

تشخیص شات برای ویدیو HEVC/H.265

مصطفی رعیتی فرد^۱، دانشجوی دکتری، مهدی محرابی^۲، استادیار، محمد قنبری^۳، استاد

- ۱- گروه مهندسی کامپیوتر - دانشکده فنی مهندسی - واحد شیراز - دانشگاه آزاد اسلامی - شیراز - ایران - mostafa_rayati24@yahoo.com
 ۲- گروه مهندسی کامپیوتر - دانشکده فنی مهندسی - واحد شیراز - دانشگاه آزاد اسلامی - شیراز - ایران - mehrabi@iaushiraz.ac.ir
 ۳- دانشکده مهندسی کامپیوتر و الکترونیک - دانشگاه تهران - تهران - ایران - ghan@essex.ac.uk

چکیده: رشد سریع تعداد و حجم ویدیوها باعث شده که ویدیوها به صورت فایل فشرده ذخیره و ردوبدل شوند. امروزه، نمایه‌گذاری، جستجو و بازیابی فایل فشرده ویدیو از اهمیت بالایی برخوردار است. اولین گام در تحلیل، جستجو و بازیابی ویدیو، قطعه‌بندی زمانی ویدیوها، به مجموعه‌های معنادار و قطعه‌های قابل مدیریت یا همان شات است. در سال‌های اخیر استاندارد جدید فشرده‌سازی ویدیو H.265 به وجود آمده است که نیاز به گسترش و تولید روش‌هایی جهت تحلیل و بازیابی ویدیو در این استاندارد شدیداً احساس می‌گردد. در این مقاله روشی برای قطعه‌بندی زمانی و تشخیص شات در ویدیو فشرده‌شده با استاندارد H.265 بدون فشرده‌سازی کامل با استفاده از نوع خاصی از اطلاعات کدگذاری یعنی پرچم‌های تبدیل که تا کنون استفاده نشده ارائه شده است. در این روش مقادیر ضرایب غیر صفر پرچم‌های ماکرو بلوک‌های تبدیل قاب‌ها، که در سراینده رشته بیت فشرده وجود دارد محاسبه و در نهایت با یک آستانه مناسب، مرز شات تشخیص داده می‌شود. آزمایش‌ها نشان می‌دهد روش ارائه‌شده با دقت ۸۰٪ و فراخوانی ۸۴/۳٪ بدون فشرده‌سازی ویدیو، شات‌ها را تشخیص دهد.

واژه‌های کلیدی: استاندارد ویدیویی HEVC/H.265، حوزه فشرده، تشخیص شات، قطعه‌بندی ویدیو.

Shot detection in HEVC/H.265 video

Mostafa Rayati Fard¹, PhD Student, Mahdi Mehrabi², Assistant Professor, and Mohammad Ghanbari³, Professor

1- Department of Computer Engineering, Shiraz Branch, Islamic Azad University, Shiraz, Iran,
 Email: Mostafa_rayati24@yahoo.com

2- Department of Computer Engineering, Shiraz Branch, Islamic Azad University, Shiraz, Iran, Email: mehrabi@iaushiraz.ac.ir

3- School of Electrical and Computer Engineering, University of Tehran, Tehran, Iran, Email: ghan@essex.ac.uk

Abstract: The huge number and volume of video and video usage have caused most of them saved and transferred as compressed video. Nowadays, indexing, searching and retrieving video directly in compressed domain has been taken great attention. The first step in video indexing and retrieval is segmenting a video into chronological sets and manageable pieces or shots. In recent years, a new video coding standard say H.265 has been introduced and it is needed to develop the methods and algorithms for analyzing, indexing and retrieving H.265 compressed video. In this paper, a novel method has been proposed for shot detection in H.265 compressed video without full decompression. In the proposed method, macro block coding information which is in headers of compressed H.265 bit-stream by a threshold is used for shot detection of compressed video. The results indicate that the proposed method detect video shots in H.265 video by 80% accuracy and 84/3% recall.

Keywords: HEVC/H.265, compressed domain, shot detection, video segmentation.

۱- مقدمه

تغییر در مقادیر تشابه و عدم تشابه ظاهر خواهد شد [۱۹]. پس هر شات با استخراج قاب‌های کلیدی و استخراج ویژگی‌های مکانی و زمانی نشان داده می‌شود. یکی از راه‌های پایه‌ای تشخیص شات مقایسه پیکسل‌های دو قاب متوالی پشت سر هم است. این مقایسه به صورت پیکسل یا آمارهای دیگر آن مثل هیستوگرام، میانگین و ... صورت می‌گیرد، سپس بازبایی بر پایه شباهت بین ویژگی تصویر و ویژگی نمونه‌های ذخیره‌شده در پایگاه داده انجام می‌شود. رویکرد تشخیص مرز شات می‌تواند به‌طور گسترده‌ای به دو دسته: دامنه فشرده و غیر فشرده تقسیم شود.

به‌طور پایه تجزیه و تحلیل، قطعه‌بندی و تشخیص شات ویدیو بر روی پیکسل‌ها در حوزه غیر فشرده انجام می‌شود. در [۴] به‌مرور تکنیک‌های قطعه‌بندی یا شاخص‌گذاری ویدیویی در حوزه غیر فشرده پرداخته شده است، اما در سال‌های اخیر استانداردهای فشرده‌سازی ویدیو به دلیل حجم زیاد ویدیوها و همچنین ترافیک ارسال داده‌های ویدیویی بر روی شبکه‌ها گسترش سریع‌تری داشته و انگیزه استفاده از این استانداردها بیشتر شده است.

امروزه اکثر ویدیوها به‌صورت فشرده ذخیره یا ردوبدل می‌شوند لذا روش‌های بازبایی در حوزه فشرده به وجود آمده است [۲۱-۲۳]. یکی از مهم‌ترین استانداردهای دهه اخیر در زمینه فشرده‌سازی ویدیو استاندارد خانواده H.26x است که H.265 به‌عنوان یک عضو جدید از این خانواده پرکاربرد در زمینه فشرده‌سازی ویدیو مورد توجه قرار گرفته است [۲۰].

استاندارد فشرده‌سازی ویدیو H.265 از دو قسمت کدگذار و کدگشا تشکیل شده است. فایل ویدیو برای فشرده شدن به کدگذار این استاندارد داده می‌شود و با یکسری از پارامترها شامل، گام کوانتایز و چگونگی قرار گرفتن انواع قاب‌ها و ... به یک فایل فشرده با حجم حدوداً یک هزارم فایل اصلی ویدیو تبدیل می‌گردد. پس این عمل باعث می‌شود این فایل فشرده شده ویدیو با حجم خیلی کمتر نسبت به ویدیو اصلی ذخیره و با سرعت بیشتری در شبکه منتقل گردد. در این مقاله روشی ارائه شده است که از روی فایل فشرده شده ویدیو و بدن نیاز به فایل اصلی ویدیو، می‌توان ویدیوها را به قطعه‌های قابل مدیریت تقسیم و شات را تشخیص داد. یکی از مزیت‌های روش ارائه شده این است که در این روش مستقیماً از روی فایل فشرده‌شده (بدون غیر فشرده‌سازی) به اطلاعات ویدیو جهت تشخیص شات دسترسی پیدا می‌کند.

با توجه به اینکه قاب‌های نوع I نسبت به قاب‌های نوع P و B دارای حجم بیشتری از اطلاعات هستند و گاهی بر اساس شرایط و برای کاربردهای بلادرنگ نیاز به کم شدن حجم فایل فشرده ویدیو با کم کردن تعداد قاب‌های نوع I در کدگذار است، در روش ارائه‌شده از اطلاعات موجود در سراینده قاب نوع P فایل فشرده‌شده ویدیو برای تشخیص شات استفاده می‌شود.

مهمترین نوآوری روش پیشنهادی ما، استفاده از اطلاعات موجود در پرچم‌های شاخص^۲ قاب نوع P است، این اطلاعات در رشته بیت

رشد روزافزون اینترنت، شبکه‌های اجتماعی اشتراک‌گذاری ویدیو، تصاویر و دستگاه‌های همراه با قابلیت تصویربرداری مناسب، باعث شده تصویر و ویدیو حجم زیادی از داده‌های ذخیره‌شده و انتقالی را در دستگاه‌ها و شبکه‌ها تشکیل دهد [۱، ۲]. بر اساس آمار منتشر شده، مصرف و تولید ویدیو سالیانه با نرخ ۳۰٪ در حال افزایش است [۳]. حجم زیاد این اطلاعات باعث شده جستجو و بازبایی در این حجم عظیم از اطلاعات دیداری (ویدیو) به یک ضرورت تبدیل شود [۴، ۵]. این امر موجب ایجاد و بهبود تکنیک‌هایی برای تجزیه و تحلیل، جستجو و بازبایی این نوع اطلاعات شده است [۶] و با توجه به اینکه ویدیوها از لحاظ اطلاعاتی بسیار غنی هستند [۷]، پس قابلیت آن را دارند که به دسته‌های بسیار دقیق و جزئی طبقه‌بندی شوند، بر این اساس روش‌های تجزیه و تحلیل ویدیو مبتنی بر محتوا گسترش یافته است [۸].

در روش تجزیه و تحلیل ویدیو مبتنی بر محتوا، بر اساس ویژگی‌هایی که به‌صورت خودکار از ویدیوها استخراج می‌گردد، ویدیوها نمایه‌گذاری شده و مورد تجزیه و تحلیل، جستجو و بازبایی قرار می‌گیرند، این روش‌ها به‌طور گسترده‌ای توسعه یافته‌اند، به‌طوری‌که در تجزیه و تحلیل رویدادهای خبری، مدیریت هوشمند ویدیوها در سایت، نظارت تصویری [۹]، پنهان‌سازی خطای ویدیو [۱۰]، استخراج قاب‌های کلیدی [۱۱]، قسمت‌بندی رویداد ویدیویی [۱۲، ۱۳] از روش‌های تجزیه و تحلیل ویدیو مبتنی بر محتوا استفاده می‌گردد.

