

ردیابی دقیق اشیای متحرک با استفاده از الگوریتم‌های DBSCAN و KLT، Sift

عزیز کرمانی و عسگرعلی بویر

دارای زمان مصرفی کم و دقت بالا باشند منجر به افزایش سطح کیفیت عملکرد این سیستم‌ها خواهد شد. بر اساس تعریف ارائه شده در [۱]، ردیابی اشیای متحرک، دنبال کردن مسیر حرکت یک شیء یا اشیای متحرک در دنباله‌ای از تصاویر ورودی است. اشیای متحرک مورد ردیابی می‌تواند هر شیء متحرکی مانند ماهی درون آب، قایق درون دریا، عابر پیاده در پیاده‌رو، اتومبیل در بزرگراه‌ها و غیره باشد که بر اساس کاربرد نیازمند به تعیین مکان آنها هستیم. به عنوان مثال در یک سیستم ناوبری خودکار (اتومبیل بدون راننده) باید بتوان اشیای متحرک موجود در اطراف سیستم را تشخیص داده و در هر زمان موقیت این اشیا را برای اجتناب از برخورد با دیگر اشیا تعیین کرد. الگوریتم‌های ردیابی باید بتوانند در مقابل چالش‌های زیادی مانند تغییرات شدت روشنایی محیط، تغییر جهت ناگهانی اشیا، وجود انواع مختلف اشیا در حوزه دید دوربین، همپوشانی و غیره از خود انعطاف نشان دهند.

ادامه مقاله به صورت زیر بخش‌بندی شده است: در بخش ۲ کارهای مربوط معرفی می‌شوند. در بخش ۳ روش پیشنهادی و در نهایت در بخش ۴ نتیجه‌گیری روش پیشنهادی آورده شده است.

۲- کارهای مربوط

ادبیات ردیابی به علت کاربردهای گسترده و قابل توجه در زمینه امنیت و آسایش بشر روز به روز در حال پیشرفت می‌باشد به گونه‌ای که خود را از سایه کاربردهای نظامی خارج کرده و کاربرد خود را در سیستم‌های نوین مورد استفاده بشر نشان داده است. با توجه به مقالات موجود در زمینه ردیابی اشیای متحرک می‌توان گفت که شروع کار در مورد ردیابی به طور تقریبی از اوایل دهه ۱۹۸۰ شروع شده و در طول سالیان متمادی دامنه و گستردگی آن افزایش یافته است. از جمله کارهای جامع و کامل در زمینه ردیابی اشیای متحرک که به مرور و طبقه‌بندی روش‌های ردیابی تا سال ۲۰۰۶ پرداخته است، توسط ایلماز انجام شده است [۱]. از جمله کارهای مهم دیگر که کاربردها و روش‌های ردیابی مختلف را مورد تجزیه و تحلیل قرار داده‌اند می‌توان به [۲] تا [۶] اشاره کرد. نویسندگان [۷] ردیابی اشیای متحرک را با استفاده از ویژگی‌های لبه‌های متحرک انجام دادند. در این مقاله ابتدا لبه‌های ثابت پس‌زمینه با الگوریتم لبه‌یابی کنی^۶ استخراج شده و در هر مرحله از میان لبه‌های موجود در فریم جاری لبه‌های ثابت پس‌زمینه و لبه‌هایی با جابه‌جایی جزئی را با استفاده از تفاضل لبه‌های متحرک از لبه‌های ثابت از هم متمایز ساخته و در نهایت با استفاده از لبه‌های متحرک کشف شده، عمل ردیابی را انجام دادند. در [۸] کشف و ردیابی اشیای متحرک را با به روز رسانی مدل پس‌زمینه با الگوریتم فیلتر میانه انجام داده‌اند. آنها در هر گام پس‌زمینه را با جایگذاری مقدار میانه همسایه‌های هر پیکسل به روز رسانی کرده و با اعمال

چکیده: کشف و ردیابی اشیای متحرک گامی اساسی در تجزیه و تحلیل ویدئو می‌باشد. در این مقاله روشی جدید را برای ردیابی هم‌زمان چندین شیء متحرک در حوزه دید دوربین ثابت ارائه خواهیم کرد. در روش پیشنهادی مکان اشیای متحرک موجود در حوزه دید دوربین را در هر مرحله و با استفاده از اطلاعات حرکت موجود بین دو فریم متوالی شامل فریم قبلی و فریم جاری از نظر زمانی تعیین می‌کنیم. در هر مرحله نقاط ویژگی Sift را روی فریم قبلی استخراج کرده و تناظر این نقاط ویژگی را با استفاده از الگوریتم تناظریابی نقاط کلیدی KLT روی فریم جاری به دست می‌آوریم. در ادامه و با در اختیار داشتن نقاط ویژگی متناظر بین دو فریم متوالی، اندازه حرکت نقاط ویژگی را محاسبه کرده و با حذف نقاط ویژگی با جابه‌جایی ثابت و یا ناچیز، نقاط ویژگی مرتبط به اشیای متحرک را کشف خواهیم کرد. سپس نقاط ویژگی برچسب‌گذاری شده به عنوان اشیای متحرک را با استفاده از الگوریتم خوشه‌بندی DBSCAN به خوشه‌های مختلف به عنوان اشیای متحرک دسته‌بندی می‌کنیم. با این روش و در هر لحظه مکان تمامی اشیای متحرک موجود در حوزه دید دوربین به دست آمده که با تناظریابی یک به یک بین این اشیا و اشیای به دست آمده در فریم قبلی مکان جدید هر شیء را تعیین می‌کنیم. نتایج روش پیشنهادی حاکی از دقت بالا و زمان مصرفی قابل قبول برای ردیابی اشیای متحرک می‌باشد. روش پیشنهادی دارای دقت ۹۵٪ برای ردیابی اشیای متحرک بوده و در هر ثانیه ۳۳ فریم را پردازش می‌کند که در مقایسه با روش‌های معمول از نظر دقت و سرعت عملکرد مطلوبی دارد.

کلیدواژه: اشیای متحرک، ردیابی، DBSCAN، KLT، SIFT.

۱- مقدمه

کشف و ردیابی اشیای متحرک گامی اساسی در تجزیه و تحلیل ویدئو می‌باشد و از این رو به طور گسترده‌ای در سیستم‌های بینایی ماشین مانند سیستم‌های نظارتی^۱، کنترل ترافیک^۲، ناوبری خودکار^۳، تعامل انسان با کامپیوتر^۴ و رباتیک^۵ استفاده می‌شود، زیرا سیستم‌های مذکور نیازمند دریافت و پردازش ویدئوهای دریافتی از محیط اطراف خود و در نهایت تجزیه و تحلیل رفتار و رخدادهای موجود در این ویدئوها می‌باشند. از آنجایی که دقت و سرعت از عوامل مهم در عملکرد مطلوب سیستم‌های مذکور می‌باشند در نتیجه ارائه روش‌هایی برای ردیابی اشیای متحرک که

این مقاله در تاریخ ۱۸ مرداد ماه ۱۳۹۵ دریافت و در تاریخ ۹ فروردین ماه ۱۳۹۶ بازنگری شد.

عزیز کرمانی، دانشکده فناوری اطلاعات و مهندسی کامپیوتر، دانشگاه شهید مدنی آذربایجان، تبریز، (email: a.karamiani@azaruniv.edu).

عسگرعلی بویر، دانشکده فناوری اطلاعات و مهندسی کامپیوتر، دانشگاه شهید مدنی آذربایجان، تبریز، (email: a.bouyer@azaruniv.edu).

