پنجمین کنفرانس ملی فناوری‌های نوین در مهندسی برق و کامپیوتر
۳۰ شهریور ماه ۱۴۰۱ - اصفهان
وزارت علوم، تحقیقات و فناوری
موسسه آموزش عالی جهاد دانشگاهی استان اصفهان
معاونت پژوهش و فناوری
ISC
پایگاه استنادی علوم جهان اسلام
۰۱۲۲۰-۹۰۱۳۵

Arch

سیستم تشخیص هدف جنگنده با استفاده از واحدهای پردازش گرافیکی

علی نوروزی، دکترای کامپیوتر و استادیار، Ali_NoRuZi4732@yahoo.com [۱]

فائزه اکبری، کارشناسی ارشد کامپیوتر و مربی [۲]

فاطمه ملک پور فرد فینی، کارشناس ارشد کامپیوتر و مربی، NilufarMalekpour@gmail.com [۳]

چکیده

تشخیص و ردیابی اهداف متحرک در دنباله‌ای از تصاویر، یکی از شاخه‌های مهم در حوزه بینایی ماشین است. در گذشته کاربرد سیستم‌های ردیابی، محدود به مسائل نظامی، سامانه‌های امنیتی و نظارتی، موشک‌های هدایت شونده و تشخیص اهداف جنگنده بوده‌است. ردیابی موثر از نظر سرعت اجرای سیستم و میزان دقت در تشخیص و ردیابی اهداف، بخصوص در مسائل امنیتی و نظامی، نقش تعیین کننده‌ای دارد. در این مقاله به بررسی میزان سرعت اجرا و میزان دقت در پردازش، با استفاده از واحد پردازش مرکزی CPU و واحدهای پردازش گرافیکی GPU می‌پردازیم.

هدف اصلی این مقاله رسیدن به سیستمی سریع‌تر و دقیق‌تر در ردیابی هدف متحرک است. از این جهت، در اجرای سیستم با روش پیشنهادی، توسط GPU و با استفاده از CUDA انجام می‌شود تا از فرآیند پردازش موازی برای تصاویر بهره بگیریم. در فرآیند تشخیص شی هدف، برای تعیین مرزهای شی هدف و زمینه در فرآیند آشکارسازی لبه، از الگوریتم‌های لبه‌یابی Sobel، Robert، Prewitt، Canny، Log، Zerocross استفاده شده و علکرد هر یک از این متدها از نظر سرعت اجرا و دقت پردازش، ارزیابی شده است. در این مقاله شبیه‌سازی سیستم ردیابی و قفل روی هدف، در محیط نرم‌افزار متلب صورت گرفته و از ابزارهای دقیق آن، جهت ارزیابی میزان سرعت اجرا و میزان دقت هدف‌یابی استفاده شده است.

نتایج نشان می‌دهد سرعت اجرای سیستم با استفاده از واحدهای پردازش گرافیکی نسبت به اجرا در پردازنده اصلی در برخی متدهای آشکارساز لبه ۵۸٪ و ۷۵٪ افزایش داشته است. همچنین دقت پردازش تصاویر در متدهای لبه‌یاب با استفاده از واحدهای پردزاش گرافیکی بین ۳۰٪ تا ۹۰٪ نسبت به میزان دقت اجرا با پردازنده اصلی افزایش یافته است. روش پیشنهادی و بررسی‌های انجام شده در ارزیابی سیستم فوق، در بسیاری از سیستم‌های مبتنی بر پردازش، بخصوص در زمینه‌هایی که سرعت و دقت پردازش مولفه مهمی است، کاربردی می‌باشد.

واژه‌های کلیدی: ردیابی هدف متحرک، تشخیص هدف، پردازش گرافیکی، پردازش تصویر، آشکار سازی لبه، متلب/شبیه سازی

[۱] موسسه آموزش عالی سینا کاشان
[۲] موسسه آموزش عالی سینا کاشان
[۳] موسسه آموزش عالی سینا کاشان

www.SID.ir

۱- مقدمه

ردیابی اهداف متحرک در دنباله‌ای از تصاویر یکی از مباحث مهم و پرکاربرد در پردازش تصویر و بینایی ماشین می‌باشد. اگرچه در گذشته این موضوع محدود به کاربرد در مسائل نظامی و امنیتی بوده است، اما امروزه پردازش تصویر برای تشخیص اشیا موجود در تصاویر دیجیتال در زمینه‌های پزشکی، کنترل تردد، رباتیک، فشرده‌سازی تصاویر، بازسازی تصاویرآسیب دیده، صنعت و تولیدات کاربرد بسیاری دارد. در برخی کاربردها مانند سامانه‌های امنیتی، تشخیص خودکار جنگنده، بمب و موشک، سرعت اجرا و دقت در هدف یابی، مسئله بسیار مهم و حیاتی می‌باشد؛ چراکه اگر این دو پارامتر در چنین سیستم‌هایی دارای ضعف و کاستی باشد، سیستم مربوطه در شرایط حیاتی دچار کندی سرعت تصمیم‌گیری یا خطا در صحت هدف یابی شده و ممکن است جان هزاران انسان و یا حتی یک قلمرو از دست برود.

در زمینه پردازش تصاویر و ردیابی اشیا، روش‌های مختلفی برای تشخیص و نمایش شی هدف وجود دارد که هر یک براساس ویژگی‌هایی از تصویر عمل می‌کنند. محققان زیادی به پژوهش در رابطه با بهبود عملکرد این الگوریتم‌ها و افزایش کارایی سیستم‌های مربوطه پرداخته اند. همچنین با پیشرفت روز افزون ابزارهای مرتبط و پردازنده‌های گرافیکی با قابلیت‌های بالقوه، شاهد افزایش کارایی این سامانه‌ها خواهیم بود. در این تحقیق به ارزیابی تفاوت دقت و سرعت هدف‌یابی در واحدهای پردازش گرافیکی و واحد پردازش مرکزی می‌پردازیم. در پیاده‌سازی روش پیشنهادی از الگوریتم‌های لبه‌یابی برای تشخیص شی متحرک در تصاویر متوالی از یک رشته ویدیو، استفاده شده‌است.