اولین گام در تجزیه و تحلیل ویدیو مبتنی بر محتوا، قطعه‌بندی زمانی ویدیوها است که هدف آن تقسیم کردن رشته‌های ویدیو به مجموعه‌های معنادار و قطعه‌های قابل مدیریتی مانند شات یا صحنه است. یک شات ویدیو، گروهی از قاب‌های پیوسته است که اطلاعات یکسان و ویژگی‌های دیداری شبیه به هم، از قبیل رنگ، حرکات و الگوها را دارند. انتقال حالت بین شات‌ها، دو نوع ناگهانی و تدریجی دارد. انتقال ناگهانی یا سخت^۱ زمانی رخ می‌دهد که، دو قاب متوالی به‌طور مستقیم و بدون هیچ‌گونه ویرایش (جلوه‌های ویژه) پیوند خورده باشند. در این انتقال، شات بعدی بلافاصله پس از آخرین قاب شات قبلی ظاهر می‌شود، بنابراین می‌توانیم نتیجه بگیریم که انتقال ناگهانی بین آخرین قاب یک شات و قاب اول شات بعدی اتفاق می‌افتد. در مقابل، انتقال تدریجی یا نرم^۲ زمانی اتفاق می‌افتد که دو قاب با استفاده از جلوه‌های ویژه در طول دوره تولید تلفیق می‌شوند. در این انتقال قاب‌های همسایه در دو شات متوالی به آرامی تغییر می‌کند، نظیر تداخل تصویر، محو شدن، تار شدن و دیگر اثرات تدریجی [۴]. پس شات به‌عنوان پایه‌ای‌ترین عنصر برای شاخص‌گذاری مورد استفاده قرار می‌گیرد [۱۶-۱۴].

هدف از تشخیص مرز شات این است که ویدیو به واحدهای اساسی تقسیم شود [۱۷، ۱۸]. در یک ویدیو، قاب‌ها با توجه به محتوای بصری بسیار شبیه به هم هستند. هنگامی که انتقال شات اتفاق می‌افتد،

در حوزه ویژگی‌های مکانی و حرکتی، از ضرایب تبدیل گسسته کسینوسی تصاویر کدگذاری شده به‌طور گسترده‌ای برای تولید بردارهای ویژگی از ویدیوهای فشرده می‌توان استفاده کرد. در [۲۶] روشی بر اساس قطعه‌بندی زمانی ویدئو مبتنی بر ضرایب تبدیل گسسته کسینوسی، نوع کدگذاری و بردارهای حرکت بر روی داده‌های ویدیویی فشرده شده با استاندارد MPEG ارائه شده است، در این روش برای تشخیص شات ناگهانی از مقایسه تعداد بردارهای حرکت در ناحیه‌های انتخاب شده، استفاده می‌شود. این الگوریتم به علت استراتژی چند مرحله‌ای دارای سرعت پردازش بالا و دقت خوبی است، اما مشکلی که در این روش وجود دارد این است که چگونگی حرکت اشیاء در شات تدریجی تشخیص داده نمی‌شود. این مشکل با ارائه روشی در [۲۷] که مبتنی بر ضرایب DC و نوع کدگذاری ماکرو بلوک است، کمتر خودنمایی می‌کند (ضرایب DC با استفاده از روش ارائه شده در [۳۴] تولید می‌شوند). در این روش با محاسبه واریانس و مربع ضرایب DC برای قاب‌های فشرده شده با استاندارد MPEG-2/MPEG و مشخص کردن شکل‌های سهمی در نمودار، تغییرات تدریجی تشخیص داده می‌شود. این روش نسبتاً ساده و دارای دقت خوبی است، مجموع پارامترهای موردنیاز برای این الگوریتم ۷ پارامتر است.

یکی از عوامل مهم در تشخیص شات، تشخیص محل دقیق شات‌ها است که در [۲۸] با ارائه یک روش ترکیبی بر اساس قطعه‌بندی زمانی ویدئو مبتنی بر نوع کدگذاری ماکرو بلوک و نرخ بیت توانستند محل دقیق شات‌های ناگهانی را در استاندارد MPEG تشخیص دهند. در این روش فقط از اطلاعات نرخ بیت در سطح ماکرو بلوک و تعداد حرکات تخمینی ماکرو بلوک‌ها استفاده می‌شود. این الگوریتم به‌طور سلسله‌مراتبی شات ناگهانی را بین دو قاب تشخیص می‌داد، اما در تشخیص شات‌های تدریجی دچار مشکل است.

برای تشخیص شات‌های تدریجی در استاندارد MPEG از روش قطعه‌بندی زمانی ویدئو مبتنی بر نوع کدگذاری ماکرو بلوک و بردارهای حرکت که در [۲۹] معرفی شده، استفاده شده است. در این روش یک طرح دو مرحله‌ای بر اساس شبکه‌های عصبی دنبال می‌شود و پردازش دقیقاً از روی قاب‌های همسایه انجام می‌گیرد. در روش ارائه شده مرزهای ساده مانند شات ناگهانی با مدل ریاضی و مرزهای پیچیده مانند شات تدریجی با استفاده از شبکه‌های عصبی تشخیص داده می‌شوند، برای تشخیص شات ناگهانی از تعداد ماکرو بلوک‌های رو به عقب و روبه‌جلو، و برای تشخیص شات تدریجی از تعداد درونی‌هایی و ماکرو بلوک‌های رو به عقب استفاده می‌شود، این روش دارای دقت بالا، در پردازش و تشخیص شات است.

علاوه بر روش‌های مطرح شده، در [۳۰] روشی برای تشخیص شات‌های ناگهانی و نیز دسته‌بندی انواع شات‌های تدریجی، در استاندارد فشرده‌سازی MPEG ارائه شده است. در این روش، شدت لبه‌ها بین قاب‌های متوالی بررسی می‌گردد، قابل ذکر است که در یک شات ناگهانی، شدت لبه‌ها بیشتر از شدت لبه‌های قدیمی است، به

فشرده شده ویدئو قرار می‌گیرند و تا کنون در هیچ روشی استفاده نگردیده است. اطلاعات این پرچم‌ها مستقیماً از روی فایل فشرده شده استخراج می‌گردد. یکی از مزیت‌های روش ارائه‌شده در این مقاله سرعت بالا و پیچیدگی کم در تشخیص شات است، همچنین روش ارائه‌شده توانایی شناسایی محل دقیق شات را نیز دارد. این مزیت باعث می‌شود بتوان فایل فشرده ویدئو را بدون غیر فشرده کردن با استفاده از روش ارائه‌شده در این مقاله، خلاصه‌سازی کرد.

در این روش برای تجزیه و تحلیل فایل فشرده ویدئو و تشخیص شات از شناسایی ضرایب غیر صفر بلوک‌های تبدیل^۴ استفاده می‌شود. در روش پیشنهادی، از یک پرچم در یک مجموعه ساختار سلسله‌مراتبی که وظیفه آن کاهش تعداد ضرایبی است که باید منتقل شود، استفاده شده است. این پرچم بدون محاسبات اضافی و به‌طور مستقیم از ترکیب ماکرو بلوک به دست می‌آید و زمان محاسبه را به‌شدت کاهش می‌دهد. نتایج شبیه‌سازی و آزمایش‌ها نشان می‌دهد در روش ارائه شده، علاوه بر اینکه نیاز به غیر فشرده کردن ویدئو نیست بلکه دقت بسیار بالایی برای تشخیص و تعیین مکان دقیق شات‌ها وجود دارد. این روش یک ابزار قدرتمند برای تجزیه و تحلیل اطلاعات بصری در طیف گسترده‌ای از برنامه‌های کاربردی است.

در بخش بعدی این مقاله به‌مرور خلاصه‌ای از تحقیقات انجام شده برای بازیابی و تشخیص شات در حوزه فشرده پرداخته می‌شود، سپس در بخش سوم مقاله ساختار استاندارد H.265 شرح داده می‌شود، روش پیشنهادی برای پارامترهای تصویر و کدگذاری در بخش چهارم مورد مطالعه قرار گرفته است. در بخش پنجم عملکرد روش پیشنهادی برای انواع مختلف مجموعه پارامترهای کدگذاری ارزیابی می‌شود، در بخش ششم روش ارائه شده در این مقاله را با روش‌های دیگر مقایسه می‌کنیم و در نهایت نتیجه‌گیری در بخش هفتم مطرح شده است.

۲- پیشینه

در حوزه فشرده، اطلاعات کدگذاری به‌صورت مستقیم یا با پردازش، جهت نمایه‌گذاری ویدئو و تشخیص شات استفاده می‌گردد. مقالات [۲۱-۲۳] به مرور انواع ویژگی‌های مستخرج از اطلاعات کدگذاری از جمله ویژگی‌های مکانی، حرکتی (برداری حرکت) و کدگذاری ویدئو، برای تجزیه و تحلیل و تشخیص شات در ویدئو می‌پردازد. مهم‌ترین عناصر و اطلاعات کدگذاری که جهت نمایه‌گذاری و تشخیص شات استفاده می‌شوند عبارت‌اند از: نرخ بیت [۲۴]، هیستوگرام رنگ و بافت با استفاده از ضرایب تبدیل گسسته کسینوسی^۵ [۲۵] و نوع کدگذاری ماکرو بلوک و بردارهای حرکت [۲۶]، ضرایب DC و نوع کدگذاری ماکرو بلوک و بردارهای حرکت [۲۷]، نوع کدگذاری ماکرو بلوک و نرخ بیت اطلاعات [۲۸]، نوع کدگذاری ماکرو بلوک و بردارهای حرکت [۲۹]، شدت لبه [۳۰] و بردار حرکتی و جهت آن [۳۱-۳۳].

خود نگه می‌دارند پس این روش برای استریم ویدیوهای که به خاطر کمتر شدن حجم فایل فشرده، از تعداد قاب نوع I کمتری استفاده می‌کنند اصلاً مناسب نیست.

برای حل این مشکل ما در این مقاله یک الگوریتم قوی را در حوزه فشرده‌شده استاندارد H.265 ارائه داده‌ایم که از اطلاعات موجود در سراینده فریم نوع P برای تشخیص شات استفاده می‌کند. نتایج آزمایش‌ها نشان می‌دهد روش پیشنهادی توانسته است به خوبی شات‌ها را در ویدیو فشرده‌شده با استاندارد H.265 تشخیص دهد، ضمناً مقاومت خوبی نسبت به تغییرات شرایط کدگذاری دارد.