1. Automated Surveillance
2. Traffic Monitoring
3. Vehicle Navigation
4. Interaction Human Computer
5. Robotic

تعداد اشیا و سرعت‌های مختلف آنها حساسیت ندارد. در این روش ابتدا با استفاده از نقاط گوشه Sift و تناظریابی این نقاط با الگوریتم KLT و بر اساس اطلاعات حرکت نقاط کلیدی مرتبط به اشیا متحرک، آنها را کشف کرده و این نقاط را با استفاده از الگوریتم DBSCAN به عنوان اشیا متحرک موجود در صحنه ردیابی برچسب‌گذاری می‌کنیم و در هر مرحله این کلاسترها^۲ یا اشیا متحرک را روی فریم‌های دریافتی با استفاده از شباهت ناحیه‌شیء و بر اساس میانگین‌گیری از شدت روشنایی نقاط متعلق به آن ناحیه به دست می‌آوریم. شکل ۱ دیاگرام روش پیشنهادی را برای ردیابی هم‌زمان اشیا متحرک موجود در حوزه دید دوربین ثابت نشان می‌دهد.

۳-۱ دریافت فریم‌های ورودی

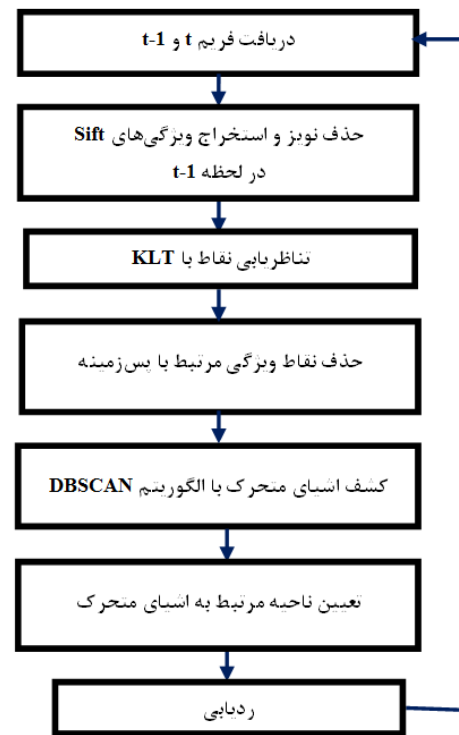
از آنجایی که در فرایند ردیابی به دنبال تعیین مکان شیء یا اشیا متحرک در طول زمان هستیم، در نتیجه باید تصویری پی در پی را از اشیا متحرک در اختیار داشته باشیم و سپس در هر لحظه مکان اشیا متحرک را تعیین کنیم. برای این کار با توجه به کاربرد مفروض می‌توان حالت‌های مختلفی را در نظر گرفت. به عنوان مثال در سیستم نظارتی از دوربین‌های نصب‌شده در یک مکان خاص و تصاویر دریافتی از آنها استفاده می‌شود. در این مقاله ما برای نمایش روند کار روش پیشنهادی از ویدئوهای آماده در [۲۲] تا [۲۴] استفاده می‌کنیم (حالت دوربین در این مقاله ثابت در نظر گرفته شده است).

۳-۲ حذف نویز

وجود نویز در تصاویر امری اجتناب‌ناپذیر است زیرا کیفیت یک تصویر تحت تأثیر عواملی زیادی قرار می‌گیرد. به عنوان مثال ابزار فیلم‌برداری، شرایط آب و هوایی، بافت پس‌زمینه ردیابی و تغییرات شدت روشنایی در طول شبانه‌روز تصاویر دریافتی را تحت تأثیر قرار داده و الگوریتم‌های ارائه‌شده در بینایی ماشین را با چالش مواجه می‌کنند. بنابراین حذف نویز و یا کاهش آن می‌تواند نتایج بهتری را برای این الگوریتم‌ها به همراه داشته باشد. در این مقاله ما پس از دریافت هر فریم و قبل از هر پردازشی نویزهای آن را با استفاده از الگوریتم حذف نویز گاوسی حذف می‌کنیم. نتایج حذف نویز به روش گاوسی برای کرنل‌های مختلف شامل ۳، ۵ و ۹ در شکل ۲ روی فریم ۳۸۵ از ویدئو $seq_{01_cam1_300305_A}$ در [۲۴] نشان داده شده است. در این مقاله ما از کرنل با اندازه ۳ برای شبیه‌سازی استفاده خواهیم کرد زیرا اندازه‌های بیشتر تصویر را بیشتر صیقل داده و تعداد نقاط ویژگی استخراج‌شده را کاهش می‌دهد و اندازه‌های کمتر هم تأثیر نویز را از بین نمی‌برد.

۳-۳ استخراج ویژگی

Sift یک ابزار توصیف‌گر تصویر است که در سال ۱۹۹۹ توسط David Lowe معرفی شد [۱۴]. عملگر Sift نقاطی را در یک تصویر استخراج می‌کند که نسبت به تغییراتی مثل دوران و تغییر مقیاس پایدار است. همچنین نقاط استخراج‌شده نسبت به تبدیلات affine، تغییرات نورپردازی، نویز و تغییر زاویه دید و انسداد تا حدی پایدار هستند. این عملگر در کاربردهای بسیاری مانند تشخیص اشیا، شناسایی چهره، دنبال‌کردن شیء و بازیابی شیء در بانک‌های اطلاعاتی چندرسانه‌ای به طور گسترده‌ای استفاده شده است. در این مقاله نیز ما از این الگوریتم



شکل ۱: دیاگرام روش پیشنهادی برای ردیابی اشیا متحرک.

الگوریتم تفریق پس‌زمینه نواحی متحرک را به دست آورده و در ادامه با الگوریتم مؤلفه‌های متصل^۱ اشیا متحرک را کشف کرده و از مرکز هر شیء به عنوان فاکتور ردیابی استفاده کردند. نویسندگان [۹] و [۱۰] ردیابی عابرین پیاده را با استفاده از اطلاعات رنگ، حرکت و الگوریتم‌های AdaBoost و تفاضل فریم انجام داده‌اند. آنها ابتدا ویژگی‌های مرتبط به اشیا متحرک را استخراج کرده و در هر مرحله به دنبال اشیا با این ویژگی‌های مشخص می‌گردند. در الگوریتم‌های ارائه‌شده برای ردیابی از ویژگی‌های زیادی مانند ویژگی‌های لبه [۱۱] و [۱۲]، نقاط گوشه [۱۳] تا [۱۶]، بافت [۱۷] و [۱۸]، رنگ [۱۹] و [۲۰] و غیره استفاده می‌شود. در [۲۱] یکی از نویسندگان این مقاله به همراه همکاران ردیابی هم‌زمان اشیا متحرک را با استفاده از تخمین تعداد اشیا متحرک در هر فریم و با تناظریابی این نقاط ویژگی مابین دو فریم پی در پی بر اساس ویژگی‌های شدت روشنایی، موقعیت، اندازه و جهت حرکت نقاط ویژگی متناظر انجام دادند.

انتخاب مناسب ویژگی‌ها در ردیابی نقشی کلیدی را بازی می‌کند. از میان ویژگی‌های نام‌برده ویژگی‌های گوشه به طور گسترده‌ای در ردیابی استفاده شده‌اند زیرا نقاط گوشه نسبت به تغییرات شدت روشنایی محیط، تغییر جهت اشیا متحرک و تغییر زاویه شیء مورد ردیابی معطف‌ترند.

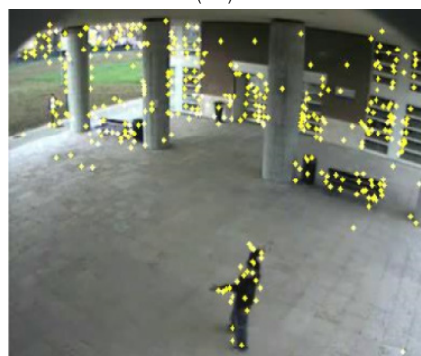
در این مقاله روشی را برای ردیابی اشیا متحرک با ترکیب ۳ الگوریتم استخراج ویژگی Sift، الگوریتم تناظریابی KLT و الگوریتم خوشه‌بندی DBSCAN ارائه خواهیم کرد که از نظر سرعت و دقت دارای نتایج مطلوبی می‌باشد.

