invarience به مدل است. این لایه همچنین دارای Dropout با ضریب ۱/۴ است یعنی در هنگام آموزش هر نورون به احتمال ۴/۱ خاموش می‌شوند. اینکار موجب جلوگیری از فرابرازش می‌شود که با توجه به کم بودن داده‌های آموزشی مهم است و همچنین سبب می‌گردد ویژگی‌هایی که شبکه استخراج می‌کند بهم متکی نباشند. لایه‌ی بعدی شامل ۱۲۸ فیلتر می‌باشد پس از آن دوباره یک لایه‌ی Maxpooling دیگر و دوباره یک لایه‌ی کانولوشنی با ۱۲۸ فیلتر و دوباره Maxpooling به همراه Dropout با ضریب ۴/۱ می‌آیند.

سپس با اعمال لایه‌ی Flatten تانسور را تبدیل به یک بردار ویژگی تبدیل می‌کنیم. حال یک MLP داریم که لایه‌ی پنهان آن ۱۰۲۴ نورون دارد همراه با Dropout با ضریب ۱/۲. و لایه‌ی آخر یک لایه‌ی Softmax است.

نورون‌های استفاده شده در شبکه به جز لایه‌ی اخر ReLU هستند چرا که در شبکه‌های عصبی عمیق استفاده از تابع sigmoid (هلالی) باعث ایجاد مسئله Exploding/Vanishing Gradient می‌شود. تابع Rectified Linear از نظر محاسباتی سبک‌تر است. شمای کلی شبکه به طور خلاصه‌تر در شکل زیر آمده است.



به طور آزمایشی این شبکه را با سه محصول پنیر، شیرکاکائو و نان سوخاری با حدود ۷۰۰ عکس که خودمان تهیه کردیم آموزش دادیم. نرخ دقت در داده‌های آموزشی به ۹۹,۲ درصد و در داده‌های ارزیابی به ۹۷,۵ درصد رسید.
آموزش شبکه‌های عصبی عمیق به خاطر وجود عملیات‌های یکسان بسیار موازی کاندید خوبی برای اجرا شدن بر روی پردازنده‌های گرافیکی هستند بدین منظور با استفاده از بستر tensorflow و cudNN شبکه‌ی آموزش دیده به روی CPU، بر روی دو پردازنده‌ی گرافیکی Nvidia Geforce 840M که از معماری Maxwell بهره می‌برد و ۳۸۴ Cuda Core دارد و همچنین Nvidia Geforce 960 GTX که از همین معماری برخوردار است و دارای ۱۰۲۴ Cuda Core است با داده‌های مشابه آموزش داده شد.

تشخیص بارکد

برای یافتن بارکد در هر جهت نیاز به یک تونل تصویر بردار همه جهته داریم. بدین منظور با مطالعه‌ی patentهای محصولات مشابه ساختار تونل تصویربردار با ۱۲ جهت تصویربرداری طراحی شد.
میتوان از آینه ها نیز برای کم کردن دوربین و بالابردن دقت استفاده کرد.

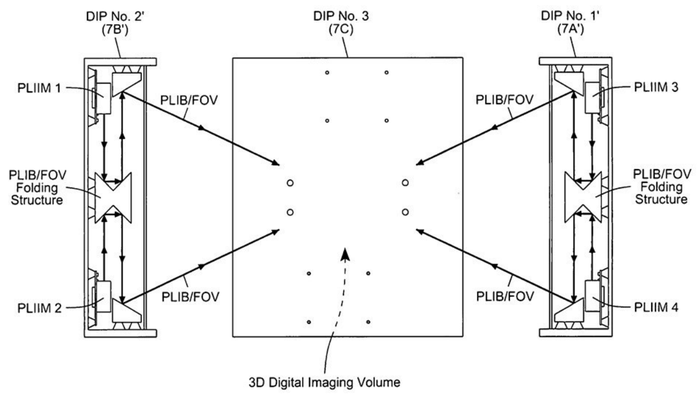


FIG. 8B

ساختار تصویربرداری و کالاهای سه بعدی در نرم افزارهای شبیه‌ساز سه بعدی، شبیه سازی شدند و دقت ابزارهای متن باز بارکدخوانی موجود روی آن ها بررسی شد. دقت ناکافی ابزارهای موجود به دو دلیل ۱- اثر دید تصویر ۲- کم کیفیتی و noiseشناسایی شد.