۳- ساختار استاندارد H.265

کدگذاری ویدیویی با کارایی بالا (H.265) جدیدترین استاندارد کدگذاری ویدیویی است که در سال ۲۰۱۰ توسط ITU-T و ISO / IEC ایجاد گردید و در سال ۲۰۱۳ اولین نسخه آن نهایی شد [۳۷]. این استاندارد با هدف بهبود ۵۰٪ نرخ فشرده‌سازی بهتر نسبت به استاندارد قبلی H.264/AVC ولی با پیچیدگی بیشتر وارد بازار شد [۳۸، ۳۹]. استاندارد H.265 تعدادی از ویژگی‌های سطح بالای خود را از قبیل واحد لایه انتزاعی شبکه^۷ و مجموعه‌ای از پارامترها، از جمله، پارامترهای تنظیم تصویر و پارامترهای تنظیم رشته ویدیویی را از H.264 به ارث برده است، علاوه بر این، H.265 یک نوع جدید از مجموعه پارامترها به نام مجموعه پارامترهای ویدیویی^۸ را نیز معرفی می‌کند [۴۲].

۳-۱- کدگذاری در استاندارد H.265

در استاندارد H.265، در ابتدا قاب‌های ورودی بر اساس ساختاری به نام درخت چهارگوش^۹ به ماکرو بلوک‌های یکسان 64×64 تقسیم می‌شوند که به آن واحد درخت کدگذاری^{۱۰} می‌گویند، واحد درخت کدگذاری، مشابه مفهوم ماکرو بلوک در استاندارد H.264 است. این امکان در استاندارد وجود دارد تا در صورت لزوم هر ماکرو بلوک به قسمت‌های کوچک‌تر تقسیم شود. با توجه به اینکه این استاندارد از فضای رنگی YCbCr استفاده می‌کند پس هر واحد درخت کدگذاری شامل یک بلوک درخت کدگذاری^{۱۱} برای فضای روشنایی^{۱۲} به همراه دو بلوک درخت کدگذاری متناظرش برای فضای رنگ‌دانه^{۱۳} است. استاندارد H.265 ساختار تقسیم‌بندی بلوکی مناسب، با انعطاف‌پذیری بالایی را توسط چهار مفهوم مختلف واحد درخت کدگذاری، واحد کدگذاری^{۱۴}، واحد پیش‌بینی^{۱۵} و واحد تبدیل به وجود آورده است. هر یک از واحدهای کدگذاری، تبدیل و پیش‌بینی به ترتیب (همانند واحد درخت کدگذاری که از یک بلوک درخت کدگذاری نمونه روشنایی به همراه دو بلوک درخت کدگذاری متناظرش از نمونه رنگ‌دانه، تشکیل شده است) از سه بلوک کدگذاری^{۱۶}، سه بلوک پیش‌بینی^{۱۷} و سه بلوک تبدیل برای نمایش هر یک از فضاها رنگی YCbCr که به واحد

همین ترتیب، لبه‌های قدیمی از محل لبه‌های جدید محو می‌شوند. بنابراین، با شمارش پیکسل‌های وارد شده و خارج شده لبه می‌توان شات ناگهانی و تدریجی را تشخیص و دسته‌بندی کرد.

در حوزه ویژگی‌های حرکتی که در قبل نیز به آن اشاره شده است، در [۳۱] روشی برای نمایه‌گذاری ویدیو و تشخیص شات بر اساس بردارهای حرکت ارائه شد. در این روش یک هیستوگرام از موقعیت بلوک‌ها که در جبران حرکت استفاده می‌شود تشکیل و نشان داده شده است، هیستوگرام موقعیت بلوک کارایی بالایی در نمایه‌گذاری ویدیو و تشخیص شات دارد.

در ادامه ویژگی‌های حرکتی، در [۳۲] یک روش برای طبقه‌بندی شات‌ها در ویدیو فشرده‌شده با استاندارد H.264/AVC ارائه شد. در این روش، اطلاعات جهت، از بردارهای حرکت استخراج و سپس یک هیستوگرام از جهت بردارهای حرکت برای قاب‌ها در طول زمان تشکیل می‌شود. این هیستوگرام‌ها برای آنچه مکعب‌های مکانی، زمانی نامیده شده، تشکیل می‌شود و از آنجا که تعداد این مکعب‌ها زیاد است، با استفاده از روش طبقه‌بندی K-means به طبقه‌بندی قاب‌ها و تشخیص شات پرداخته می‌شود. در [۳۵] نیز یک توصیف‌گر هیستوگرام حرکت دوربین برای تشخیص شات معرفی شده است. در این روش اطلاعات خام حرکت قاب‌های متوالی به وسیله محاسبه بردار حرکت هر ماکروبلوک استخراج می‌گردد تا یک توصیف‌گر به نام فیلدهای بردار حرکت ایجاد شود، این توصیف‌گر برای نه ناحیه 3×3 در هر ماکرو بلوک محاسبه و به مقادیر یکتا تجزیه می‌شود، سپس با استفاده از ماشین بردار پشتیبان قاب‌ها طبقه‌بندی، و شات‌ها تشخیص داده می‌شوند.

در حوزه ویژگی‌های کدگذاری، در [۱۵] با استفاده از هیستوگرام ساده شده HOG شات در قسمت کدگذار استاندارد H.265 تشخیص داده می‌شود. در این روش قبل از کدگذاری واقعی، شیب پیکسل‌ها برای هر دو جهت عمودی و افقی با استفاده از شیب اختلاف مرکزی قاب‌ها محاسبه می‌شوند، سپس با مقایسه واریانس‌های قاب فعلی و مقادیر میانگین واریانس دو قاب متوالی قبلی که به صورت داخلی کدگذاری می‌شوند، مرز شات تشخیص داده می‌شود.

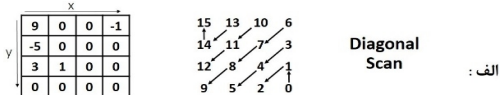
در [۲۲] نیز یک روش تشخیص شات در دامنه فشرده‌شده استاندارد کدگذاری ویدیویی H.265 پیشنهاد شده است. این روش مبتنی بر آنالیز توزیع واحد تبدیل^{۱۶} در استاندارد H.265 است، روش پیشنهادی شامل سه مرحله است: در مرحله اول، اندازه واحد تبدیل به دست می‌آید. در مرحله دوم، تعداد هراندازه واحد تبدیل محاسبه می‌شود و هیستوگرام مربوطه ترسیم می‌شود. در مرحله آخر، دو هیستوگرام مرتبط باهم مقایسه می‌شوند. نتایج تجربی نشان می‌دهد که الگوریتم پیشنهادی با یک پیچیدگی محاسباتی کم عملکرد خوبی را کسب می‌کند. اما مشکلی که در این روش وجود دارد این است که در این روش تمرکز بر روی فریم I است و با توجه به اینکه قاب‌های نوع I نسبت به قاب‌های نوع P,B حجم خیلی بیشتری از اطلاعات را در

پیش‌بینی از روی قاب‌ها یا بلوک‌های قبلی که کدگشایی شده است، استخراج می‌شود. بلوک پیش‌بینی با بلوک مانده ترکیب می‌شوند تا بلوک‌های کدگذاری به دست آید و در نهایت از ترکیب بلوک‌های کدگذاری واحد درخت کدگذار، کل قاب کدگشایی می‌شود.

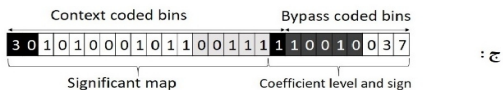
۴- روش پیشنهادی

از آنجا که ضرایب تبدیل گسسته کسینوسی در حجم نهایی رشته بیت تأثیر زیادی دارند و همچنین از بین رفتن آن‌ها نویز زیادی به ویدئو غیر فشرده‌شده تحمیل می‌کند و از طرفی یکی از زمان‌برترین قسمت‌های کدگشایی کامل و رسیدن به قاب‌ها، عکس تبدیل گسسته کسینوسی است، اطلاعاتی به صورت پرچم در سرآیندهای رشته بیت ویدئو فشرده‌شده با استاندارد H.265 ایجاد می‌شود تا غیر فشرده‌سازی، سریع‌تر و راحت‌تر انجام گیرد.

در این مقاله از این اطلاعات که در سرآیند بلوک‌های تبدیل وجود دارد استفاده می‌کنیم، تا بدون کدگشایی رشته بیت، خصوصاً انجام عکس تبدیل گسسته کسینوسی، محتوای قاب‌های نوع P را برای بررسی تغییرات و تشخیص صحنه یا شات تحلیل کنیم. اطلاعات استفاده‌شده در مجموعه‌ای از پرچم‌ها که به آن‌ها پرچم‌های شاخص می‌گویند، قرار می‌گیرند و از آنجا که بخش بزرگی از ضرایب بلوک تبدیل پس از چندی سازی به صفر نگاشت می‌شوند این پرچم‌ها تعریف شده‌اند تا تعداد ضرایبی که باید منتقل شوند کاهش یابند. در شکل ۱ نمونه‌ای از این پرچم‌ها و نحوه مقداردهی آن‌ها نشان داده شده است.



last_Significant_coeff_x	3									
last_Significant_coeff_y	0									
Significant_coeff_flag	1	0	1	0	0	0	1	0	1	1
Coeff_abs_level_greater1_flag	0	0				1	1	1		
Coeff_abs_level_greater2_flag	1									
Coeff_abs_level_remaining						0	3	7		
Coeff_sign_flag	1	0				0	1	0		



شکل ۱: ساختار سلسله‌مراتبی از پرچم‌های شاخص در H.265 [۴۰]

الف: نحوه پیمایش بلوک تبدیل - ب: نحوه مقداردهی به پرچم‌ها بر

اساس پیمایش انجام شده - ج: نقشه اهمیت بلوک تبدیل

شکل ۱ از سه قسمت تشکیل شده است. در قسمت الف این شکل یک بلوک تبدیل و نحوه پیمایش آن نشان داده شده است، در قسمت ب پرچم‌هایی هستند که بر اساس پیمایشی که در قسمت الف انجام شده مقداردهی می‌شوند و در پایین‌ترین سطح سلسله‌مراتب، برای هر بلوک پیمایش شده، همان‌طور که در قسمت ج نشان داده شده است، یک نقشه به اصطلاح معنی‌دار، یا نقشه اهمیت که حاوی اطلاعاتی از جمله، مکان و مقدار هر موقعیت یا ضرایب قابل توجه (غیر صفر) بلوک

کدگذاری، واحد پیش‌بینی و واحد تبدیل اختصاص یافته، استفاده می‌کنند [۴۲-۴۳].