۳- روش پیشنهادی

در این مقاله روشی جدید را برای ردیابی هم‌زمان اشیا متحرک ارائه می‌کنیم که نسبت به نوع اشیا متحرک موجود در حوزه دید دوربین،



(الف)



(ب)



(ج)



(د)

شکل ۳: نتایج استخراج نقاط ویژگی Sift روی ۴ فریم از ویدئو seq01_cam1_300305_A در [۲۴]. (الف) فریم ۱، (ب) فریم ۵۰۰، (ج) فریم ۷۲۵ و (د) فریم ۸۳۰.



(الف)



(ب)



(ج)



(د)

شکل ۲: نتایج حذف نویز با روش گاوسی برای کرنل‌های مختلف روی ویدئو seq01_cam1_300305_A در [۲۴]. (الف) فریم ۳۸۵، (ب) حذف نویز با کرنل ۳، (ج) حذف نویز با کرنل ۵ و (د) حذف نویز با کرنل برابر با ۹.

کنتراست^۱ و ماتریس Hessian می‌باشد و توصیف‌گر sift که زمان محاسباتی بالایی دارد روی این نقاط اعمال نشده و تناظریابی نقاط به دست آمده در گام بعدی و با الگوریتم KLT انجام می‌شود.

برای استخراج ویژگی برای تشخیص اشیای متحرک استفاده می‌کنیم. در روش پیشنهادی و در هر مرحله نقاط ویژگی sift روی فریم $t-1$ و به منظور محاسبه اطلاعات حرکت مورد نیاز در گام‌های بعدی نتایج استخراج ویژگی Sift روی فریم‌های ۱، ۵۰۰، ۷۲۵ و ۹۳۰ از ویدئو seq01_cam1_300305_A در [۲۴] در شکل ۳ نشان داده شده است. توجه داشته باشید که در این گام از روش پیشنهادی منظور از استخراج نقاط ویژگی sift برای ردیابی، استخراج نقاط ویژگی بعد از اعمال شرایط

حرکات کوچک^۲ و وابستگی فضایی^۳ می‌باشد. بر طبق این سه فرض شدت روشنایی پیکسل‌های مربوط به یک شیء در فریم‌های متوالی ثابت خواهد بود و پیکسل‌ها در این توالی دارای جابه‌جایی‌های کوچک بوده و همچنین نقاط مربوط به همسایگی یک پیکسل خاص دارای خواص و جابه‌جایی‌های مشابه به همدیگر هستند. در این مقاله از تناظرایی نقاط ویژگی برای استخراج اطلاعات حرکت بین دو فریم پی در پی استفاده می‌شود. نتایج روش KLT بر روی نقاط متناظر برای فریم‌های ۴۰۰، ۷۲۵، ۷۷۰ و ۱۰۲۰ از نمونه مورد آزمایش در گام قبلی در شکل ۴ نشان داده شده است. در شکل ۴ نقاط ویژگی متناظر بین دو فریم متوالی که تناظرشان بین دو فریم پیدا شده باشد را با استفاده از رسم یک خط آبی‌رنگ نشان داده‌ایم (توجه داشته باشید که برای نقاط ویژگی پس‌زمینه به علت جابه‌جایی صفر نقاط متناظر روی هم افتاده‌اند).

۳-۵ کشف نقاط ویژگی مرتبط به اشیای متحرک

پس از تناظرایی نقاط ویژگی Sift بین دو فریم متوالی می‌توان اندازه حرکت (میزان جابه‌جایی) نقاط ویژگی را بر اساس فاصله اقلیدوسی مختصات نقاط برای دو فریم پی در پی از نظر زمانی محاسبه کرد. برای این کار از (۱) استفاده خواهیم کرد [۲۱]. حال از میان نقاط ویژگی متناظر که تناظرشان در بین دو فریم متوالی با الگوریتم KLT به دست آمده باشد نقاطی که جابه‌جایی قابل قبولی داشته باشند (نقاط کلیدی) را به عنوان اشیای متحرک موجود در صحنه ردیابی برچسب‌گذاری کرده و سایر نقاط را از مجموعه نقاط ویژگی حذف می‌کنیم. برای این کار از (۲) استفاده می‌کنیم [۲۱]. نتایج این گام از روش پیشنهادی برای فریم‌های ۶۰۰، ۷۲۵، ۷۸۰ و ۱۰۴۵ از ویدئو seq۰۱_cam۱_۳۰۰۳۰۵_A در شکل ۵ [۲۴] نشان داده شده است

$$m_i = \sqrt{(x_{pi} - x_{ci})^2 + (y_{pi} - y_{ci})^2}, \quad i = 1, \dots, n \quad (1)$$

$$FP = \begin{cases} Object & \text{if } (m_i > \lambda) \\ Background & \text{otherwise} \end{cases}, \quad i = 1, \dots, n \quad (2)$$

در (۱)، n برابر تعداد نقاط ویژگی می‌باشد که تناظرشان در بین دو فریم متوالی به دست آمده است. (x_{ci}, y_{ci}) و (x_{pi}, y_{pi}) به ترتیب مختصات نقاط ویژگی و تناظرشان را در فریم $t-1$ و t نشان می‌دهد. در (۲) FP نقاط ویژگی را نشان می‌دهد که با حد آستانه λ برای میزان جابه‌جایی به دو دسته اشیای متحرک و پس‌زمینه دسته‌بندی می‌شوند. در این مقاله λ را برابر با ۵ پیکسل در نظر گرفته‌ایم.

همان‌طور که از شکل ۵ مشخص است نقاط ویژگی مرتبط به اشیای متحرک با دقت بالایی از نقاط پس‌زمینه جدا می‌شوند.

۳-۶ کشف اشیای متحرک

در این گام از روش پیشنهادی و با استفاده از الگوریتم خوشه‌بندی DBSCAN [۲۶]، نقاط ویژگی مرتبط به اشیای متحرک که در مرحله قبل به دست آمده‌اند را به خوشه‌های مختلف که هر خوشه نماینده یک شیء متحرک می‌باشد تقسیم‌بندی می‌کنیم. هدف از این گام کشف ناحیه‌های کاندید برای اشیای متحرک است تا بتوانیم در مرحله ردیابی و بر اساس معیار شباهت مکان‌های جدید اشیای را کشف کنیم. مبنای الگوریتم DBSCAN بر این اصل استوار است که خوشه‌ها،



(الف)



(ب)



(ج)



(د)

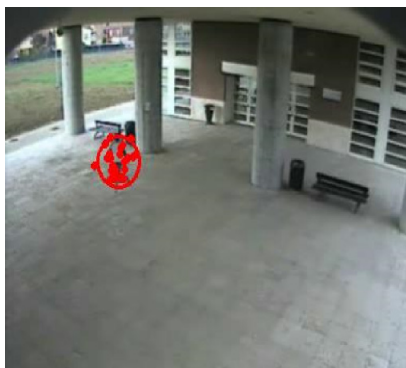
شکل ۴: نتایج تناظرایی نقاط ویژگی Sift روی ۴ فریم از ویدئو seq۰۱_cam۱_۳۰۰۳۰۵_A در [۲۴] با الگوریتم KLT. (الف) فریم ۴۰۰، (ب) فریم ۷۲۵، (ج) فریم ۷۷۰ و (د) فریم ۱۰۲۰.