روش پیشنهادی از آن نظر که موجب افزایش سرعت و دقتی چند برابر بیشتر از روش‌های پیاده‌سازی شده با پردازشگر مرکزی می‌شود، در اکثر سیستم‌های پردازش تصویر مورد استفاده است. میزان افزایش سرعت، بسته به نوع سیستم و الگوریتم‌های بکار برده شده نیز ممکن است حتی تا چندین برابر هم برسد. روش پیشنهادی موجب کاهش استفاده از تمام توان پردازشگر مرکزی، برای پردازش تصاویر شده و این فرآیند به واحد پردازش گرافیکی که توان بالاتری دارد محول می‌شود، در نتیجه کارایی سیستم‌های مربوطه بطور چشمگیری افزایش می‌یابد.

۲- بررسی کارهای مرتبط

مرجع [۱] به توسعه مدل مناسب قطعه‌بندی برای تصاویر ویدیویی و ترکیب آنها با الگوریتم‌های ردیابی پرداخته‌است. در روند سیستم، قطعه‌بندی با الگوریتم انتقال میانگین انجام شده، سپس با الگوریتم k میانگین به اصلاح نواحی کوچک و نویزی پرداخته‌است. سپس نواحی کوچک‌تر با نواحی بزرگتر همجوار خود ادغام می‌شود. الگوریتم پیشنهادی، پیچیدگی زمانی و محاسباتی کمی دارد ونسبت به شرایط مختلف پایدار است. در روش پیشنهادی، دقت ردیابی در شرایطی که رنگ هدف شبیه رنگ زمینه باشد، افزایش یافته‌است.

مرجع [۲] از الگوریتم فیلتر ذره‌ای برای تشخیص و ردیابی اهداف هوایی متحرک، استفاده کرده‌است. سپس هدف را بر مبنای چهار معیار تحلیل اندازه، بردار سرعت، مسیریابی و تصاویر حرارتی، نوع اهداف پرنده نظامی و غیر نظامی، شناسایی می‌کند. روش پیشنهادی با GPU و قابلیت CUDA و با چند ویدیو با چالش تعداد اهداف، سرعت، جهت و مسیر حرکت متفاوت، و نوع هدف اجرا شده‌است. الگوریتم بکار برده شده در هر سناریو عملکرد بسیارمطلوبی داشته‌است.

مرجع [۳] از الگوریتم انتقال میانگین، که دارای ویژگی سرعت و سادگی است، به‌عنوان الگوریتم پایه برای ردیابی استفاده کرده‌است. برای رفع مشکلات این روش، از هیستوگرام وزن دار پس زمینه اصلاح شده در شناسایی بهتر هدف استفاده شده‌است. الگوریتم انتقال میانگین اصلی، روش پیشنهادی و دو روش دیگر مبتنی بر الگوریتم انتقال میانگین اصلی، در محیط نرم افزار متلب و به صورت فریم به فریم ارزیابی شده‌است. طبق نتایج، الگوریتم پیشنهادی، به میزان قابل توجهی دقت و کیفیت سیستم را نسبت به الگوریتم ردیابی انتقال میانگین پایه، افزایش داده‌است. طبق نتایج، الگوریتم پیشنهادی، از سه الگوریتم دیگر عملکرد بالاتری دارد.

در مرجع [۴] یک الگوریتم تلفیقی برای تشخیص و ردیابی هدف پرنده انتخاب شده‌است. هدف، در فریم اول به‌صورت دستی، مشخص شده و در فریم دوم با مرکزیت ناحیه مشخص شده، یک الگو استخراج می‌شود. سپس با الگوریتم ASIFT نقاط کلیدی استخراج و تطبیق‌های بین این دو فریم با الگوریتم ORSA و حذف تناظرهای اشتباه، با الگوریتم RANSAC انجام شده‌است. برای

www.SID.ir

حل مشکل انسداد، از اطلاعات تاریخچه مقیاس هدف در ۱۰ فریم متوالی، استفاده می‌شود. روش پیشنهادی با زمان اجرای ۱۶/۳ ثانیه نسبت به سایر روش‌ها به سرعت و دقت بالاتری دست یافته‌است.

مرجع [۵] به طراحی الگوریتم‌های تخمین براساس تقریب جریان نوری، متناسب با یک GPU و در چارچوب CUDA می‌پردازد. هدف از فرمول‌بندی مجدد مسئله بعنوان تشخیص تغییرات در هنگام رخداد حالت‌های جدید و با استفاده از ویژگی‌های ردیابی شده قبلی، با بکارگیری پردازش گرافیکی، رسیدن به نتیجه منطقی در سریع‌ترین زمان ممکن می‌باشد. دو الگوریتم موازی پیشنهاد شده‌اند: درون‌یابی مستقیم و آزمایش فرضیه‌های متعدد. هر دو روش روی واحد پردازش گرافیکی، با سرعت نزدیک به ویدیو اجرا می‌شود که نتیجه قابل قبولی را برای کاربردهای حساس به زمان فراهم می‌کند. نتیجه آزمایش‌ها نشان می‌دهد، الگوریتم‌های انتخاب شده متناسب با واحد پردازش گرافیکی، با اجرای حدود ۲۴۰ برابر سریع‌تر از پیاده‌سازی‌های سریالی معادل، که در اجرا بسیار کند هستند، عمل می‌کنند.

مرجع [۶] به بررسی ویژگی‌های متمایز GPU، در پیاده‌سازی چند الگوریتم پردازش تصویر کلاسیک با استفاده از CUDA می‌پردازد. الگوریتم‌هایی مانند یکسان‌سازی هیستوگرام، حذف ابرها، تشخیص لبه، رمزگذاری و رمزگشایی DCT، برای پیاده‌سازی سیستم معرفی شده‌اند. با در نظر گرفتن زمان انتقال داده بین حافظه میزبان و حافظه دستگاه، با افزایش اندازه تصویر، محاسبه هیستوگرام، سرعتی بیشتر از ۴۰ برابر بدست آورده‌است. همچنین روش حذف ابر حدود ۷۹ برابر، DCT حدود ۸ برابر و روش تشخیص لبه‌ای بیش از ۲۰۰ برابر نسبت به پیاده‌سازی سریالی افزایش سرعت داشته‌است.