هر واحد درخت کدگذاری به بلوک‌های کوچک‌تری تقسیم می‌شود که به آن واحد کدگذاری می‌گویند. واحد کدگذاری، بلوکی است که یک کدگذار باید برایش تصمیم بگیرد که پیش‌بینی درون قابی یا پیش‌بینی بین قابی روی آن اعمال شود. اندازه واحد کدگذار به صورت $2^n \times 2^n$ است، n یک عدد صحیح است و می‌تواند بین ۳ تا ۶ تغییر کند. سپس برای پیش‌بینی یک بلوک بر اساس ساختار پیش‌بینی، واحد پیش‌بینی ایجاد می‌گردد، اندازه واحد پیش‌بینی به صورت $2^n \times 2^n$ است که n یک عدد صحیح است و می‌تواند بین ۲ تا ۶ تغییر کند. بعد از واحد پیش‌بینی، واحد تبدیل که برای فرایند تبدیل و چندی سازی^{۱۸} است، ایجاد می‌گردد. واحد تبدیل همیشه مربعی شکل است و اندازه آن می‌تواند مشابه واحد پیش‌بینی به صورت $2^n \times 2^n$ باشد، n یک عدد صحیح بین ۲ تا ۵ است.

در هر صورت برای هر واحد کدگذار یک واحد پیش‌بینی ایجاد می‌شود. برای تشکیل واحد پیش‌بینی، می‌توان از دو حالت پیش‌بینی داخلی و یا پیش‌بینی بین قابی استفاده نمود. سپس واحد پیش‌بینی از واحد کدگذاری کم می‌شود و تشکیل مانده یا تفاضل را می‌دهد و بعد از آن واحد تبدیل ایجاد می‌گردد.

واحد تبدیل، یک بلوک پایه و حاوی ضرایبی است که تابع تبدیل گسسته کسینوسی و چندی سازی روی آن اعمال شده است. اندازه‌های ممکن برای هر واحد تبدیل، 4×4 ، 8×8 ، 16×16 و 32×32 است. برای ایجاد هر واحد تبدیل یک درخت تشکیل می‌شود که آن را درخت چهارگوش مانده می‌نامند. این بدان معنی است که اندازه واقعی بلوک تبدیل، می‌تواند بر اساس ویژگی‌های سیگنال باقی‌مانده با استفاده از یک پارتیشن‌بندی مبتنی بر درخت چهارگوش مانده، تعریف شود. جاهایی که اندازه بلوک تبدیل کوچک‌تر است تغییرات تصویر بیشتر بوده و تصویر رزولوشن بیشتری خواهد داشت [۴۲].

در هر صورت از بلوک مانده، تبدیل گسسته کسینوسی گرفته شده و ضرایب بلوک تبدیل، چندی سازی می‌شود، این ضرایب توسط یک کدگذار آن‌تروپی، کم می‌شوند، سپس به همراه اطلاعات کدگذاری که در سرآیندها قرار می‌گیرند، از قبیل نوع پیش‌بینی بلوک، اندازه گام چندی سازی، اطلاعات پیش‌بینی، بردار حرکت و غیره که خود توسط آن‌تروپی کدگذار، کد شده‌اند، در رشته بیت خروجی قرار می‌گیرند. رشته بیت خروجی به واحد لایه انتزاعی شبکه داده می‌شود، این لایه مقدار رشته بیت را به صورت بیت، بایت درآورده و در واحدهای مخصوص خود جای می‌دهد [۴۲، ۴۳].

برای کدگشایی، عکس مراحل فوق انجام می‌شود. در واقع با کدگشایی، آن‌تروپی ضرایب بلوک تبدیل به دست می‌آید و با عکس تبدیل گسسته کسینوسی و استفاده از اطلاعات سرآیند درخت چهارگوش مانده، به بلوک مانده می‌رسیم. از طرفی با استفاده از اطلاعات تخمین یا پیش‌بینی که در سرآیندها موجود است، بلوک

تغییر محتوای قاب‌های مرجع با قاب فعلی است که خود نشان از تغییر محتوا در قاب‌ها یا در واقع نشان از ایجاد شات است؛ لذا از اختلاف CGF بین قاب‌ها برای تشخیص تغییر محتوای قاب‌ها و تشخیص شات استفاده می‌کنیم.

برای پیاده سازی این ایده طبق الگوریتم ۱ در ابتدا مجموع ضرایب قابل توجه بیشتر از یک در هر بلوک تبدیل را به دست می‌آوریم و سپس مجموع این ضرایب را برای هر یک از قاب‌های P در یک ویدیو محاسبه می‌کنیم. در نهایت مجموع CGF قاب P جاری را از مجموع CGF قاب P قبل بر اساس رابطه (۱) کم می‌کنیم تا اختلاف آن‌ها یا Dif_CGF مشخص شود، اگر این اختلاف از یک مقدار آستانه مناسب بیشتر باشد می‌توان تشخیص داد که در کدام یک از قاب‌ها شات اتفاق افتاده است.

$$Dif_CGF(i, i-1) = \quad (1)$$

$$\sum_{j=0}^n \text{abs} \frac{(CGF(j) - CGF(i-1))}{\text{picturewidth} \times \text{pictureheight}} \\ 4 \times 4$$

همان‌طور که در رابطه (۱) مشخص است، ما اختلاف ضرایب تبدیل غیر صفر بزرگ‌تر از یک، در دو قاب متوالی نوع P را در رشته قاب‌های یک ویدیو فشرده شده محاسبه می‌کنیم، اگر این اختلاف از یک آستانه طبق رابطه (۲) بیشتر باشد، مشخص می‌شود که محتوای دو قاب اختلاف زیادی دارند و شات اتفاق افتاده است.

$$Dif_CGF(i, i-1) > \text{Threshold} \quad (2)$$

به صورت کلی می‌توان بیان کرد که طرح‌های محاسباتی مطرح شده، میزان شباهت بین دو قاب متوالی را توصیف می‌کنند. پس اگر جمع اختلاف ضرایب غیر صفر، بین دو قاب متوالی بیشتر از حد آستانه شود، بر اساس الگوریتم (۱) شات تشخیص داده می‌شود.

- ۱ شروع
- ۲ رشته بیت فشرده ویدیو را دریافت کن
- ۳ تعداد فریم‌های موجود در رشته بیت ← leng
- ۴ n ← ۰
- ۵ Threshold ← ۲۵۰۰
- ۶ تا زمانیکه (n ≤ leng) دستورات ۷ تا ۱۲ را تکرار کن
- ۷ shot ← ۰
- ۸ مجموع مقادیر پرچم CFG فریم جاری ← CFG(n)
- ۹ dif_CFG ← CFG(n) - CFG(n-1)
- ۱۰ اگر (dif_CFG ≥ Threshold) شماره فریم جاری ← shot
- ۱۱ shot را چاپ کن
- ۱۲ n ← n+1
- ۱۳ پایان

الگوریتم ۱: روند تشخیص شات

تبدیل است، مقداره‌ی می‌گردد. ما از اطلاعات موجود در نقشه اهمیت جهت تحلیل قاب‌ها و تشخیص شات استفاده خواهیم کرد.

برای مقداره‌ی نقشه اهمیت، در ابتدا هر بلوک تبدیل پیمایش و سپس اطلاعاتی از آن توسط مجموعه‌ای از پرچم‌ها مطابق مثالی که در شکل ۱ قسمت الف آمده است، کم می‌شود. پیمایش بلوک از موقعیت شماره ۰ که مقدار آن عدد ۰ است (بالاترین فرکانس) شروع می‌شود و به صورت مورب به سمت موقعیت شماره ۱۵ که مقدار آن عدد ۹ است (پایین‌ترین فرکانس) حرکت می‌کند. در مسیر پیمایش، کدگذار به اولین ضریب غیر صفر که رسید مختصات آن ضریب را با استفاده از پرچم‌های Last_significant_coefficient (X & Y) کم می‌کند تا مقداره‌ی نقشه اهمیت از این موقعیت شروع شود و از افزونگی و ذخیره بی‌مورد صفرهایی که بعد از چندی سازی در فرکانس‌های بالای بلوک تبدیل ایجاد می‌شوند جلوگیری به عمل آید.

به موقعیت مشخص شده توسط پرچم ذکر شده در پاراگراف قبل مرز فرکانسی نیز می‌گویند، (مقداره‌ی نقشه اهمیت از موقعیت شماره ۶ در قسمت الف شکل ۱، شروع می‌شود) در کدگذاری هر بلوک، مشخص می‌گردد که ضرایبی با فرکانس بالاتر از این مرز همگی صفر هستند (موقعیت‌های شماره ۰ تا ۵) و نیاز به ذخیره در نقشه اهمیت ندارند، و ضرایبی با فرکانس پایین‌تر از این مرز اغلب غیر صفر هستند (موقعیت‌های شماره ۷ تا ۱۵) و در نقشه اهمیت ذخیره می‌شوند.

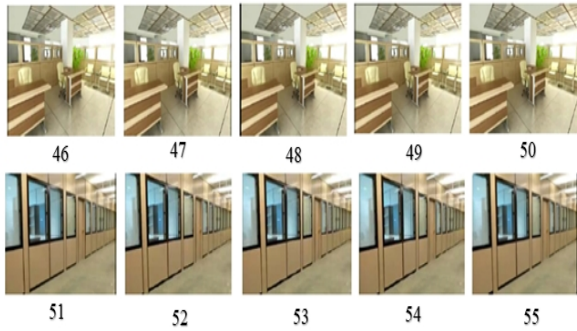
بعد از مشخص شدن مرز فرکانسی، کدگذار پیمایش را ادامه خواهد داد و ضرایب مرز فرکانسی و همگی ضرایب با فرکانس‌های پایین‌تر از مرز فرکانسی را به تفکیک، توسط پرچم Significant_coefficient_flag کم می‌کند (ضرایب قابل توجه یا ضرایب غیر صفر، با مقدار یک و ضرایب غیر قابل توجه یا صفر، با مقدار صفر کم می‌شوند). در نهایت، ضرایب بزرگ‌تر از یک، ضرایب بزرگ‌تر از دو و ضرایب منفی مشخص می‌گردند و به ترتیب توسط پرچم‌های که در پاراگراف بعدی ذکر شده، کم می‌شوند. به شکل ۱ قسمت ب نگاه کنید.