۳-۴ تناظرایی نقاط با الگوریتم KLT

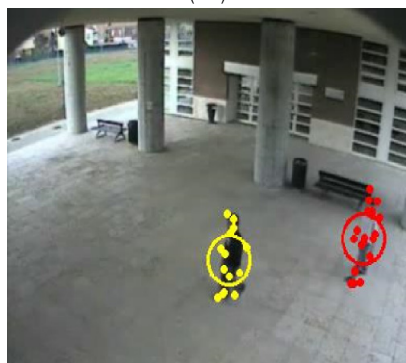
در این گام از روش پیشنهادی ما با استفاده از روش Lucas-Kanade [۲۵] نقاط ویژگی متناظر بین دو فریم در لحظه t و $t+1$ را پیدا می‌کنیم. ایده اصلی روش KLT بر مبنای سه فرض ثابت روشنایی^۱،

2. Temporal Persistence or Small Movements
3. Spatial Coherence

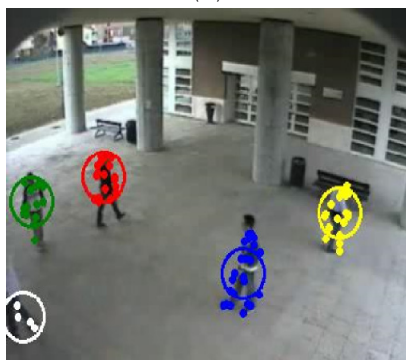
1. Brightness Constancy



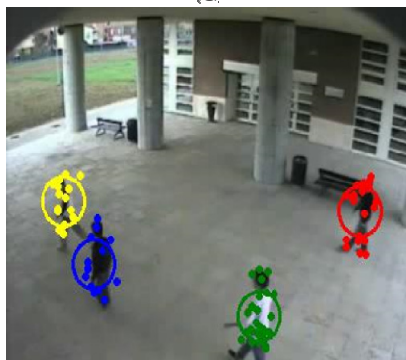
(الف)



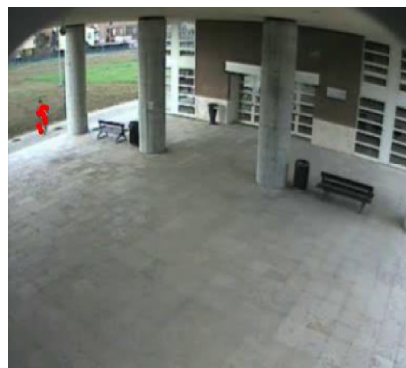
(ب)



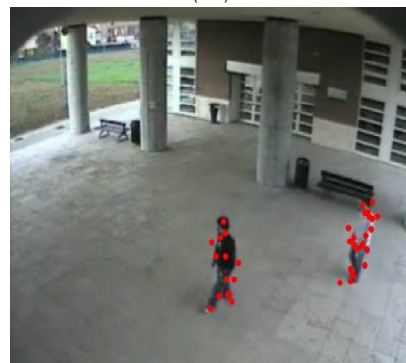
(ج)



(د)



(الف)



(ب)



(ج)



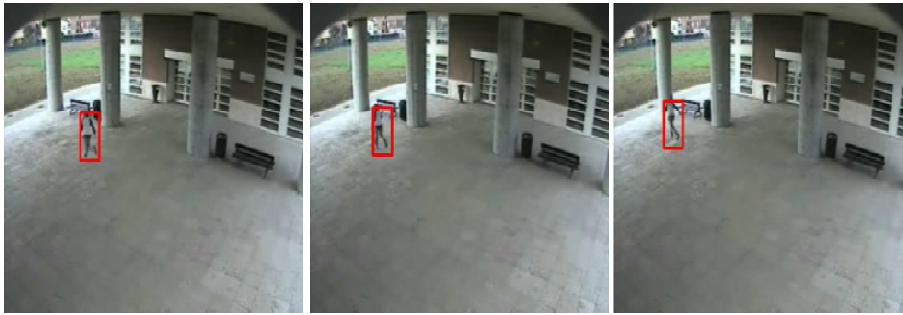
(د)

شکل ۶: نتایج کشف اشیای متحرک روی ۴ فریم از ویدئو seq01_cam1_300305_A در [۲۴] با الگوریتم DBSCAN. (الف) فریم ۳۹۰، (ب) فریم ۷۲۵، (ج) فریم ۷۸۰ و (د) فریم ۱۰۲۲.

شکل ۵: نتایج کشف نقاط ویژگی Sift متعلق به اشیای متحرک روی ۴ فریم از ویدئو seq01_cam1_300305_A در [۲۴] با الگوریتم KLT. (الف) فریم ۶۰۰، (ب) فریم ۷۲۵، (ج) فریم ۷۸۰ و (د) فریم ۱۰۴۵.

۲۵ و حداقل نقاط برای تشکیل یک خوشه ۵ می‌باشد. پس از خوشه‌بندی نقاط ویژگی با الگوریتم DBSCAN، مرکز هر خوشه را با میانگین‌گیری از موقعیت نقاط هر خوشه با استفاده از (۳) به دست آورده و دایره‌ای را به عنوان نماینده شیء به مرکزیت آن رسم می‌کنیم. نتایج خوشه‌بندی نقاط مرتبط به اشیای متحرک برای کشف اشیای متحرک در شکل ۶ نشان داده شده است

ناحیه‌هایی از فضای داده با چگالی زیادی هستند که توسط نواحی با چگالی کمتر از همدیگر جدا شده‌اند. در این روش به منظور تخمین چگالی توزیع نقاط از دو پارامتر شعاع همسایگی (Eps) و حداقل نقاط مورد نیاز برای تشکیل یک خوشه (MinPts) استفاده می‌شود. در این مقاله ما این دو پارامتر را بر اساس آزمایشات زیاد روی ویدئوهای مختلف در شرایط گوناگون به صورت زیر تعیین می‌کنیم. شعاع همسایگی برابر با



(الف)



(ب)

شکل ۷: نتایج ردیابی اشیای متحرک روی فریم‌های مختلف ویدئو A_۳۰۰۳۰۵_cam1_seq۱ در [۲۴] با ترکیب الگوریتم‌های DBSCAN-Sift-klf، (الف) فریم‌های ۳۹۰-۳۹۲-۳۹۴ و (ب) فریم‌های ۷۱۵-۷۲۰-۷۲۵.

کشف‌شده در لحظه‌های $t-1$ و t می‌باشند. در این معادله χ حد آستانه برای محاسبه میانگین شدت روشنایی می‌باشد که در این مقاله برابر ۵ است. علت این که میانگین شدت روشنایی را برای ناحیه دایره‌ای شکل با مرکزیت هر خوشه در نظر گرفتیم این است که مرکز هر شیء در فریم‌های پی در پی تغییرات ناچیزی دارد. به عنوان مثال برای اشیای بندبند مانند انسان مرکز ثقل ثابت بوده و مانند دست‌ها و پاها جابه‌جایی‌های متغیری ندارد.

با توجه به (۴) در صورتی که میانگین شدت روشنایی اشیای کشف‌شده در حوزه دید دوربین در هر تکرار از روش پیشنهادی از حد آستانه تعیین شده بیشتر باشد، این شیء به عنوان یک شیء جدید تلقی می‌شود. لذا در صورت وارد شدن یک شیء جدید به صحنه ردیابی ویژگی‌های این شیء ذخیره شده و در تکرارهای بعدی مورد ردیابی قرار خواهد گرفت. همچنین زمانی که یک شیء از صحنه تحت پوشش دوربین خارج شود از آنجایی که در میان اشیای جدید کشف‌شده در تکرار بعدی قرار ندارد به صورت خودکار و تا زمانی که دوباره وارد صحنه تصویربرداری شود عمل ردیابی روی آن در نظر گرفته نمی‌شود. با این روش در صورتی که اشیای متحرک با شدت روشنایی مختلف به دلایلی مانند قرار گرفتن در سایه، وارد صحنه ردیابی شوند با دقت بالایی مورد ردیابی قرار خواهند گرفت زیرا در هر تکرار معرف هر شیء به روز رسانی می‌شود (از طریق میانگین‌گیری از شدت روشنایی).

۴- نتایج آزمایشات

در این بخش از مقاله ابتدا جزئیات پیاده‌سازی و پارامترهای مورد استفاده در روش پیشنهادی و پایگاه داده‌های مورد آزمایش را معرفی کرده و نتایج روش پیشنهادی را از نظر دقت و زمان پردازش محاسبه کرده و با دیگر روش‌های موجود مقایسه می‌کنیم.