در [۷] یک روش تطبیقی، با در نظر گرفتن اطلاعات پیشینه هدف، در رابطه فضایی بین تصاویر پیشنهاد شده است. منطقه همپوشانی، بعنوان یکی از محدودیت‌ها برای کاهش دامنه جستجو در طول فرآیند تطبیق تصویر استفاده می‌شود. در این روش، الگوریتم تطبیق تصویر براساس معماری CUDA موازی‌سازی می‌شود. نتیجه آزمایش‌ها نشان می‌دهد که در مقایسه با طرح سریالی بر روی CPU، کارایی این روش در پیاده‌سازی موازی روی واحدهای پردازش گرافیکی، با سرعت اجرای بیشتر از ۳۰ برابر و با ۱۲.۸ فریم برثانیه بهبود یافته است.

در مرجع [۸] یک سیستم بینایی تک لنز RGB، برای تخمین حالت و موقعیت یک پهباد بدون سرنشین، ارائه شده‌است. با استفاده از این تخمین، یک ایستگاه کنترل زمینی می‌تواند مسیر پهباد را در حین فرود روی یک قایق گشت‌زنی تندرو کنترل کند. در پیاده‌سازی سیستم پیشنهادی، از یک رویکرد مبتنی‌بر مدل سه بعدی براساس فیلتر ذره‌ای استفاده شده‌است که به پنج مرحله تقسیم می‌شود: ثبت فریم، تشخیص هدف، تصحیح اعوجاج، نمونه بردار حالت مبتنی‌بر ظاهر و مرحله تخمین حالت شی. تشخیص موقعیت هدف با استفاده از یک آشکارساز مبتنی‌بر شبکه عصبی عمیق، تشخیص داده شده و پس از تصحیح اعوجاج‌های فریم، از یک پایگاه‌داده برای مقداردهی اولیه الگو، استفاده می‌شود. در مرحله آخر یک الگوریتم بهینه‌سازی، برای تخمین وضعیت پهباد، اعمال می‌شود. طبق نتایج، افزایش عملکرد سیستم با استفاده از واحد پردازش گرافیکی، نشان می‌دهد که بکارگیری منابع محاسباتی GPU در یک سیستم تخمین حالت بلادرنگ ضروری و در کیفیت اجرا بسیار موثر است.

۳- تشخیص و ردیابی اهداف متحرک

در یک سیستم تشخیص و ردیابی تصویری، نمایش شی یا مدل‌سازی، از آن جهت که شی هدف را مشخص می‌کند، در کیفیت عملکرد الگوریتم‌های ردیابی، نقش بسزایی دارد. ارائه یک توصیف بصری مناسب با ساختار شی هدف، بدلیل تمایز هدف از سایر اشیاء موجود در تصویر و پس زمینه، مولفه مهمی است. روش‌های متعددی برای نمایش شی وجود دارد که مسئله محور می‌باشد و متناسب با نوع هدف بکار برده می‌شود. برخی از روش‌ها برای نمایش اهداف پیچیده مناسب‌تر هستند و بعضی دیگر برای نمایش اشیاء ساده استفاده می‌شوند. در یک دسته‌بندی کلی، روش‌های نمایش عبارتند از: نمایش نقطه‌ای، نمایش براساس اشکال هندسی و نمایش براساس کانتور.

در روش نقطه‌ای، هدف براساس نقاط کلیدی و یا براساس نقطه‌ی مرکز ثقل شی، نمایش داده می‌شود. در روش استفاده از شکل‌های هندسی، معمولا شی در یک قاب مستطیل یا بیضی قرار می‌گیرد. در شکل دیگری از این روش، اجزای تشکیل دهنده شی، با استفاده از اشکال هندسی یکسان و متصل، بصورت بند بند نمایش داده می‌شوند. در روش کانتور، سه حالت وجود دارد که مرز لبه‌های

www.SID.ir

شی با استفاده از نقاط، و یا به شکل خطوط مرزی لبه، مشخص می‌شود؛ در حالت سوم از نمایش کانتور، شی هدف بصورت کانتور کامل یا سایه نما نشان داده می‌شود.

در ردیابی اهداف متحرک، روش‌های بسیاری وجود دارد که در یک دسته‌بندی کلی به دو روش شناسایی متوالی و ردیابی مبتنی‌بر فیلتر تقسیم می‌شوند. در ردیابی هدف با فیلتر، از الگوریتم‌های قطعه‌بندی برای جداسازی نواحی تصویر استفاده می‌شود. روش‌های قطعه‌بندی، بر مبنای ناحیه، مرز و لبه‌ها عمل می‌کنند. در روش قطعه‌بندی مبتنی‌بر ناحیه، هر پیکسل به یک ناحیه مشخص برچسب می‌خورد. در رویکرد لبه، ابتدا لبه‌های نواحی با الگوریتم‌های لبه‌یابی آشکارسازی شده و با اتصال این لبه‌ها مرز اشیاء شناسایی می‌شود.

در مبحث آشکارسازی لبه‌ها، دو روش معمول در تصاویر، استفاده از عملگرهای گرادیان و لاپلاسین، یعنی مشتق مرتبه اول و دوم می‌باشد. این عملگرها به عنوان فیلترهای بالاگذر پایه شناخته می‌شوند. به منظور آشکارسازی لبه‌ها، الگوریتم‌ها و متدهایی در جهت بهبود روش‌های پایه فوق طراحی شده‌اند که عبارتند از: Sobel، Robert، Prewitt، Canny، Log، Zerocross .