پرچم Coefficient_abs_level_greater_1_flag نشان دهنده‌ی ضرایب قابل توجه بیشتر از یک است (پرچم مورد استفاده در این مقاله) و پرچم Coefficient_abs_level_greater_2_flag نشان دهنده‌ی ضرایب قابل توجه بیشتر از دو (به جز مقادیر منفی و مقدار DC) است، پرچم Coefficients_Significant_flag نیز برای کد کردن نشانه‌های سازگار یا مقادیر منفی، مورد استفاده قرار می‌گیرد.

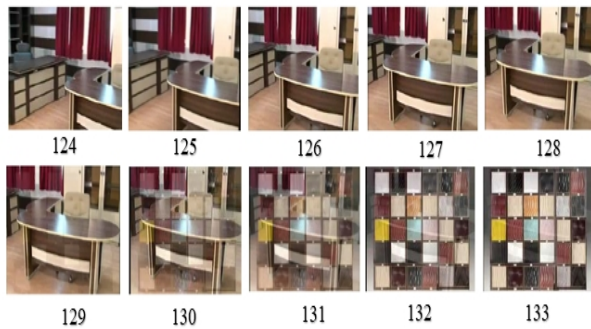
در بین پرچم‌های مشخص شده در شکل ۱، ما در این مقاله از محتوای پرچم coeff_abs_level_greater_1 با عنوان CGF که در مجموعه ساختار سلسله‌مراتبی، در قسمت سراینده استاندارد H.265 و در رشته بیت فشرده (فایل فشرده) وجود دارد، استفاده کرده‌ایم و بدون غیر فشرده کردن رشته بیت، محتوای این پرچم را محاسبه می‌کنیم.

در واقع این پرچم نشان می‌دهد که ضرایب تبدیل گسسته کسینوسی به چه میزان بزرگ هستند. از آنجا که تبدیل بعد از تخمین انجام می‌شود، تغییر قابل توجه ضرایب نشان دهنده تخمین نادرست یا

۵- شبیه‌سازی و نتایج



شکل ۲: نمونه‌ای از شات ناگهانی تشخیص داده شده در ویدیو ۱



شکل ۳: نمونه‌ای از شات تدریجی تشخیص داده شده در ویدیو ۱

برای ویدیو ۲ در جدول ۲ تعداد شات‌های درست تشخیص داده شده، تعداد شات‌های اشتباه تشخیص داده شده و تعداد شات‌هایی که تشخیص داده نشده با سه آستانه مختلف، بیان شده است و نمونه‌ای از شات تدریجی درست تشخیص داده شده نیز، با اجرای روش ارائه شده در شکل ۴ نشان داده شده است. برای ویدیو ۳ نیز در جدول ۳ تعداد شات‌های درست تشخیص داده شده، تعداد شات‌های اشتباه تشخیص داده شده و تعداد شات‌هایی که تشخیص داده نشده با سه آستانه مختلف، بیان شده است و نمونه‌ای از شات ناگهانی درست تشخیص داده شده، با اجرای روش ارائه شده در شکل ۵ نشان داده شده است. با توجه به آزمایش‌های انجام شده می‌توان به این نتیجه رسید که، روش ارائه شده به خوبی توانسته است، شات را در ویدیوهای فشرده شده تشخیص دهد، اما برای ارزیابی و مقایسه بهتر، از معیارهای عددی که در بخش بعد بیان شده استفاده خواهیم کرد.

جدول ۲: تعداد انواع شات‌های ویدیو ۲

حد آستانه	شات درست تشخیص داده شده	شات اشتباه تشخیص داده شده	شات تشخیص داده نشده
۱۵۰۰	۶	۱۵	۰
۲۰۰۰	۶	۱۴	۰
۲۵۰۰	۶	۶	۰

برای ارزیابی روش ارائه شده، این روش را جهت تشخیص شات، بر روی ویدیوهای فشرده شده با استاندارد H.265 اجرا کرده‌ایم. برای این منظور، از یک ویدیو ۴۷۱ قابی و دو ویدیو ۹۰۰ قابی که از صدا و سیمای ایران دریافت گردیده، استفاده شده است، این ویدیوها دارای تعداد زیادی شات از انواع مختلف می‌باشند. ویدیوها برای آزمایش به فرمت YUV تبدیل و ذخیره شده‌اند، سپس این ویدیوها به عنوان فایل ورودی به کدگذار استاندارد H.265 داده می‌شود تا رشته بیت فشرده شده‌ی هر یک از ویدیوها (فایل باینری) ایجاد شود.

برای کدگذاری یا فشرده کردن هر ویدیو، از نرم‌افزار آزمایشی فشرده‌سازی ویدیو نسخه HM-16/9 و همچنین تنظیمات و پارامترهای پیش فرض مشخص شده، استفاده شده است [۳۶]. بعد از فشرده‌سازی، طبق روش ارائه شده، بدون غیر فشرده سازی، مقدار CGF قاب‌ها را از رشته بیت فشرده‌ی ویدیو استخراج می‌کنیم و در نهایت با مقایسه اختلاف این مقادیر طبق رابطه (۱) و (۲) شات را در رشته بیت فشرده‌ی ویدیو تشخیص می‌دهیم و با شات‌های واقعی موجود در ویدیو مقایسه می‌نماییم.

در این راستا سه آستانه ثابت ۱۵۰۰، ۲۰۰۰ و ۲۵۰۰ به صورت تجربی و با آزمایش‌های مختلف به دست آمده است، جهت به چالش کشیدن روش ارائه شده در این مقاله از ویدیوهایی استفاده شده است که از نظر نوع شات و شدت روشنایی و حتی شباهت پس‌زمینه تفاوت‌های زیادی باهم دارند. پس باید آستانه ثابتی انتخاب شود تا در هر سه ویدیو که تفاوت‌های زیادی باهم دارند شات و محل دقیق آن مشخص شود.

برای مقایسه، تشخیص شات را با سه آستانه مذکور که به صورت تجربی به دست آمده است، انجام می‌دهیم:

- مجموع ضرایب قابل توجه بزرگ‌تر از یک، با آستانه ثابت ۱۵۰۰
- مجموع ضرایب قابل توجه بزرگ‌تر از یک، با آستانه ثابت ۲۰۰۰
- مجموع ضرایب قابل توجه بزرگ‌تر از یک، با آستانه ثابت ۲۵۰۰

برای ویدیو ۱ در جدول ۱ تعداد شات‌های درست تشخیص داده شده، تعداد شات‌های اشتباه تشخیص داده شده و تعداد شات‌هایی که تشخیص داده نشده با آستانه‌های ۱۵۰۰، ۲۰۰۰ و ۲۵۰۰ بیان شده است.

جدول ۱: تعداد انواع شات‌ها در ویدیو ۱

حد آستانه	شات درست تشخیص داده شده	شات اشتباه تشخیص داده شده	شات تشخیص داده نشده
۱۵۰۰	۹	۴	۱
۲۰۰۰	۷	۴	۳
۲۵۰۰	۹	۱	۱

در شکل ۲ نمونه‌ای از شات ناگهانی درست تشخیص داده شده و همچنین در شکل ۳ نمونه‌ای از شات تدریجی درست تشخیص داده شده در ویدیو ۱، با اجرای روش ارائه شده، نشان داده شده است.

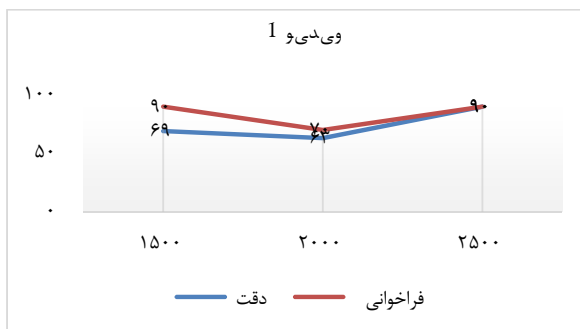
با پیاده‌سازی روش ارائه شده در تشخیص شات، مقدار دقت و فراخوانی را برای رشته بیت فشرده شده هر سه ویدیو با سه آستانه‌ی مختلف، محاسبه کرده‌ایم، شات‌های تشخیص داده شده و نوع و مکان آن‌ها نیز با مقدار آستانه‌های ذکر شده، در جداول ۴ تا ۶ نشان داده شده است. در جداول، علامت × به معنای عدم تشخیص شات و علامت ✓ به معنای تشخیص درست شات در روش ارائه شده است.

طبق جدول ۴، دقت و فراخوانی، پیاده‌سازی روش ارائه شده بر روی فایل فشرده شده ویدیو ۱ بر اساس سه آستانه‌ی مختلف، مورد ارزیابی قرار گرفته و همچنین همان‌طور که نتایج نشان می‌دهد، با روش ارائه شده و آستانه ۲۵۰۰ به دقت ۹۰٪ و فراخوانی ۹۰٪ رسیده‌ایم، این آستانه در ویدیو ۱، نسبت به آستانه‌های ثابت دیگر عملکرد بهتری در تشخیص شات داشته است.