۴-۱ جزئیات پیاده‌سازی و تعیین پارامترها

سیستمی که الگوریتم پیشنهادی روی آن پیاده‌سازی و اجرا شده لپ‌تاپ

$$(x_c, y_c) = \left(\frac{\sum_{i=1}^{n_i} x_i}{n_i}, \frac{\sum_{i=1}^{n_i} y_i}{n_i} \right) \quad (3)$$

در (۳)، (x_c, y_c) به عنوان معرف اشیای مورد ردیابی در نظر گرفته شده و n_i تعداد نقاط موجود در خوشه i ام می‌باشد.

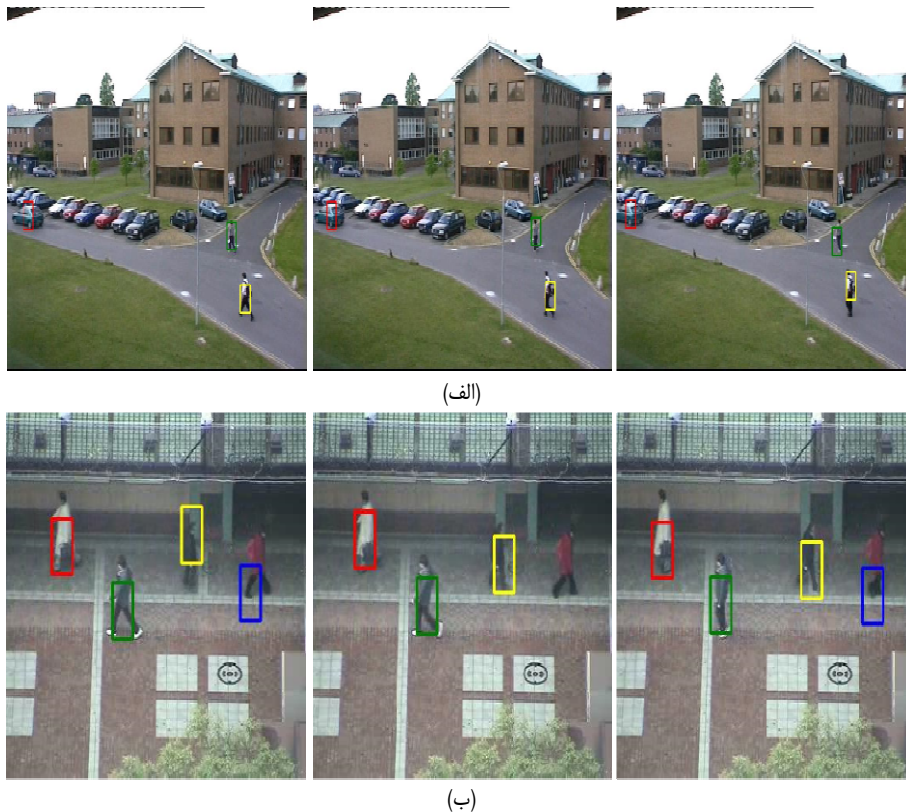
۳-۷ ردیابی

در مرحله ردیابی باید مکان‌های جدید اشیای متحرک موجود در صحنه را در هر فریم دریافتی تعیین کنیم. برای اشیای متحرک کشف‌شده در مرحله قبل میانگین شدت روشنایی نواحی دایره‌ای شکل را با نصف شعاع نمایش داده شده در شکل ۴ به عنوان یک شناسه برای هر شیء تعریف می‌کنیم. در هر تکرار از الگوریتم پیشنهادی و با کشف اشیای متحرک موجود در حوزه دید دوربین مقدار به دست آمده برای میانگین شدت روشنایی هر شیء را با مقادیر موجود در گام قبلی مقایسه کرده و بر اساس یک حد آستانه تناظری یک به یک را بین اشیای متحرک برقرار می‌کنیم. برای این کار از (۴) استفاده خواهیم کرد. در صورتی که اشیای متحرک کشف‌شده بین دو فریم از نظر مقدار میانگین شدت روشنایی ناحیه به دست آمده برای آن شیء شبیه به هم باشند در این حالت مکانی را به عنوان مکان جدید شیء در نظر می‌گیریم که از نظر فاصله اقلیدوسی کمترین مقدار را بین این چند کاندید داشته باشد. نتایج حاصل برای ردیابی در شکل ۷ نشان داده شده است (هدف کلی این بخش تناظریابی اشیای متحرک کشف‌شده بین دو فریم پی در پی است، یعنی اشیای را تطبیق می‌دهیم)

$$C_j = \begin{cases} C_i & \text{if } |V_i - V_j| < \chi \\ NewObject & \text{otherwise} \end{cases} \quad (4)$$

$$i = 1, 2, \dots, N_i, \quad j = 1, 2, \dots, N_j$$

در (۴)، N_i و N_j به ترتیب تعداد خوشه‌ها در فریم قبلی و فریم جاری و V_i و V_j به ترتیب میانگین شدت روشنایی برای اشیای متحرک



(ب)

شکل ۸: نتایج ردیابی اشیای متحرک روی فریم‌های مختلف از ویدئو walking و subway در [۲۲]. (الف) نتایج ردیابی روش پیشنهادی برای انواع مختلف اشیای متحرک روی فریم‌های ۳۳-۳۶ از ویدئو walking و (ب) نتایج ردیابی پیشنهادی برای جهت‌های مختلف حرکت اشیای متحرک روی فریم‌های ۱۱۶-۱۱۹-۱۲۲ از ویدئو subway.

۴-۲ تحلیل روش پیشنهادی

برای تحلیل روش پیشنهادی در مقایسه با دیگر روش‌ها نحوه کار الگوریتم پیشنهادی و دلیل استفاده از الگوریتم‌های مورد استفاده را بیان و با دیگر روش‌ها مقایسه کرده و در بخش‌های بعدی زمان و دقت روش پیشنهادی را با دیگر روش‌ها مقایسه می‌کنیم. در روش پیشنهادی برای استخراج ویژگی از الگوریتم Sift استفاده کردیم زیرا نقاط استخراج‌شده توسط این الگوریتم در برابر تغییرات مقیاس مقاوم بوده و حساسیت کمتری نسبت به نور و تغییرات هندسی سه‌بعدی دارند. ویژگی‌های sift با وجود تغییرات مقیاس، چرخش، نویز، تغییر در شدت روشنایی، پارازیت و انسداد یک تطابق‌یابی قدرتمند را فراهم می‌کند. یکی دیگر از دلایل استفاده از این نقاط ویژگی این است که در مرحله کشف اشیای متحرک در مقایسه با روش‌های [۸] تا [۱۰]، کار را بر روی نقاط ویژگی کمتری انجام داده و توانایی جداسازی اشیای را فراهم می‌کند. در مرحله کشف اشیای متحرک ما از الگوریتم خوشه‌بندی مبتنی بر چگالی DBSCAN استفاده کردیم که بر روی نقاط ویژگی مرتبط به اشیای متحرک انجام می‌شود و این کار باعث زمان مصرفی کمتر برای کشف اشیای متحرک در مقایسه با روش [۸] که از مؤلفه‌های همبند استفاده می‌کند و روش‌های [۸] تا [۱۰] که از هیستوگرام رنگی با زمان مصرفی بالا استفاده می‌کنند. هدف از استفاده از الگوریتم خوشه‌بندی DBSCAN جداسازی اشیای متحرک از هم و در واقع متمایزسازی آنهاست. نتایج روش پیشنهادی روی سه ویدئوی مختلف در شکل ۸ نشان داده شده که نشان می‌دهد الگوریتم پیشنهادی برای ردیابی اشیای متحرک نسبت به نوع اشیای، تغییرات سرعت و جهت حرکت و وجود چندین نوع شیء متحرک به صورت هم‌زمان در حوزه دید دوربین حساسیت ندارد.

نوئو B590 با مقدار حافظه رم ۴ گیگا بایت، کارت گرافیک NVIDIA GEFORCE ۱ گیگابایت و پردازنده اینتل Corei3 ۴ هسته ۲.۵ گیگاهرتزی می‌باشد. برای پیاده‌سازی روش پیشنهادی از زبان برنامه‌نویسی سی‌شارپ و کتابخانه Emgu cv استفاده کرده‌ایم.