متد Sobel، لبه‌ها را با استفاده از تخمین مشتق پیدا می‌کند و هر جا شیبی در مقدار پیکسل‌ها بیابد، آنجا را لبه معرفی خواهد کرد؛ این متد، لبه‌ها را در نقاطی بر می‌گرداند که گرادیان تصویر ماکزیمم است و دو ماسک برای پیداکردن لبه‌های افقی و عمودی دارد. متد Prewitt نیز از تقریب مشتق، در شناسایی لبه‌ها استفاده می‌کند و شباهت زیادی با الگوریتم Sobel دارد، با این تفاوت که ضرایب ماسک آنها متفاوت است. این متد شامل تصریح کننده تصویر نیز می‌باشد.

متد Robert به دلیل حساسیت زیاد به نویز، پیکسل‌های کمتری برای تخمین مشتق بکار می‌برد. این متد از روش تقریب مشتق تصویر برای یافتن مرزها استفاده می‌کند و تنها تفاوت آن با دو متد Prewitt و Sobel، وجود افزایش‌دهنده حجم لبه می‌باشد. الگوی این متد به این صورت است که به مرزهای با شیب ۴۵ درجه حساسیت نشان می‌دهد. فیلتر canny یکی از معروف‌ترین روش‌های آشکارساز لبه در پردازش تصویر است که برای لبه‌یابی از یک الگوریتم با ۵ مرحله اساسی استفاده می‌کند، که به ترتیب شامل اعمال فیلتر گوسی، یافتن گرادیان شدت روشنایی تصویر، حذف نقاط غیربیشینه، آستانه‌گذاری دوگانه برای تشخیص لبه‌های قوی و دنبال کردن لبه‌های تصویر با روش پسماند می‌باشد.

متد Log از روش لاپلاس تصویر، برای شناسایی لبه استفاده می‌کند. لاپلاس، یک اندازه‌گیری ایزوتروپیک دوبعدی از مشتق فضایی مرتبه دوم از یک تصویر است. لاپلاس یک تصویر، مناطق تغییرات شدت سریع را نشان می‌دهد و بنابراین اغلب برای تشخیص لبه استفاده می‌شود. در این فیلتر ابتدا تصویر با فیلتر گووسی هموار شده و سپس با عبور از فیلتر لاپلاسین، لبه‌ها در تصویر آشکار می‌شود. الگوریتم Zerocross، یکی دیگر از متدهای محبوب برای آشکارسازی لبه و با حساسیت بسیار بالا می‌باشد. این متد قسمت‌هایی از لاپلاس یک تصویر را جستجو می‌کند که مقدار لاپلاس از صفر می‌گذرد. به عبارت دیگر نقاطی که لاپلاس، علامت را تغییر می‌دهد. این متد بصورت همزمان عمل هموارسازی را نیز انجام می‌دهد. در این مقاله، از متدهای لبه‌یاب ذکر شده، برای تشخیص مرز شی هدف، در فریم‌های متوالی از رشته ویدیو، استفاده شده‌است.

٤- روش پیاده‌سازی

در روش پیشنهادی ابتدا در فریم اول از ویدیو (لحظه t) هدف متحرک (جنگنده) بصورت دستی انتخاب و مختصات شی هدف، برای استفاده در فریم‌های بعدی استخراج می‌شود. شی هدف از پس زمینه جدا شده و بصورت کانتور کامل، در یک تصویر باینری نمایش داده می‌شود. سپس لبه‌یابی در فریم‌های بعدی بطور متوالی براساس مختصات بدست آمده از فریم قبل، انجام می‌شود. در پردازش فریم‌ها توسط CPU، هر لحظه یک فریم از ویدیو واکشی و پردازش می‌شود؛ اما در پردازش با GPU مربوط به رایانه بکار برده شده در این تحقیق، همزمان دو فریم جهت پردازش واکشی می‌شود. این امر موجب افزایش سرعت اجرا می‌گردد، اما همچنان عواملی مثل سرعت دسترسی GPU به حافظه موجب کاهش سرعت اجرا خواهد شد. مقادیر حاصل از اندازه‌گیری زمان اجرا، در واحدهای پردازش گرافیکی و واحد پردازش مرکزی، برحسب ثانیه، استخراج و سپس میزان افزایش سرعت اجرا طبق رابطه (۱) محاسبه می‌شود:

$$Speedup = T(S) / T(P) \quad (۱)$$

که در آن T(S)، میزان سرعت اجرا با استفاده از CPU و T(P) سرعت اجرا با استفاده از GPU، برحسب ثانیه می‌باشد. دقت در تشخیص هدف و همپوشانی فریم‌ها با قاب اصلی، جهت تشخیص صحیح هدف نیز، با رابطه (۲) محاسبه می‌شود:

www.SID.ir

$$Precision = TP/(TP+FP) \quad (۲)$$

که در آن، TP تعداد فریم‌هایی را، که هدف در آن وجود داشته و سیستم نیز درست تشخیص داده، شامل می‌شود؛ FP شامل تعداد فریم‌هایی می‌باشد که هدف در آن نبوده و سیستم به اشتباه، پس زمینه را هدف تشخیص داده‌است.

برای تشخیص شی هدف، در دنباله‌ای از تصاویر ویدیویی، شش الگوریتم لبه‌یابی اصلی استفاده شده است. این الگوریتم‌ها عبارتند از: Sobel، Robert، Prewitt، Canny، Log و Zerocross . هدف از این مقاله، ارزیابی سرعت و دقت هدف‌یابی این الگوریتم‌ها در استفاده از قابلیت‌های واحدهای پردازش گرافیکی، بجای استفاده از توان واحد پردازش مرکزی می‌باشد. همچنین با ارزیابی میزان دقت هدف‌یابی، به تاثیر استفاده از واحد پردازش گرافیکی در بهبود عملکرد الگوریتم‌های لبه‌یابی، پی خواهیم برد. در نهایت قصد داریم با مقایسه مقادیر بدست آمده از اجرای هر لبه‌یاب با سیستم و رشته ویدیو یکسان، میزان توانایی GPU نسبت به CPU را در فرآیند تشخیص و ردیابی هدف بسنجیم. این روش در اکثر سیستم‌های مبنی‌بر پردازش تصویر، که سرعت اجرا و کیفیت عملکرد فرآیند بسیار مهم است، می‌تواند کاربردی باشد.