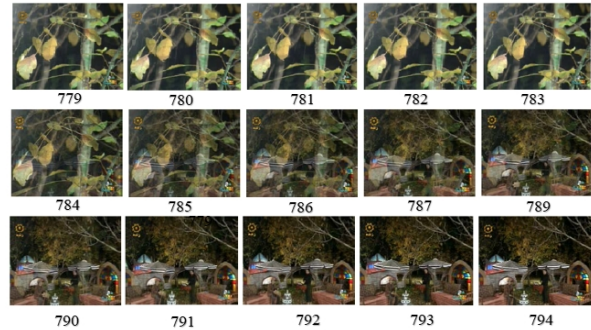
جدول ۴: ارزیابی ویدیو ۱، نوع و مکان شات‌ها با سه آستانه مختلف

ردیف	قاب‌های شات شده	نوع شات	آستانه: ۱۵۰۰	آستانه: ۲۰۰۰	آستانه: ۲۵۰۰
۱	۹-۶	تدریجی	✓	✓	✓
۲	۵۲، ۵۱	ناگهانی	✓	✓	✓
۳	۷۴-۷۱	تدریجی	✓	✓	✓
۴	۹۷-۹۳	تدریجی	✓	✓	✓
۵	۱۳۲-۱۲۹	تدریجی	✓	×	✓
۶	۱۵۸-۱۵۵	تدریجی	✓	✓	✓
۷	۱۸۰-۱۷۶	تدریجی	✓	×	✓
۸	۲۱۳-۲۱۲	تدریجی	✓	✓	✓
۹	۲۳۷-۲۳۴	تدریجی	✓	✓	✓
۱۰	۳۹۶، ۳۹۵	ناگهانی	×	×	×
۱۰	۱۰	تعداد شات موجود	۱۰	۱۰	۱۰
۹	۷	تعداد شات درست تشخیص داده شده	۹	۷	۹
۱۰	۱۱	تعداد کل شات‌های تشخیص داده شده	۱۳	۱۱	۱۰
		دقت	۶۹٪	۶۳٪	۹۰٪
		فراخوانی	۹۰٪	۷۰٪	۹۰٪

شکل ۶ نمودار دقت و فراخوانی روش ارائه شده در تشخیص شات را در فایل فشرده شده ویدیو ۱، با سه آستانه مختلف بیان می‌کند. همان‌طور که در این شکل نشان داده شده، مقدار آستانه ۲۵۰۰ از سایر آستانه‌ها عملکرد بهتری در تشخیص شات داشته است.



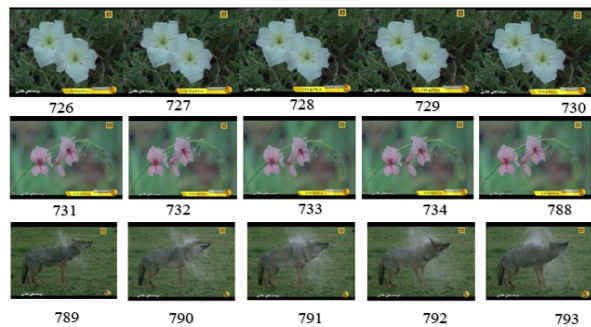
شکل ۶: دقت و فراخوانی آستانه‌های مختلف در ویدیو ۱



شکل ۴: نمونه‌ای از شات تدریجی تشخیص داده شده در ویدیو ۲

جدول ۳: تعداد انواع شات‌های ویدیو ۳

حد آستانه	شات درست تشخیص داده شده	شات اشتباه تشخیص داده شده	شات تشخیص داده نشده
۱۵۰۰	۸	۱	۳
۲۰۰۰	۷	۲	۴
۲۵۰۰	۷	۰	۴



شکل ۵: نمونه‌ای از شات ناگهانی تشخیص داده شده در ویدیو ۳

۵-۱- ارزیابی نتایج

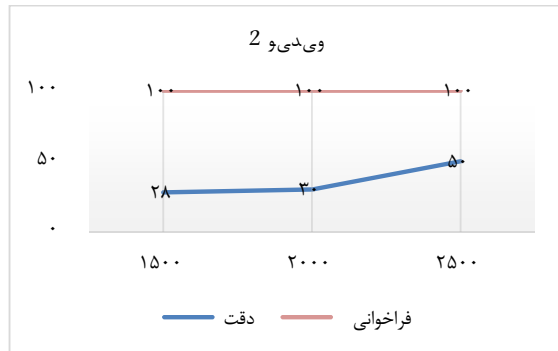
در این بخش معیارهای معروف سنجش کارایی که شامل، دقت^{۱۹} و فراخوانی^{۲۰} است [۴۱] را در روش ارائه شده محاسبه می‌کنیم. دقت و فراخوانی، معیارهای سنجش موفقیت روش ارائه شده، در تشخیص شات هستند. دقت به این معنی است که از میان شات‌های تشخیص داده شده، چه تعداد شات، درست تشخیص داده شده است، و فراخوانی به این معنی است که چه تعداد از شات‌های موجود در ویدیو، تشخیص داده شده است. این دو معیار از رابطه (۳) و (۴) محاسبه می‌شود.

$$R = \frac{c}{c+m} \quad (۳)$$

$$P = \frac{c}{c+fp} \quad (۴)$$

در رابطه (۳)، R مقدار دقت، c تعداد شات‌های درست تشخیص داده شده، m تعداد شات‌های تشخیص داده نشده است. در رابطه (۴)، P مقدار فراخوانی و همچنین fp تعداد شات‌هایی که اشتباه تشخیص داده شده را بیان می‌کند.

همان‌طور که در این شکل نشان داده شده، مقدار آستانه ۲۵۰۰ از سایر آستانه‌ها عملکرد بهتری در تشخیص شات داشته است.



شکل ۷: دقت و فراخوانی آستانه‌های مختلف در ویدیو ۲

در نهایت طبق جدول ۶، با پیاده‌سازی روش ارائه شده بر روی فایل فشرده ویدیو ۳ با آستانه‌های مختلف، دقت و فراخوانی مورد ارزیابی قرار گرفته است، همان‌طور که نتایج نشان می‌دهد، با روش ارائه شده و آستانه ۲۵۰۰ به دقت ۱۰۰٪ و فراخوانی ۶۳٪ رسیده‌ایم، این آستانه در ویدیو ۳، نسبت به آستانه‌های ثابت دیگر عملکرد نسبتاً بهتری در تشخیص شات داشته است.

جدول ۶: ارزیابی ویدیو ۳، نوع و مکان شات‌ها با سه آستانه مختلف

آستانه:	آستانه:	آستانه:	نوع شات	قاب‌های شات شده	ردیف
۲۵۰۰	۲۰۰۰	۱۵۰۰			
x	x	√	ناگهانی	۱۶، ۱۵	۱
x	x	√	ناگهانی	۱۰۷، ۱۰۶	۲
x	x	x	ناگهانی	۱۸۱، ۱۸۰	۳
√	√	√	ناگهانی	۲۷۴، ۲۷۳	۴
√	√	√	ناگهانی	۳۶۳، ۳۶۲	۵
√	√	√	ناگهانی	۴۵۵، ۴۵۴	۶
√	√	√	ناگهانی	۵۸۲، ۵۸۱	۷
√	√	√	ناگهانی	۶۶۳، ۶۶۲	۸
√	√	√	ناگهانی	۷۳۱، ۷۳۰	۹
√	√	x	ناگهانی	۷۸۹، ۷۸۸	۱۰
x	x	x	ناگهانی	۸۶۲، ۸۶۱	۱۱
۱۱	۱۱	۱۱	تعداد شات موجود		
۷	۷	۸	تعداد شات درست تشخیص داده شده		
۷	۹	۹	تعداد کل قاب‌های تشخیص داده شده		
% ۱۰۰	% ۷۷	% ۸۸	دقت		
% ۶۳	% ۶۳	% ۷۲	فراخوانی		

شکل ۸ نمودار دقت و فراخوانی روش ارائه شده در تشخیص شات را، در ویدیو ۳، با سه آستانه مختلف بیان می‌کند. همان‌طور که در این شکل نشان داده شده، مقدار آستانه ۲۵۰۰ از سایر آستانه‌ها عملکرد نسبتاً بهتری در تشخیص شات داشته است.

در ادامه فایل فشرده‌شده ویدیو ۲ را با روش ارائه شده و آستانه‌های مختلف مورد آزمایش قرار داده‌ایم، همان‌طور که در جدول ۵ مشاهده می‌کنید نتایج این آزمایش به تفکیک آستانه‌های مختلف بیان شده است بر اساس آزمایش‌ها، با روش ارائه شده و آستانه ۲۵۰۰ به دقت ۵۰٪ و فراخوانی ۱۰۰٪ رسیده‌ایم که این آستانه در ویدیو ۲، نیز نسبت به آستانه‌های ثابت دیگر عملکرد بهتری در تشخیص شات داشته است.

جدول ۵: ارزیابی ویدیو ۲، نوع و مکان شات‌ها با سه آستانه مختلف

ردیف	قاب‌های شات شده	نوع شات	آستانه:	آستانه:	آستانه:
			۲۵۰۰	۲۰۰۰	۱۵۰۰
۱	۵۵-۳۴	تدریجی	√	√	√
۲	۱۸۵، ۱۸۴	ناگهانی	√	√	√
۳	۲۳۹، ۲۳۸	ناگهانی	√	√	√
۴	۳۵۸-۳۳۳	تدریجی	√	√	√
۵	۵۱۳-۴۴۵	تدریجی	√	√	√
۶	۸۵۲-۷۷۲	تدریجی	√	√	√
تعداد شات موجود					
تعداد شات درست تشخیص داده شده					
تعداد کل قاب‌های تشخیص داده شده					
دقت					
فراخوانی					
۶	۶	۶	% ۵۰	% ۳۰	% ۲۸
۶	۶	۶	% ۱۰۰	% ۱۰۰	% ۱۰۰
۱۲	۲۰	۲۱			

در اینجا این نکته قابل ذکر است که ویدیو ۲، همان‌طور که در جدول ۵ نشان داده شده است، از ۲ شات ناگهانی و ۴ شات تدریجی تشکیل شده است، روش ارائه شده با آستانه ۲۵۰۰ به خوبی توانسته است محل دقیق این ۶ شات را تشخیص و به فراخوانی ۱۰۰٪ برسد، اما به دلیل وجود شات‌های تدریجی طولانی در این ویدیو که گاهی تعداد زیادی قاب را درگیر خود می‌کند، (به‌عنوان مثال شات شماره ۴ شامل ۸۰ قاب است) تعداد قاب بیشتری به‌عنوان شات تشخیص داده شده است، به‌عنوان مثال در شات ردیف ۱ قاب‌های شماره ۳۴ و ۵۰، در شات ردیف ۴ قاب‌های شماره ۳۳۳ و ۳۴۲، در شات ردیف ۵ قاب‌های شماره ۴۴۵، ۴۸۹ و ۵۰۱ و در نهایت در شات ردیف ۶ قاب‌های ۷۷۲، ۷۹۴ و ۷۶۰ به‌عنوان شات تشخیص داده شده است و باعث شده است که دقت روش ارائه شده به ۵۰٪ برسد، اگر در شات‌های تدریجی ذکر شده (شات‌های ردیف ۱، ۴، ۵ و ۶) که بیشتر از یک قاب به‌عنوان شات تشخیص داده شده است، یک قاب را به‌عنوان شات در نظر بگیریم (در شات ردیف ۱ قاب شماره ۳۴، در شات ردیف ۴ قاب شماره ۳۳۳، در شات ردیف ۵ قاب شماره ۴۴۵ و در شات ردیف ۶ قاب شماره ۷۷۲) و از شات‌های تشخیص داده شده دیگر صرف‌نظر کنیم به دقت ۱۰۰٪ خواهیم رسید.