در روش پیشنهادی ما از ترکیب سه الگوریتم برای کشف و ردیابی اشیای متحرک استفاده کردیم که هر کدام از این الگوریتم‌ها نیازمند به تعریف پارامترهای خاص خود می‌باشند که به صورت زیر بیان می‌شوند:

- **الگوریتم Sift**: این الگوریتم که برای استخراج ویژگی استفاده شده دارای ۵ پارامتر است. پارامتر اول این الگوریتم مربوط به تعداد نقاط استخراج‌شده توسط الگوریتم است که در این مقاله ما این تعداد را برابر با ۱۵۰۰ در نظر می‌گیریم. پارامتر دوم تعداد اکتاو را نشان می‌دهد که ما مقدار پیش‌فرض یعنی ۳، پارامتر سوم را که حد آستانه Contrast است برابر با ۰.۰۴، پارامتر چهارم را برای حد آستانه لبه‌یابی برابر ۱۰ و آخرین پارامتر را مقدار پیش‌فرض ۱/۶ در نظر می‌گیریم.

- **الگوریتم KLT**: الگوریتم KLT برای یافتن نقاط متناظر بین دو فریم متوالی استفاده می‌شود. این الگوریتم دارای پارامترهای مختلفی مهم از اندازه پنجره جستجو، حداکثر تعداد تکرار برای یافتن نقطه متناظر برای هر نقطه ویژگی و تعداد سطح می‌باشد. در این مقاله اندازه پنجره جستجو برابر با 10×10 ، حداکثر تعداد تکرار برابر با ۱۰ و تعداد سطح برابر با ۹ در نظر گرفته شده است.

- **الگوریتم DBSCAN**: از این الگوریتم برای کشف اشیای متحرک استفاده کرده‌ایم که دو پارامتر دارد. پارامتر اول برای حداقل تعداد نقاط برای تشکیل یک کلاستر است که ما آن را برابر با ۵ در نظر گرفته و پارامتر دوم را برای ماکسیمم فاصله نقاط ویژگی برابر ۲۵ قرار می‌دهیم.

جدول ۱: نتایج مربوط به دقت روش پیشنهادی.

دقت	TN	TF	TT	تعداد فریم	ویدئو [۲۲] تا [۲۴] و [۲۷]
٪۹۵	۱۱	۳	۲۹۱	۲۹۱	coke
٪۹۶	۱۱	۴	۳۲۹	۱۲۰	crossing
٪۹۷	۱۶	۱۳	۹۶۴	۵۵۱	Fight_RunAway۲
٪۹۶	۱۳	۸	۵۱۳	۳۹۵	OneLeaveShopReenter۱
٪۹۵	۲۵۷	۱۳	۵۳۸۴	۲۲۹۲	seq۰۱_cam۱_۳۰۰۳۰۵_A
٪۹۳	۹۲	۷	۱۴۲۹	۱۷۶	subway
٪۹۵	۱۹	۴۸	۱۳۴۵	۱۳۴۵	Sylvester
٪۸۹	۵	۳۵	۳۵۴	۳۵۴	Tiger۱
٪۹۹	۹	۰	۸۷۵	۴۱۲	Walking
٪۹۶	۵۸	۷	۱۷۰۰	۵۰۰	Walking۲
۹۷٪	۱۸	۱۱	۹۴۸	۱۷۹	TUD-Stadtmitte
۹۰٪	۵۳	۲۵	۴۶۱	۷۱	TUD-Campus
۹۵٪	۱۲۲	۹۶	۴۳۶۷	۷۹۴	PETS۰۹-S2L۱
٪۹۵					میانگین دقت

جدول ۳: نتایج مرتبط با زمان اجرای روش پیشنهادی.

زمان پردازش هر فریم	زمان کل	تعداد فریم	ویدئو [۲۲] تا [۲۴] و [۲۷]
۲۹	۸۴۳۹	۲۹۱	coke
۳۱	۳۷۲۰	۱۲۰	crossing
۲۶	۱۴۳۲۶	۵۵۱	Fight_RunAway۲
۳۰	۱۱۸۵۳	۳۹۵	OneLeaveShopReenter۱
۳۱	۷۱۳۶۹	۲۲۹۲	seq۰۱_cam۱_۳۰۰۳۰۵_A
۳۳	۵۸۰۸	۱۷۶	subway
۳۲	۴۳۰۴۷	۱۳۴۵	Sylvester
۲۹	۱۰۲۶۶	۳۵۴	Tiger۱
۳۱	۱۲۷۷۲	۴۱۲	Walking
۳۲	۱۶۰۳۷	۵۰۰	Walking۲
۳۵	۶۳۵۴	۱۷۹	TUD-Stadtmitte
۳۲	۲۲۹۳	۷۱	TUD-Campus
۳۴	۲۶۹۶۳	۷۹۴	PETS۰۹-S2L۱
۳۱/۱ (ms)			میانگین زمان اجرا

در (۵)، TT مجموع تعداد اشیای متحرک موجود در دنباله فریم‌هاست، TN تعداد اشیایی است که روش پیشنهادی آنها را تشخیص نداده و TF تعداد ناحیه‌هایی است که روش پیشنهادی به عنوان شیء متحرک برچسب‌گذاری کرده در حالی که واقعاً شیء متحرک نیستند.

۴- ارزیابی زمان پردازش

برای ارزیابی زمان اجرا و سرعت، روش پیشنهادی را روی ۱۳ نمونه از ویدئوهای مورد آزمایش اجرا کردیم، سپس زمان کل را برای هر ویدئو محاسبه و در نهایت بر تعداد فریم‌های تشکیل‌دهنده همان ویدئو تقسیم کردیم. بدین ترتیب زمان میانگین برای پردازش هر فریم روی هر ویدئو به دست خواهد آمد که در نهایت با میانگین‌گیری از این ۱۳ نمونه، زمان میانگین را برای پردازش هر فریم خواهیم داشت. جدول ۳ نتایج زمان پردازش برای روش پیشنهادی را نشان می‌دهد. از آنجایی که روش پیشنهادی فقط روی نقاط ویژگی مرتبط به اشیای متحرک عمل می‌کند، زمان مصرفی روش پیشنهادی در مقایسه با روش‌های دیگر که به صورت سراسری کل تصویر را در هر مرحله بررسی می‌کنند کمتر می‌باشد. توجه داشته باشید که تعداد اشیای متحرک موجود در حوزه دید دوربین در ۱۳ ویدئوی مورد آزمایش از ۱ تا ۱۰ شیء متغیر است و علت انتخاب این ویدئوها ارزیابی روش پیشنهادی برای حالت‌های مختلف مانند تغییرات در تعداد و نوع اشیای متحرک می‌باشد.

۵- مقایسه روش پیشنهادی با سایر روش‌ها

در این بخش از مقاله به ارزیابی و مقایسه روش پیشنهادی با سایر روش‌ها از نظر زمان و دقت پردازش خواهیم پرداخت. روش پیشنهادی در مقایسه با [۷] سرعت و دقت بالاتری را نشان می‌دهد زیرا نقاط ویژگی sift مورد استفاده در روش پیشنهادی در برابر تغییرات شدت روشنایی پس‌زمینه، تغییر جهت اشیای متحرک، تغییر جهت اشیا و ... مقاوم بوده و استفاده از اطلاعات حرکت نقاط ویژگی در مقایسه با الگوریتم تفاضل فریم در [۷] اطلاعات دقیق‌تری را در اختیار قرار می‌دهد. همچنین روش پیشنهادی در این مقاله در مقایسه با [۲۱] که تعداد اشیا را به صورت اتوماتیک برای الگوریتم k-means تخمین می‌زند وابستگی به داشتن دانش قبلی برای تعداد اشیا ندارد زیرا در گام تشخیص اشیا از الگوریتم

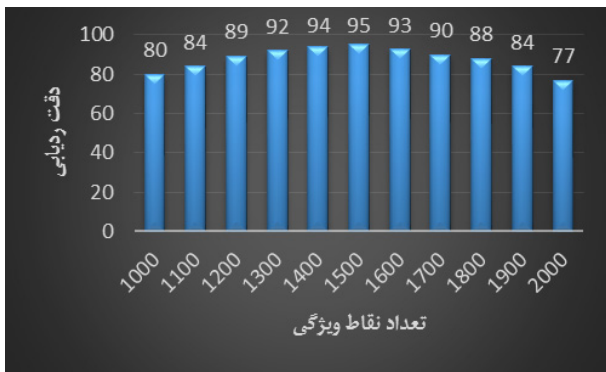
جدول ۲: مقایسه دقت و سرعت روش پیشنهادی با دیگر روش‌ها.