۵- ابزارهای شبیه‌سازی

یکی از قطعات سخت افزاری در زمینه پردازش تصویر، واحد پردازش گرافیکی است که به راه حلی مقرون به‌صرفه برای تسریع فرآیند یک سیستم تجاری تبدیل شده‌است. عملکرد یک الگوریتم اگر چه ممکن است در یک CPU بی فایده باشد اما می‌توان آن را با سخت افزار موازی طراحی کرد و شاهد عملکرد بهتر الگوریتم بود [۵]. با پیشرفت تکنولوژی، پیاده‌سازی فرایند پردازش تصویر، تشخیص و ردیابی هدف، بصورت مجازی، با استفاده از امکاناتی که رایانه‌های شخصی دارند، از جمله واحدهای پردازش مرکزی، واحد پردازش گرافیکی، حافظه‌های اصلی و جانبی امکان‌پذیر شده‌است.

برای ارزیابی روش پیشنهادی، شبیه‌سازی سیستم تشخیص و ردیابی هدف جنگنده، و همچنین فرآیند اندازه‌گیری سرعت و دقت اجرای سیستم با استفاده از واحد پردازش مرکزی و واحدهای پردازش گرافیکی، از محیط شبیه‌سازی نرم افزار مهندسی متلب و قابلیت کودا استفاده شده‌است. کودا یک سکوی پردازش موازی است که به توسعه دهنده‌گان نرم افزار اجازه می‌دهد از یک GPU با قابلیت کودا در پردازش استفاده کنند. استفاده از کودا موجب می‌شود GPU قابلیت پردازش‌هایی مانند CPU و همچنین امکان دسترسی به حافظه را داشته باشد.

استفاده از نرم افزار متلب از جهت وجود ابزارهای دقیق و مناسب، و همچنین سهولت در مدیریت فایل‌ها، در بیشتر پژوهش‌های مرتبط، برای استفاده بعنوان یک محیط مجازی در شبیه‌سازی سیستم، بسیار مورد توجه می‌باشد. در کنار قدرت نرم افزار متلب، وجود رایانه‌های شخصی انجام چنین تحقیقاتی آسان‌تر شده است. با قابلیت های سخت افزاری موجود، مزیت استفاده از رایانه در چنین سیستم‌هایی، تسریع در فرایندهای مقایسه و ارزیابی عملکرد سیستم، دسترسی سریع به داده‌ها و انتقال اطلاعات، ویژگی چند رسانه‌ای بودن و استفاده از ابزارهای دقیق اندازه‌گیری می‌باشد. برای پیاده‌سازی سیستم پیشنهادی و همچنین ارزیابی سرعت و دقت، از یک رایانه شخصی بعنوان آزمایشگاه مجازی استفاده شده‌است. مشخصات سیستم بکار گرفته شده، در جدول (۱) درج شده‌است.

جدول ۱- مشخصات رایانه موجود بعنوان آزمایشگاه مجازی

اجزا	مشخصات
سیستم عامل (OS)	Microsoft Windows 10 Pro
حافظه اصلی (RAM)	8.00 GB
واحد پردازنده مرکزی (CPU)	Intel(R) Core(TM) i5-2450M CPU @ 2.50GHz 2.50 GHz
واحد پردازنده گرافیکی (GPU)	Intel(R) HD Graphics 3000 NVIDIA GeForce GT 525M
نرم افزار شبیه‌سازی	Matlab R2016b

www.SID.ir

در این مقاله از پایگاه‌داده استاندارد AIRCRAFT TRACKING استفاده شده‌است [۹,۴]. این پایگاه‌داده شامل ۱۴ ویدیو با ابعاد ۷۲۰*۵۷۶ پیکسل، با چالش‌هایی از جمله حرکت سریع هدف جنگنده، انسداد جزئی هدف با ابرها، حرکت دوربین، چرخش هدف، تغییر مسیر حرکت، انعکاس نور از سطح شی هدف، تعداد هدف در زمینه می‌باشد. از بین ویدیوهای موجود سه ویدیو برای ارزیابی سرعت و دقت سیستم پیشنهادی در عملکرد لبه‌یاب‌ها استفاده شده است، که از هر ویدیو تعداد ۱۰۰ فریم مورد پردازش قرار می‌گیرد. در شکل (۱) تصاویر باینری از لبه‌یابی هدف در آخرین فریم پردازش شده از دنباله‌های ویدویی aerobatics_1 و aerobatics_2 را مشاهده می‌کنید.

شکل ۱- خروجی الگوریتم Robert روی فریم‌های ویدیویی

٦- نتیجه آزمایش‌ها

در ارزیابی سرعت و دقت سیستم پیشنهادی، هر یک از ۶ الگوریتم لبه‌یاب، بطور جداگانه با ۳ رشته ویدیو، اجرا شده‌است. ویدیو اول شامل چالش‌های سرعت حرکت جنگنده، چرخش هدف و تغییر مسیر حرکت می‌باشد. در ویدیو دوم، سیستم با چالش‌های انسداد جزئی، بازتاب نور خورشید از سطح شی هدف، دید کم و همرنگ بودن شی با پس زمینه مواجه است که تشخیص هدف از پس زمینه را سخت‌تر می‌کند. در ویدیو سوم چالش‌های سرعت حرکت، چرخش هدف، تغییر زاویه شی، حرکت دوربین، تابش نور خورشید به سطح شی هدف، تغییرات روشنایی و انسداد شی هدف با ابر وجود دارد.

در جدول (۲) میزان سرعت اجرا برای هر الگورریتم لبه‌یاب، به ازای ۳ دنباله ویدیویی، درج شده است. مقدار کمتر از ۱ نشان‌دهنده کاهش سرعت اجرای روش پیشنهادی نسبت به CPU، و مقادیر بالای ۱ نشان‌دهنده میزان افزایش سرعت می‌باشد.