در شکل ۷ نیز نمودار دقت و فراخوانی روش ارائه شده در تشخیص شات، در فایل فشرده ویدیو ۲، با سه آستانه مختلف بیان شده است.

است، از طرفی انجام کار روی کدگذار نیاز به زمان زیادی دارد تا قاب های هر ویدیو به صورت جداگانه کدگذاری و جهت تشخیص شات مقایسه گردد. ما برای حل این مشکل روش جدیدی را ارائه داده ایم که می تواند مستقیماً و با سرعت بالا از روی فایل فشرده شده به اطلاعات قابها دسترسی و شات را تشخیص دهد.

در جدول ۷ روش ارائه شده در مقاله [۱۵] را از نظر دقت و فراخوانی با روش ارائه شده در این مقاله مقایسه کرده ایم، همان طور که در جدول ۷ نشان داده شده است، روش ارائه شده در این مقاله علاوه بر اینکه دقت و فراخوانی بالاتری در تشخیص شات نسبت به روش ارائه شده در مقاله [۱۵] دارد سرعت بسیار بالاتری نیز در تشخیص شات (به دلیل اینکه روش ارائه شده در این مقاله بر روی فایل فشرده ویدیو پیاده سازی می شود) دارد.

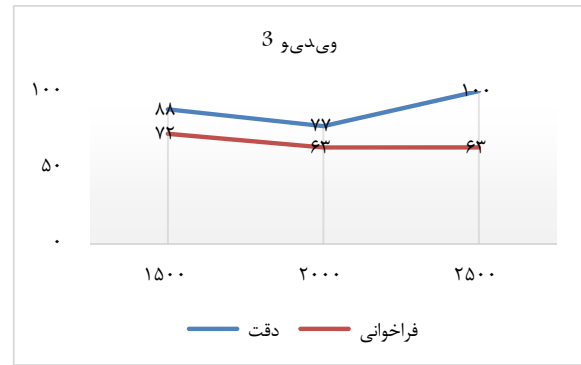
جدول ۷: مقایسه روش پیشنهادی با روش ارائه شده در [۱۵]

روش	دقت	فراخوانی
تشخیص شات در کدگذار استاندارد H.265 [۱۵]	۹۵٪	۹۲٪
روش پیشنهادی تشخیص شات، در فایل فشرده ویدیو	۹۶٪	۹۴٪

در مقاله [۲۲] یک روش برای تشخیص شات مبتنی بر آنالیز توزیع واحد تبدیل قاب های نوع I در فایل فشرده ویدیو بر اساس استاندارد H.265 ارائه شده است. مشکلی که در این روش وجود دارد این است که فقط بر روی فایل فشرده شده ویدیوهایی که با قاب های نوع I ایجاد شده اند اجرا می شود، با توجه به اینکه قاب های نوع I نسبت به قاب های نوع P و B حجم خیلی بیشتری از اطلاعات را در خود نگه می دارند و بر اساس شرایط خاص، برای کاربردهای بلادرنگ نیاز به کم شدن حجم فایل فشرده ویدیو با کم کردن تعداد قاب های نوع I در کدگذار است، پس این روش برای استریم ویدیوهایی که به خاطر کمتر شدن حجم فایل فشرده از تعداد قاب های نوع I کمتری استفاده می کنند به عنوان مثال استریم ویدیو IPPPPPPP... یا IPBPPPPB... یا اصلاً مناسب نیست.

ما برای حل این مشکل روش جدیدی را ارائه داده ایم که می تواند مستقیماً از روی فایل فشرده شده به اطلاعات هر سه نوع قاب دسترسی و جهت تشخیص شات این اطلاعات را تحلیل نماید. اما با توجه به اینکه قاب نوع P حجم کمتری از اطلاعات را دارد و در کاربردهای بلادرنگ این قاب بیشترین کاربرد را دارند، روش ارائه شده در این مقاله بدون نیاز به قاب های نوع I شات را در فایل فشرده شده ویدیو تشخیص می دهیم.

جهت فراهم نمودن شرایط یکسان برای مقایسه، با توجه به اینکه مقاله ما این قابلیت را دارد که بر روی هر سه نوع قاب موجود در ویدیو (قاب های نوع I، P و B) پیاده سازی شود، در ابتدا روش ارائه شده خود را بر روی فایل فشرده شده مقاله [۲۲] اجرا و شات را تشخیص داده ایم، سپس دو ویدیو مورد استفاده در مقاله [۲۲] (با رزولوشن ۱۰۸۰×۱۹۲۰) بر اساس پارامترهای مقاله خود



شکل ۸: دقت و فراخوانی آستانه های مختلف در ویدیو ۳

این نکته قابل ذکر است که به عنوان مثال در ویدیو ۲ به دلیل عدم تشابه قابها و همچنین شات های تدریجی طولانی که گاهی تعداد زیادی قاب را درگیر خود می کند، مقدار مقادیر غیر صفر بلوک های تبدیل و در نهایت قابها، عدد بیشتری می شود، در این ویدیو هر چه مقدار آستانه بیشتر باشد دقت و فراخوانی در تشخیص شات بالاتر خواهد رفت. به عنوان مثال با آستانه ثابت ۲۹۰۰ به دقت ۹۰٪ و فراخوانی ۱۰۰٪ در تشخیص شات در فایل فشرده این ویدیو رسیده ایم. همچنین در ویدیو ۳ به دلیل شباهت زیاد قابها و عدم وجود شات های تدریجی مقدار مقادیر غیر صفر بلوک های تبدیل و در نهایت قابها عدد کمتری می شود، در این ویدیو هر چه مقدار آستانه کمتر باشد، دقت و فراخوانی در تشخیص شات بالاتر خواهد رفت. به عنوان مثال با آستانه ثابت ۹۰۰ به دقت ۱۰۰٪ و فراخوانی ۱۰۰٪ در تشخیص شات در فایل فشرده این ویدیو رسیده ایم. اما باید یک آستانه ثابت انتخاب می شد که بتواند در تشخیص شات در هر ویدیو عملکرد مناسبی داشته باشد. ما آستانه ثابت ۲۵۰۰ را به دلیل اینکه نسبت به آستانه های دیگر عملکرد مناسب تری داشته به عنوان آستانه ثابت در نظر گرفته ایم.

۶- مقایسه روش ارائه شده با سایر روشها

در این قسمت روش ارائه شده در این مقاله را با روش های ارائه شده در مقالات [۲۲، ۱۵] مقایسه نموده ایم. در این راستا در ابتدا روش های ارائه شده در مقالات مذکور را آنالیز و مشکلات موجود را بیان نموده ایم سپس روش پیشنهادی خود را روی ویدیوهای مقالات ذکر شده پیاده سازی نموده ایم (با توجه به اینکه ویدیوهای استفاده شده در مقالات با یکدیگر متفاوت هستند، روش ارائه شده را به صورت مجزا بر روی ویدیوهای استفاده شده در مقالات مذکور اجرا نموده ایم) و در نهایت نتایج را به تفکیک با هم مقایسه نموده ایم.

در مقاله [۱۵] با استفاده از هیستوگرام ساده شده گرادیان، شات در قسمت کدگذار استاندارد H.265 تشخیص داده می شود. با توجه به اینکه امروزه ویدیوها به صورت فشرده ذخیره و منتقل می شوند، جهت پیاده سازی این روش نیاز به فایل اصلی ویدیو (فایل غیر فشرده ویدیو) است و تبدیل فایل فشرده ویدیو به فایل غیر فشرده ویدیو زمان بر

[7] M. Parmar and M. C. Angelides, "MAC-REALM: a video content feature extraction and modelling framework," *The Computer Journal*, vol. 58, no. 9, pp. 2135-2171, 2015.

[8] Y. Ding and G. Fan, "Camera view-based american football video analysis," in *Eighth IEEE International Symposium on Multimedia (ISM'06)*, 2006: IEEE, pp. 317-322.

[9] W. Hu, N. Xie, L. Li, X. Zeng, and S. Maybank, "A survey on visual content-based video indexing and retrieval," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 41, no. 6, pp. 797-819, 2011.

[10] A. Midya and S. Sengupta, "Switchable video error concealment using encoder driven scene transition detection and edge preserving SEC," *Multimedia Tools and Applications*, vol. 74, no. 6, pp. 2033-2054, 2015.

[11] T. Liu and J. R. Kender, "Computational approaches to temporal sampling of video sequences," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 3, no. 2, p. 7, 2007.

[12] R. Trichet, R. Nevatia, and B. Burns, "Video event classification with temporal partitioning," in *2015 12th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2015: IEEE, pp. 1-6.

[13] H. Jiang, G. Zhang, H. Wang, and H. Bao, "Spatio-temporal video segmentation of static scenes and its applications," *IEEE Transactions on Multimedia*, vol. 17, no. 1, pp. 3-15, 2014.

[14] K. Choroś, "Improved video scene detection using player detection methods in temporally aggregated TV sports news," in *International Conference on Computational Collective Intelligence*, 2014: Springer, pp. 633-643.

[15] C. C. Oprea, R. O. Preda, I. Pirnóg, and R. A. Dobre, "Video Shot Boundary Detection for Low Complexity HEVC Encoders," in *2018 10th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, 2018: IEEE, pp. 1-4.

[16] Z.-M. Lu and Y. Shi, "Fast video shot boundary detection based on SVD and pattern matching," *IEEE Transactions on Image processing*, vol. 22, no. 12, pp. 5136-5145, 2013.