تعداد فریم پردازش شده در ثانیه (FPS)	زمان پردازش هر فریم (ms)	دقت
۳۸	۲۶	٪۹۳/۵ روش ارائه‌شده در [۷]
۲۰	۵۱	٪۷۵ روش ارائه‌شده در [۹]
۲۵	۴۰	٪۸۵ روش ارائه‌شده در [۲۸]
۳	۳۳۳	٪۶۴ روش ارائه‌شده در [۲۹]
۲۰	۵۱	٪۷۵ روش ارائه‌شده در [۳۰]
۳۱	۳۲	٪۸۴ روش ارائه‌شده در [۳۱]
۳۳	۳۱/۱	٪۹۵ روش پیشنهادی

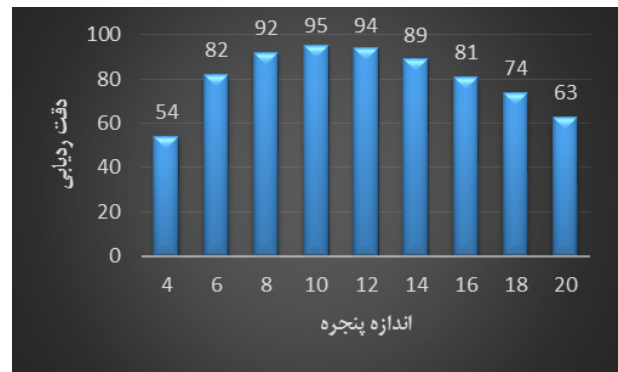
۴-۳ ارزیابی دقت روش پیشنهادی

برای ارزیابی دقت روش پیشنهادی تعداد ۱۳ نمونه از ویدئوهای را که روش‌های مختلف برای ارزیابی دقت استفاده کرده‌اند در نظر گرفته و روش خود را روی آنها اجرا می‌کنیم. در سه مورد از ویدئوهای مورد آزمایش شامل coke، Sylvester و Tiger فقط یک شیء وجود دارد ولی از آنجایی که این ویدئوها دارای اکثر چالش‌های موجود برای ردیابی هستند ما آنها را نیز به نمونه‌های آزمایش اضافه کرده‌ایم تا نتایج روش پیشنهادی با دقت بالاتری تجزیه و تحلیل شود. سپس با توجه به (۵) دقت روش خود را روی هر ویدئو محاسبه کرده و در نهایت از دقت به دست آمده برای آن ۱۳ نمونه میانگین می‌گیریم [۲۱]. نتایج دقت روش پیشنهادی برای این ۱۳ نمونه در جدول ۱ و نتایج ارزیابی دقت روش پیشنهادی در مقایسه با دیگر روش‌ها در جدول ۲ آمده است. علت دقت بالای به دست آمده برای روش پیشنهادی این است که استفاده از نقاط ویژگی Sift باعث می‌شود که روش پیشنهادی در برابر تغییرات شدت روشنایی، تغییرات اندازه شیء و تغییرات زاویه دید شیء مقاوم می‌باشد. همچنین با میانگین‌گیری از شدت روشنایی ناحیه مرکزی شیء در هر گام که تغییرات تقریباً ثابتی دارد نتایج قابل اعتمادتر می‌باشند

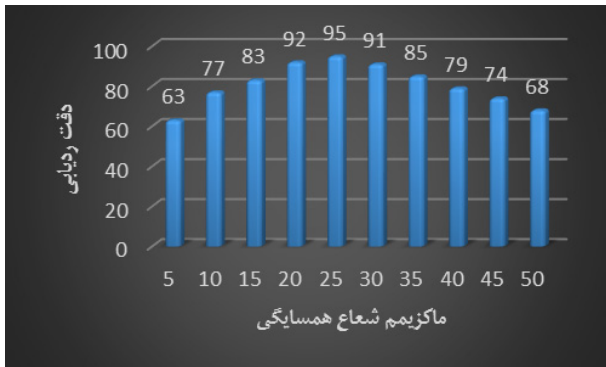
$$Accuracy = \frac{TT - (TF + TN)}{TT} \quad (۵)$$



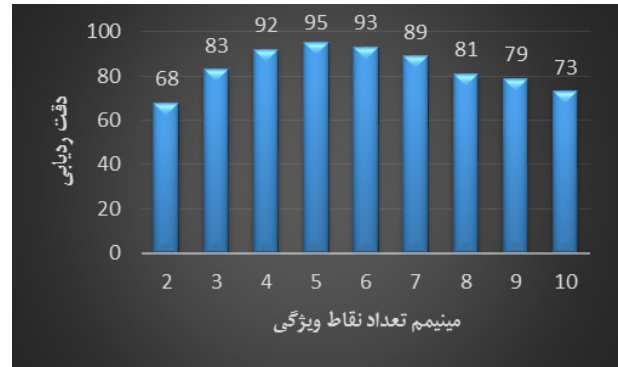
شکل ۱۰: نتایج دقت روش پیشنهادی با تغییر تعداد نقاط ویژگی در الگوریتم sift.



شکل ۹: نتایج دقت روش پیشنهادی با تغییر اندازه پنجره جستجو در الگوریتم KLT.



شکل ۱۲: نتایج دقت روش پیشنهادی با تغییر ماکسیمم شعاع همسایگی در الگوریتم خوشه‌بندی DBSCAN.



شکل ۱۱: نتایج دقت روش پیشنهادی با تغییر مینیمم تعداد نقاط ویژگی در الگوریتم خوشه‌بندی DBSCAN.

مراجع

- [1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: a survey," *Acm Computing Surveys*, vol. 38, no. 4, Article No. 13, 2006.
- [2] J. K. Aggarwal and Q. Cai, "Human motion analysis: a review," *Computer Vision and Image Understanding*, vol. 73, no. 3, Mar 1999.
- [3] I. Haritaoglu, D. Harwood, and L. S. Davis, "W 4: real-time surveillance of people and their activities," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 809-830, Aug 2000.
- [4] I. S. Kim, H. S. Choi, K. M. Yi, J. Y. Choi, and S. G. Kong, "Intelligent visual surveillance-a survey," *International J. of Control, Automation and Systems*, vol. 8, no. 5, pp. 926-939, Sept. 2010.
- [5] M. Piccardi, "Background subtraction techniques: a review," in *Proc. IEEE International Conf. on Systems, Man and Cybernetics*, pp. 3099-3104, Oct. 2004.
- [6] R. Poppe, "Vision-based human motion analysis: an overview," *Computer Vision and Image Understanding*, vol. 108, no. 1, pp. 4-18, Feb. 2007.
- [7] A. Karamiani and N. Farajzadeh, "Detecting and tracking moving objects in video sequences using moving edge features," in *Proc. Scientific Cooperations International Workshops on Electrical and Computer Engineering Subfields*, pp. 88-92, Aug. 2014.
- [8] R. Zhang and J. Ding, "Object tracking and detecting based on adaptive background subtraction," *Procedia Engineering*, vol. 29, no. 1, pp. 1351-1355, Feb. 2012.
- [9] J. Lim and W. Kim, "Detecting and tracking of multiple pedestrians using motion, color information and the AdaBoost algorithm," *Multimedia Tools and Applications*, vol. 65, no. 1, pp. 161-179, Jun. 2013.
- [10] J. S. Lim and W. H. Kim, "Detection and tracking multiple pedestrians from a moving camera," *Advances in Visual Computing*, vol. 3804, pp. 527-534, Dec. 2005.
- [11] K. Bowyer, C. Kranenburg, and S. Dougherty, "Edge detector evaluation using empirical ROC curves." in *Proc. Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 354-359, Jun. 1999.
- [12] J. Canny, "A computational approach to edge detection," *Pattern Analysis and Machine Intelligence, IEEE Trans. on*, vol. 8, no. 6, pp. 679-698, Nov. 1986.