جدول ۲- میزان سرعت اجرا CP

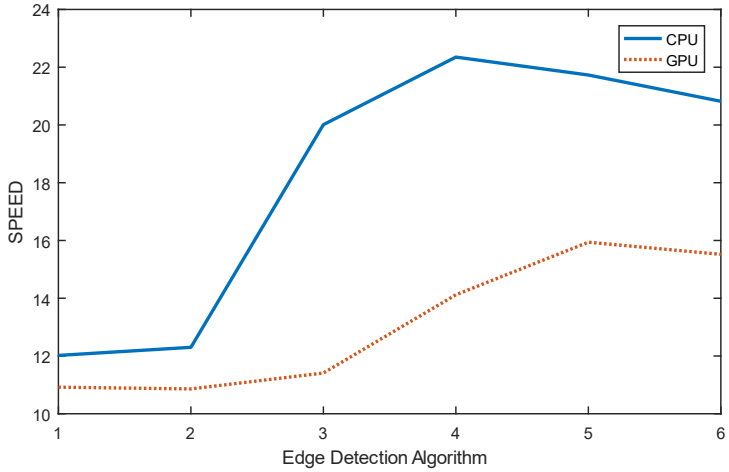
Algorithm	(Video 1) aerobatics_1	(Video 2) aerobatics_ specularities_1	(Video 3) aerobatics_2
Sobel	1.10	1.03	0.88
Prewitt	1.13	1	1
Robert	1.75	1.05	1
Canny	1.58	1.11	1.07
Log	1.36	1.13	1.15
Zerocross	1.34	1.02	1.07

نتیجه آزمایش‌ها نشان می‌دهد، سرعت اجرای الگوریتم‌ها با GPU، در بیشتر موارد، بالاتر از سرعت اجرای واحد پردازش مرکزی است. در ویدیو اول، الگوریتم Robert، ٪۷۵ و الگوریتم Canny، ٪۵۸ سریع‌تر اجرا شده‌اند و سرعت اجرا در سایر الگوریتم‌ها نیز افزایش داشته است. با

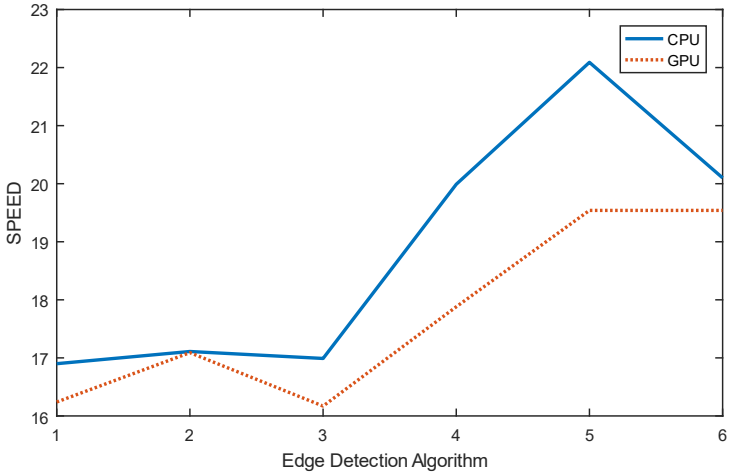
www.SID.ir

Arch

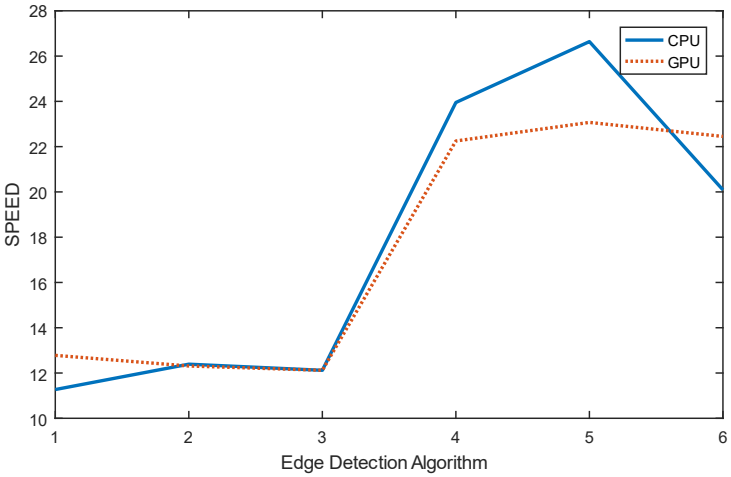
توجه به این که تنها ۱۰۰ فریم از ویدیوها پردازش شده است، سرعت اجرا در سیستم‌های پیشرفته‌تر، با توان سخت افزاری بالاتر و همچنین با استفاده از الگوریتم‌های بهینه شده در زمینه سرعت اجرا، زمانیکه با حجمی از تصاویر متوالی مواجه هستند، می‌تواند با تغییرات قابل توجهی همراه باشد. نمودار سرعت اجرا برای ۳ رشته ویدیو و ۶ الگوریتم لبه‌یاب، با استفاده از واحدهای پردازش گرافیکی و پردازش اصلی، به ترتیب در شکل‌های (۲، ۳، ۴) درج شده‌است.



شکل ۲- نمودا سرعت اجرا در ویدیو شماره ۱



شکل ۳- نمودا سرعت اجرا در ویدیو شماره۲



شکل ۴- نمودا سرعت اجرا در ویدیو شماره۳

www.SID.ir

برای سامانه‌های امنیتی و نظامی، یکی از فاکتورهای بسیار مهم، سرعت اجرای سیستم تشخیص هدف است. در پیاده‌سازی سیستم پیشنهادی، برای افزایش سرعت اجرا در نرم افزار متلب، از نمایش پنجره‌های گرافیکی به کاربر، تا حد امکان صرفه نظر شده‌است؛ بدین صورت، زمان برای اجراهای گرافیکی نرم افزار به حداقل رسیده و اجرای سیستم در کمترین زمان انجام می‌شود.