- رفع کم‌کیفیتی و اثر noise از کتاب‌خانه zxing

معمولا اگر کیفیت تصویر پایین باشد (مثلا در صورت Focus نادرست و ناواضح شدن تصویر) کتاب‌خانه‌ی zxing نمی‌تواند بارکد را به درستی تشخیص دهد. بدین منظور با جستجو برای رفع این مشکل از شبکه‌ی عصبی برای رفع noiseو ترمیم تصویر می‌توان استفاده کرد.[1]
نتیجه‌ی بدست آمده به وسیله‌ی این شبکه همین‌طور که در جدول زیر آمده است می‌تواند دقت را در حالتی که Focus خودکار نداریم بسیار بالاتر ببرد. هرچند با این کار زمان اجرا بیشتر می‌شود.

Library	Parameter	Precision	Recall	Time (ms) Min	Max	Average
ZXing		1.00	0.04	0.24	12.15	1.87
	try_harder	1.00	0.09	0.24	106.16	58.41
BaToo		0.60	0.13	9.29	19.19	10.10
JJil		/	0.00	120.41	253.26	146.89
ZXing-MOD	rowStep=1	0.99	0.70	1.34	439.36	156.59
	rowStep=5	0.99	0.64	1.34	101.41	37.52
	rowStep=40	1.00	0.47	1.34	27.18	6.62

- رفع اثر دیدِ تصویر

برای رفع اثر دید تصویر تنها دانستن زاویه‌ی دوربینی که از کالا عکس گرفته است با سطح افق کافی نیست. زیرا خود جسم سه بعدی ممکن است اثر دید داشته باشد و تصویر گرفته شده با اثر دید معمولا برای کتابخانه ی متن باز zxing قابل تشخیص نیست.
بدین منظور الگوریتم رفع دید با چند مرحله ی پردازش تصویری طراحی شد و نتیجه ی آن عملکرد بهتر این ابزار برای تشخیص بارکد بر روی تصاویر دارای دید بود.

نتایج

در بخش خواندن بارکد با الگورتیم پردازش تصویر طراحی شده و پیاده سازی آن دقت بارکدخوانی ابزارهای متن باز موجود در تصاویر دارای دید و کم کیفیت بهبود پیدا کرد.
در بخش آزمایش کاربرد شبکه ی عصبی نرخ دقت در داده‌های آموزشی به ۹۹,۲ درصد و در داده‌های ارزیابی به ۹۷,۵ درصد برای محصولات آزمایشی و داده های کم ساخته شده رسید.
با پیاده کردن شبکه بر روی پردازنده های گرافیکی که تعداد داده‌های کمی دارد استفاده از پردازنده‌ی گرافیکی با ۳۸۴ Cuda Core، به ۴,۵۷ برابر تسریع و استفاده از پردازنده‌ی قوی‌تر و جدیدتر با ۱۰۲۴ Cuda Core به ۳۰,۸ تسریع رسید.

	CPU – core i7 4500 U	GPU-Geforce 840M	GPU-Geforce 960 GTX
زمان	۳۳دقیقه و ۵۳ثانیه	۷ دقیقه و ۳۱ ثانیه	۱ دقیقه و ۶ ثانیه
تسریع	۱	۴٫۵۷	۳۰٫۸
دقت بر روی داده‌های تست آموزش	۹۹٫۲۲	۹۹٫۵۳	۹۹٫۶۹
دقت بر روی داده‌های validation	۹۷٫۵۲	۹۳٫۷۹	۹۵٫۶۵
تعداد epochچ را شده	۲۸	۲۸	۲۹

جمع بندی

در این تحقیق نتایج جستجو بر روی محصولات تجاری مشابه بیان شد و تلاش شد روش در پیش‌گرفته‌ی هر کدام بررسی شود. در ادامه تلاش شد تا اجزای مختلف چنین سیستمی طراحی و با کنار هم گذاشتن آن‌ها به سیستم قابل اطمینان‌تری برسیم. روش‌های مختلف تشخیص بارکد، بررسی شد و کاستی‌های آنان به دست آمد. سپس برای رفع آنان چند روش پیشنهاد و پیاده شد. پس از بررسی روش‌های دیگر در آخر یک شبکه‌ی عصبی عمیق طراحی و آموزش داده شد و بر روی پردازنده‌های گرافیکی نیز پیاده شد و تسریع حاصل سنجیده شد.

مراجع

1. **A. Zamberletti, I. Gallo, M. Carullo, E. Binaghi. “Neural Image Restoration For Decoding 1-D Barcodes Using Common Camera Phones”. In Computer Vision, Imaging and Computer Graphics. Theory and Applications, Springer Berlin Heidelberg, 2011.**

printed by MegaPrint Inc. www.postersession.com