DBSCAN استفاده می‌کند که مبتنی بر چگالی نقاط ویژگی در فضا است. در مقایسه با روش‌های [۲۹] و [۳۱] که به ردیابی انسان می‌پردازند، روش پیشنهادی به نوع اشیای وابسته نبوده و نتایج نشان داده شده در جدول ۲ حاکی از عملکرد بهتر آن می‌باشد.

۴-۶ ارزیابی روش پیشنهادی با تغییر پارامترها

در این بخش از مقاله به ارزیابی روش پیشنهادی با تغییر پارامترهای الگوریتم‌های sift، KLT و DBSCAN خواهیم پرداخت. بدین منظور ما نتایج دقت روش پیشنهادی را برای مقادیر مختلف اندازه پنجره جستجو در الگوریتم KLT در شکل ۹ نشان می‌دهیم. همچنین نتایج دقت روش پیشنهادی را برای مقادیر مختلف تعداد نقاط ویژگی در الگوریتم sift در شکل ۱۰ نشان می‌دهیم. در نهایت نتایج دقت ردیابی برای مینیمم تعداد نقاط و ماکسیمم شعاع همسایگی به ترتیب در شکل‌های ۱۱ و ۱۲ نشان داده شده است.

۵- نتیجه‌گیری

در این مقاله روشی را برای ردیابی اشیای متحرک با ترکیب سه الگوریتم استخراج ویژگی Sift، الگوریتم تناظریابی KLT و الگوریتم خوشه‌بندی DBSCAN ارائه کردیم. در ابتدا با استفاده از دو الگوریتم Sift و KLT و بر اساس اطلاعات حرکت نقاط ویژگی مرتبط به اشیای متحرک را به دست آوردیم و در ادامه این نقاط را با الگوریتم خوشه‌بندی DBSCAN به عنوان اشیای متحرک موجود در صحنه برجسب‌گذاری کرده و در هر فریم دریافتی مکان جدید اشیای متحرک را بر اساس میانگین‌گیری از شدت روشنایی ناحیه دایره‌ای شکل به مرکزیت هر شیء تعیین کردیم. نتایج روش پیشنهادی حاکی از دقت مطلوب روش پیشنهادی برای کشف و ردیابی اشیای متحرک و زمان قابل قبول این روش برای سیستم‌های بینایی ماشین زمان واقعی می‌باشد.

- [26] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. KDD*, pp. 226-231, 1996.
- [27] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler. "MOT16: A benchmark for multi-object tracking," arXiv preprint arXiv:1603.00831, 2016.
- [28] S. He, Q. Yang, R. W. Lau, J. Wang, and M. H. Yang, "Visual tracking via locality sensitive histograms," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 2427-2434, Jun. 2013.
- [29] T. Zhao and R. Nevatia, "Tracking multiple humans in crowded environment," in *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, CVPR*, vol. 2, pp. 406-413, 27 Jun.-2 Jul. 2004.
- [30] J. S. Lim and W. H. Kim, "Detecting and tracking of multiple pedestrians using motion, color information and the AdaBoost algorithm," *Multimed Tools Appl*, vol. 65, no. 1, pp. 161-179, Jun. 2013.
- [31] R. Zhang and J. Ding, "Object tracking and detecting based on adaptive background subtraction," *Procedia Engineering*, vol. 29, pp. 1351-1355, 2012.
- [13] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proc. 4th Alvey Vision Conf.*, pp. 147-152, 31 Aug.-2 Sept. 1988.
- [14] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International J. of Computer Vision*, vol. 60, no. 2, pp. 91-110, Nov. 2004.
- [15] H. P. Moravec, "Visual mapping by a robot rover," in *Proc. of the 6th Int. Joint Conf. on Artificial Intelligence*, vol. 1, pp. 598-600, Aug. 1979.
- [16] J. Shi and C. Tomasi, "Good features to track," in *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, CVPR'94*, vol. 10, pp. 593-600, Jun. 1994.
- [17] L. Nanni, A. Lumini, and S. Brahmam, "Survey on LBP based texture descriptors for image classification," *Expert Systems with Applications*, vol. 39, no. 3, pp. 3634-3641, Feb. 2012.
- [18] J. Ning, L. Zhang, D. Zhang, and C. Wu, "Robust object tracking using joint color-texture histogram," *International J. of Pattern Recognition and Artificial Intelligence*, vol. 23, no. 07, pp. 1245-1263, Nov. 2009.
- [19] G. Paschos, "Perceptually uniform color spaces for color texture analysis: an empirical evaluation," *IEEE Trans. on Image Processing*, vol. 10, no. 6, pp. 932-937, Jun. 2001.
- [20] L. Tao and G. Xu, "Color in machine vision and its application," *Chinese Science Bulletin*, vol. 46, no. 17, pp. 1411-1421, Feb. 2001.
- [۲۱] ع. کرمانی، ن. فرج‌زاده و ح. خانی، "ردیابی دقیق اشیای متحرک بر اساس اطلاعات حرکت و الگوریتم k-means اتوماتیک،" بیستمین کنفرانس ملی سالانه انجمن کامپیوتر ایران، صص. ۲۱۹-۲۱۱، دانشگاه فردوسی مشهد، اسفند ۱۳۹۳.
- [22] K. Chard and K. Bubendorfer, "A distributed economic meta-scheduler for the grid," in *Proc. 8th IEEE Int. Symp. on Cluster Computing and the Grid, CCGRID'08*, , pp. 542-547, Jun. 2008.
- [23] L. Lu and S. Yang, "DIRSS-G: an intelligent resource scheduling system for grid environment based on dynamic pricing," *International J. of Information Technology*, vol. 12, no. 4, pp. 120-127, Jun. 2006.
- [24] D. Abramson, R. Susic, J. Giddy, and B. Hall, "Nimrod: a tool for performing parameterized simulations using distributed workstations," in *Proc. 4th IEEE Symp. on High Performance Distributed Computing*, pp. 112-121, Aug. 1995.
- [25] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. of the 7th Int. Joint Conf. on Artificial Intelligence, IJCAI'81*, vol. 2, pp. 674-679, 24-28 Aug. 1981.

عزیز کرمانی مدرک کارشناسی و کارشناسی ارشد خود در رشته مهندسی فناوری اطلاعات را از دانشگاه شهید مدنی آذربایجان به ترتیب در سال ۱۳۹۱ و ۱۳۹۳ دریافت نمود. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: پردازش تصویر، بینایی ماشین، رایانش توزیعی و داده کاوی می‌باشد.

عسگرعلی بویر تحصیلات خود را در مقطع کارشناسی مهندسی کامپیوتر- نرم‌افزار در سال ۱۳۸۰ اتمام و همچنین مقطع کارشناسی ارشد مهندسی کامپیوتر- نرم‌افزار را نیز در ۱۳۸۵ و از دانشگاه آزاد اراک به پایان رسانده است. سپس به دوره دکترای مهندسی برق و کامپیوتر در دانشگاه صنعتی مالزی (UTM) وارد گردید و در سال ۱۳۹۰ موفق به اخذ درجه دکترا در رشته کامپیوتر از دانشگاه مذکور گردید. و هم‌اکنون استادیار دانشکده فناوری اطلاعات و مهندسی کامپیوتر دانشگاه شهید مدنی آذربایجان می‌باشد. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: محاسبات توزیعی مبتنی بر رایانش ابری/ گرید، داده کاوی و کاربردهای آن در صنعت و پزشکی، شبکه‌های کامپیوتری، پردازش تصویر و شبکه‌های اجتماعی/ پیچیده.