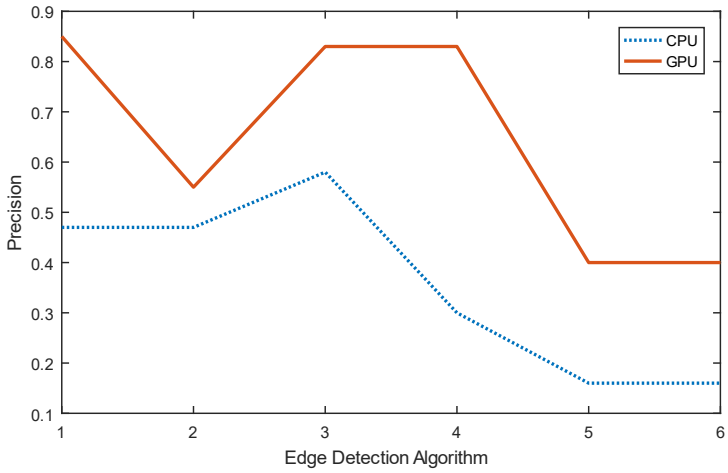
در جدول (۳) میزان همپوشانی ناحیه هدف با خروجی حقیقی نشان داده شده‌است. با توجه به مقادیر حاصل از محاسبه دقت تشخیص هدف، عملکرد صحیح الگوریتم‌ها در اجرا با استفاده از CPU و GPU نیز بسیار متفاوت است و روش پیشنهادی به میزان قابل توجهی در بهبود عملکرد سیستم موثر بوده‌است. طبق نتایج به دست آمده و اندازه‌گیری‌های انجام شده با استفاده از توان محاسباتی نرم افزار متلب، میزان دقت الگوریتم‌های آشکارسازی لبه، در بعضی شرایط نزدیک به ۳ برابر بوده‌است؛ اما در بعضی شرایط الگوریتم لبه‌یابی، توانایی تشخیص هدف را در اجرا با استفاده از CPU نداشته و در اجرا با استفاده از GPU به میزان دقت ۹۰ برابر رسیده‌است.

جدول۳- تفاوت میزان دقت تشخیص هدف در CPU و GPU

Algorithm	aerobatics_1		aerobatics_ specularities_1		aerobatics_2	
	CPU	GPU	CPU	GPU	CPU	GPU
Sobel	0.47	0.85	0.50	0.94	0	0.90
Prewitt	0.47	0.55	0.87	0.94	0	0.90
Robert	0.58	0.83	0.33	0.80	0.25	1
Canny	0.30	0.83	0.87	1	0.03	0.45
Log	0.16	0.40	0.33	0.81	0.1	0.39
Zerocross	0.16	0.40	0.33	0.81	0.01	0.50

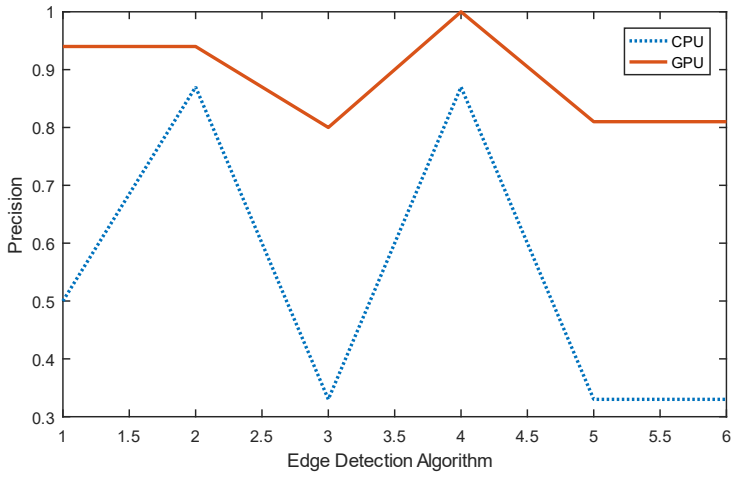
با توجه به مقادیر مربوط به میزان دقت تشخیص هدف در ویدیو سوم، CPU توانایی پردازش بسیار کمی در تشخیص اهدافی که دارای چالش‌هایی مثل دید و مقیاس کم، فاصله زیاد از دوربین، بازتاب نور و همچنین همرنگ بودن با پس زمینه هستند، توانایی بسیار کمی دارد. اما توانایی واحدهای پردازش گرافیکی در تشخیص هدف در چنین مواردی بسیار بالاتر از واحد پردازش مرکزی می‌باشد. همچنین در سایر اجراها، دقت واحدهای پردازش گرافیکی، بسیار متفاوت‌تر و بیشتر از دقت واحد پردازش مرکزی است. میزان دقت تشخیص در سامانه‌های امنیتی و نظامی به دلیل اهمیت تشخیص صحیح هدف از شی غیر هدف، بسیار مهم و ضروری است.

از بین الگوریتم‌های مورد ارزیابی، Robert در تشخیص هدف با دید کم بهترین عملکرد را با استفاده از واحدهای پردازش گرافیکی دارد. بعد از آن Sobel و Prewitt در مواجه با سایر چالش‌ها، بهتر از بقیه هدف را تشخیص می‌دهند. نمودار دقت اجرای سیستم برای ۳ ویدیو، و ۶ الگوریتم، با استفاده از واحدهای پردازش گرافیکی و پرازش اصلی، به ترتیب در شکل‌های (۵، ۶، ۷) درج شده‌است.