[17] K. Choroś, "False and miss detections in temporal segmentation of TV sports news videos—causes and remedies," in *New Research in Multimedia and Internet Systems*: Springer, 2015, pp. 35-46.

[18] L. H. Iwan and J. A. Thom, "Temporal video segmentation: detecting the end-of-act in circus performance videos," *Multimedia Tools and Applications*, vol. 76, no. 1, pp. 1379-1401, 2017.

[19] D. Dutta, S. K. Saha, and B. Chanda, "A shot detection technique using linear regression of shot transition pattern," *Multimedia Tools and Applications*, vol. 75, no. 1, pp. 93-113, 2016.

[20] I. Richardson, "An introduction to high efficiency video coding VCODEX video compression," ed. 2013.

[21] H. Wang, A. Divakaran, A. Vetro, S.-F. Chang, and H. Sun, "Survey of compressed-domain features used in audio-visual indexing and analysis," *Journal of Visual Communication and Image Representation*, vol. 14, no. 2, pp. 150-183, 2003.

[22] N. Dorfeshan, M. Ramezani, "Compressed Domain Scene Change Detection Based on Transform Units Distribution in High Efficiency Video Coding Standard," *Journal of Computer & Robotics*, Sep 18;11(2):41-8, 2018

[23] R. V. Babu, M. Tom, and P. Wadekar, "A survey on compressed domain video analysis techniques," *Multimedia Tools and Applications*, vol. 75, no. 2, pp. 1043-1078, 2016.

[24] R. Mishra and S. Singhai, "A Review on Different Methods of Video Shot Boundary Detection," *International Journal of Management, IT and Engineering*, vol. 2, no. 9, pp. 199-214, 2012.

[25] F. Zargari, M. Mehrabi, and M. Ghanbari, "A robust compressed domain feature vector for texture based image retrieval," in *2008 International Workshop on Content-Based Multimedia Indexing*, 2008: IEEE, pp. 489-495.

[26] H. Zhang, C. Y. Low, and S. W. Smoliar, "Video parsing and browsing using compressed data," *Multimedia tools and applications*, vol. 1, no. 1, pp. 89-111, 1995.

[27] J. Meng, Y. Juan, and S.-F. Chang, "Scene change detection in an MPEG-compressed video sequence," in *Digital Video Compression: Algorithms and Technologies 1995*, 1995, vol. 2419: International Society for Optics and Photonics, pp. 14-25.

[28] J. Feng, K.-T. Lo, and H. Mehrpour, "Scene change detection algorithm for MPEG video sequence," in *Proceedings of 3rd IEEE*

(...) (GOP=IPBBPBBPBB...) فشرده و روش ارائه شده را بر روی این قابها پیاده‌سازی کرده‌ایم. نتایج آزمایش‌ها در جدول ۸ نشان داده شده است. روش ارائه شده علاوه بر اینکه به خوبی شات‌ها را تشخیص داده، مقاومت خوبی نسبت به تغییرات شرایط کدگذاری از جمله تغییر چیدمان قاب‌ها دارد.

جدول ۸: مقایسه روش پیشنهادی با روش ارائه شده در [۲۲]

روش	دقت	فراخوانی
تشخیص شات بر اساس آنالیز تعداد واحد تبدیل بر پایه قاب I در فایل فشرده ویدئو [۲۲]	٪ ۸۸	٪ ۸۳
روش پیشنهادی تشخیص شات در فایل فشرده ویدئو بر پایه قاب I	٪ ۹۰	٪ ۸۸
روش پیشنهادی تشخیص شات در فایل فشرده ویدئو بر پایه قاب P	٪ ۹۰	٪ ۸۲

۷- نتیجه‌گیری

در این مقاله روشی برای تشخیص تغییر صحنه یا شات بر روی تصاویر کد شده در استاندارد کدگذاری ویدئویی H.265 ارائه شد. در این روش در ابتدا ویژگی قاب استخراج می‌شود و مقادیر ضرایب غیر صفر بلوک تبدیل، با استفاده از پرچمی که در رشته بیت قرار دارد به دست می‌آید، مجموع این ضرایب را برای هر قاب محاسبه می‌کنیم، سپس با یک آستانه مناسب تغییر صحنه در دنباله‌ای از ویدیوها را تشخیص می‌دهیم.

آزمایش‌های ما نشان می‌دهد روش ارائه شده می‌تواند با دقت ٪۸۰ و فراخوانی ٪۸۴/۳ بدون غیر فشرده‌سازی فایل ویدئو، شات‌ها را از فایل ویدئو فشرده شده با استاندارد H.265 تشخیص دهد. مزیت روش ارائه شده پیچیدگی کم و دقت و فراخوانی بالای آن در تشخیص شات‌های تدریجی و ناگهانی و مکان دقیق شات‌ها است. علاوه بر این، برای کاربردهای بلادرنگ بسیار مناسب است.

مراجع

[1] R. Priya and T. Shanmugam, "A comprehensive review of significant researches on content based indexing and retrieval of visual information," *Frontiers of Computer Science*, vol. 7, no. 5, pp. 782-799, 2013.

[2] A. Umamakeswari and A. Rajaraman, "Object based video analysis, interpretation and tracking," *Journal of Computer Science*, vol. 3, no. 10, pp. 818-822, 2007.

[3] S. H. Abdulhussain, A. R. Ramli, M. I. Saripan, B. M. Mahmmod, S. A. R. Al-Haddad, and W. A. Jassim, "Methods and challenges in shot boundary detection: a review," *Entropy*, vol. 20, no. 4, p. 214, 2018.

[4] M. Del Fabro and L. Böszörményi, "State-of-the-art and future challenges in video scene detection: a survey," *Multimedia systems*, vol. 19, no. 5, pp. 427-454, 2013.

[5] I. González-Díaz, T. Martínez-Cortés, A. Gallardo-Antolín, and F. Díaz-de-María, "Temporal segmentation and keyframe selection methods for user-generated video search-based annotation," *Expert Systems with Applications*, vol. 42, no. 1, pp. 488-502, 2015.

[6] M. B. Fayk, H. A. El Nemr, and M. M. Moussa, "Particle swarm optimisation based video abstraction," *Journal of Advanced Research*, vol. 1, no. 2, pp. 163-167, 2010.

- IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, 2017: IEEE, pp. 174-179.
- [38] J.-Y. Huang, M.-F. Lin, and P.-c. Chang, "Transcoding or not? — A study of quantization configuration for H. 264-to-HEVC transcoding," in *2017 IEEE 6th Global Conference on Consumer Electronics (GCCE)*, 2017: IEEE, pp. 1-2.
- [39] F. Zargari, M. Mehrabi, and M. Ghanbari, "Compressed domain texture based visual information retrieval method for I-frame coded pictures," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 2, pp. 728-736, 2010.
- [40] V. Sze and M. Budagavi, "High throughput CABAC entropy coding in HEVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1778-1791, 2012.
- [41] T. Chattopadhyay, A. Chaki, and U. Garain, "A fast method for detection of video shot boundaries using compressed domain features of H. 264 for PVR enabled Set Top Boxes: A novel approach," in *2009 First International Conference on Computational Intelligence, Communication Systems and Networks*, 2009: IEEE, pp. 6-11.
- [42] V. Sze, M. Budagavi, and G. J. Sullivan, "High efficiency video coding (HEVC)," in *Integrated circuit and systems, algorithms and architectures*, vol. 39: Springer, 2014, p. 40.
- [43] M. Wien, "High efficiency video coding," *Coding Tools and specification*, pp. 133-160, 2015.
- [44] E. De la Torre, R. Rodriguez-Sanchez, and J. L. Martinez, "Fast video transcoding from HEVC to VP9," *IEEE Transactions on Consumer Electronics*, vol. 61, no. 3, pp. 336-343, 2015.
- International Conference on Image Processing*, 1996, vol. 2: IEEE, pp. 821-824.
- [29] I. Koprinska and S. Carraio, "Detecting and classifying video shot boundaries in MPEG compressed sequences," in *9th European Signal Processing Conference (EUSIPCO 1998)*, 1998: IEEE, pp. 1-4.
- [30] R. Zabih, J. Miller, and K. Mai, "A feature-based algorithm for detecting and classifying production effects," *Multimedia systems*, vol. 7, no. 2, pp. 119-128, 1999.
- [31] F. Akrami and F. Zargari, "An efficient compressed domain video indexing method," *Multimedia tools and applications*, vol. 72, no. 1, pp. 705-721, 2014.
- [32] S. Biswas and R. V. Babu, "H. 264 compressed video classification using histogram of oriented motion vectors (HOMV)," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013: IEEE, pp. 2040-2044.
- [33] R. V. Babu and K. Ramakrishnan, "Compressed domain video retrieval using object and global motion descriptors," *Multimedia Tools and Applications*, vol. 32, no. 1, pp. 93-113, 2007.
- [34] B.-L. Yeo and B. Liu, "On the extraction of DC sequence from MPEG compressed video," in *Proceedings, International Conference on Image Processing*, 1995, vol. 2: IEEE, pp. 260-263.
- [35] M. A. Hasan, M. Xu, X. He, and Y. Wang, "A camera motion histogram descriptor for video shot classification," *Multimedia Tools and Applications*, vol. 74, no. 24, pp. 11073-11098, 2015.
- [36] https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/trunk/doc/software-manual.pdf
- [37] A. A. Elrowayati, M. F. L. Abdullah, A. A. Manaf, and A. S. Alfagi, "Tampering detection of double-compression with the same quantization parameter in HEVC video streams," in *2017 7th*

زیر نویس ها

- ¹¹ Coding Tree Block
- ¹² luma
- ¹³ chroma
- ¹⁴ Coding Unit
- ¹⁵ Prediction Unit
- ¹⁶ Coding Block
- ¹⁷ Prediction Block
- ¹⁸ Quantization
- ¹⁹ Precision
- ²⁰ Recall

- ¹ Hard Transition
- ² Soft Transition
- ³ Significant Flag
- ⁴ Transform Block
- ⁵ Discrete Cosine Transform
- ⁶ Transform Unit
- ⁷ Network Abstraction Layer
- ⁸ Video Parameter Set
- ⁹ quad-tree
- ¹⁰ Coding Tree Unit