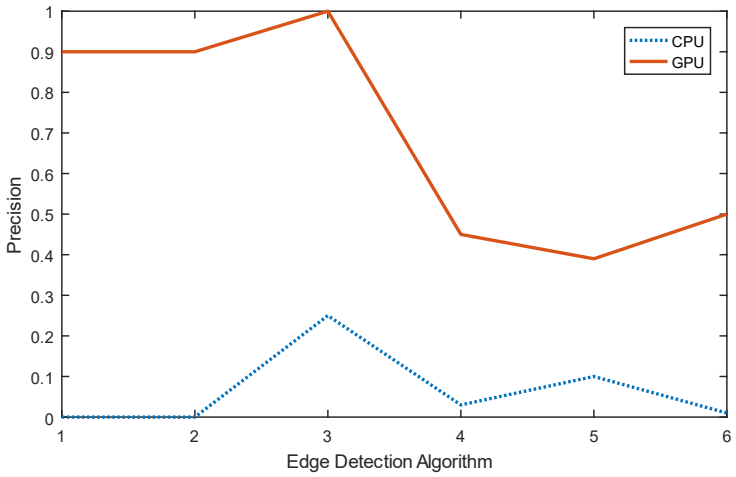


شکل ۵- نمودار دقت تشخیص هدف توسط لبه‌یاب‌ها در ویدیو شماره ۱

www.SID.ir



شکل ۶- نمودار دقت تشخیص هدف توسط لبه‌یاب‌ها در ویدیو شماره ۲



شکل ۷- نمودار دقت تشخیص هدف توسط لبه‌یاب‌ها در ویدیو شماره ۳

۷- نتیجه

در این مقاله به بررسی سرعت اجرا و دقت هدف‌یابی در سیستم تشخیص هدف متحرک (جنگنده)، با واحدهای پردازش گرافیکی و واحد پردازش مرکزی پرداخته‌ایم. در روش پیشنهادی برای تشخیص و ردیابی هدف در دنباله‌ای از تصاویر ویدیویی، از الگوریتم‌های تشخیص لبه، استفاده شده‌است. در پیاده‌سازی سیستم، با بکارگیری توان و قابلیت‌های واحد پردازش گرافیکی و قابلیت کودا برای پردازش موازی، به ارزیابی میزان تفاوت سرعت اجرا و دقت هدف‌یابی CPU و GPU در سامانه‌های تشخیص هدف متحرک پرداخته‌ایم.

با توجه به آزمایش‌های انجام شده، استفاده از واحدهای پردازش گرافیکی، در سیستم‌های پردازش تصویر و تشخیص هدف متحرک، موجب بهبود سرعت اجرا و افزایش دقت هدف‌یابی می‌شود. این روش، به موجب اجرای موازی فرآیند پردازش، نسبت به روش پیاده‌سازی سیستم با واحد پردازش مرکزی، در سرعت اجرا بهینه‌تر و همچنین از دقت بسیار بالاتری در پردازش تصویر برخوردار است. در پیاده‌سازی سیستم تشخیص هدف، برای سامانه‌های امنیتی و نظامی، سرعت اجرا و دقت در هدف‌یابی اهمیت ویژه‌ای دارد. استفاده از توان و قابلیت‌های واحد پردازش گرافیکی، در چنین سیستم‌هایی، می‌تواند بسیار کارآمدتر و بصرفه‌تر نیز باشد.

استفاده از روش پیشنهادی، از نظر کاهش خطاهای هدف‌یابی و کاهش زمان اجرا می‌تواند در بسیاری از سامانه‌های مرتبط، کاربردی باشد. افزایش سرعت اجرا در سیستم بررسی شده، با روش پردزاش موازی، می‌تواند در مقیاس گسترده‌تر، به رقم قابل توجهی برسد و در سیستم‌هایی که افزایش سرعت اولویت بیشتری دارد کاربردی باشد.

طبق نتایج، افزایش دقت الگوریتم‌ها، از ۳ برابر تا ۹۰ برابر در روش پیشنهادی، نشان می‌دهد پیاده سازی روش‌های بهینه شده قبلی در پردازش تصویر، که با اجرای سریالی صورت گرفته به روش پیشنهادی می‌تواند بسیار مطلوب‌تر باشد؛ چرا که ممکن است اجرای یک الگوریتم به روش سریالی به قدر کافی موثر نباشد اما با روش موازی به سیستمی کاربردی و بهینه منجر شود.

www.SID.ir

مراجع

[۱] سرخیل، محمدرضا؛ "بهبود الگوریتم ردیابی هدف با استفاده از مدل قطعه‌بندی موثر تصاویر ویدیویی"، *چهارمین کنفرانس ملی دانش و فناوری مهندسی برق کامپیوتر و مکانیک ایران*، ۱۸-۱، تهران، ۱۳۹۹.

[۲] اکبری سکه روانی، احسان؛ مسعودی، مهدی؛ "ردیابی و تشخیص بیدرنگ اهداف نظامی پرنده از غیر نظامی در تصاویر ویدیوی"، *مجله کهربا*، دوره ۶، شماره ۲۳، ۹-۱، ۱۳۹۸.

[۳] محرابی، محمدعلی، "ارائه الگوریتم شناسایی و ردیابی هدف بر پایه روش‌های پردازش تصویر"، کارشناسی ارشد، موسسه آموزش عالی شهاب دانش، قم، صفحه ۱۱۴-۱، ۱۳۹۶.

[۴] آزاد زاده، وحید؛ لطیف، علی محمد؛ "روشی جدید در ردیابی اهداف متحرک هوایی با استفاده از الگوریتم AFFINE – SIFT"، *نشریه علمی پدافند الکترونیک و سایبری*، شماره ۱، صفحه ۵۹-۴۷، دانشگاه یزد / سازمان قضایی نیروهای مسلح استان خراسان شمالی، ۱۳۹۵.

[5] Kim, Jun S., Hwangbo, M., and Kanade, T.; "Parallel algorithms to a parallel hardware: Designing vision algorithms for a GPU", *12th International Conference on Computer Vision Workshops, ICCV Workshops*. IEEE, 2009.

[6] Yang, Y. Z., Pu, Y.; "Parallel Image Processing Based on CUDA", *2008 International Conference on Computer Science and Software Engineering*. IEEE, Vol. 3, No. 100. 2008.

[7] Guo, Y., Shi, M., Li, Y., and Liu, D.; "Research on Fast Image Mosaic Based on CUDA", *2017 Fourth International Symposium on Computational Intelligence and Design*. IEEE, Vol. 1, pp. 401-405, 2017.

[8] Santos, N. P., Lobo, V., and Bernardino, A.; "3D Model-Based UAV Pose Estimation using GPU", *OCEANS 2019 MTS/IEEE SEATTLE*. IEEE, Vol. 102, No. 2, 2019.

[9] Ajmal S. M.; "Realtime Visual Tracking of Aircrafts", *2008 Digital Image Computing: Techniques and Applications*, IEEE, pp. 351-356 ,2008.

www.SID.ir