Różek *et al.*
EURASIP Journal on Image and Video Processing (2025) 2025:18
https://doi.org/10.1186/s13640-025-00682-3

EURASIP Journal on Image and Video Processing

RESEARCH Open Access

Video coding for machines using region-of-interest-based retargeting

Sławomir Różek[1], Olgierd Stankiewicz[1*], Sławomir Maćkowiak[1], Tomasz Grajek[1], Jakub Stankowski[1], Maciej Wawrzyniak[1], Mateusz Lorkiewicz[1], Ding Ding[2], Shan Liu[2] and Marek Domański[1]

*Correspondence: olgierd.stankiewicz@put.poznan.pl

[1] Institute of Multimedia Telecommunications, Poznan University of Technology, Poznań, Poland
[2] Tencent, Palo Alto, USA

Abstract

The work is focused on video compression for the scenarios, where the decoded video serves not only human viewers but also as input for systems implementing various machine vision tasks, such as object detection and tracking. The proposed innovative tool is based on retargeting video frames processed based on Regions of Interest (RoI), corresponding to the individual objects detected in the frames. Experimental evaluation demonstrates significant average bitrate reduction while maintaining the same quality, ranging from 3 to 57% depending on the machine vision task and encoding scenario. The proposal underwent thorough consideration within the MPEG group and was adopted for the upcoming Video Coding for Machines (VCM) technology.

1 Introduction

Traditional video coding standards, such as Advanced Video Coding (AVC), High-Efficiency Video Coding (HEVC), or Versatile Video Coding (VVC) [1–6], are mainly optimized for human perception. In today's world, with the rapid development of machine-processing algorithms, a significant share of video applications is ultimately processed by machines [7, 8]. Therefore, there is a clear mismatch between the assumed goals of using traditional video coding standards and the application requirements defined by the need for machine vision processing. Such processing presents unique challenges that traditional video coding standards cannot adequately address [7], and thus there is an increasing need to redefine the approach to vision representation.

In real scenarios, the quality of video data can be affected by a variety of factors [8]. Inappropriate lighting conditions and too much or too little illumination can lead to the loss of important details. The movement of objects and camera shake can result in blurred content. The presence of noise and interference in the material, which can be the result of equipment imperfections, weather conditions or interaction with the environment can introduce distortion. Varying camera perspectives or different viewing angles can cause content distortion. These factors can challenge machines to accurately extract relevant features for video analysis and understanding of video content.

Another source of quality degradation is compression artifacts. Research [9, 10] clearly indicates that the compression process can significantly reduce the accuracy of various

Springer Open

© The Author(s) 2025. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

machine algorithms and tasks [8], especially at low data rates. Thus, image and video coding presents significant challenges, requiring a balance to be struck between compression efficiency and the necessary data quality to guarantee machine efficiency and reliability in diverse applications, where compression quality requirements vary depending on the specific task or application.

The expectation of high performance of machine processing algorithms, dependent on data quality, while maintaining the efficiency of traditional compression methods, creates a challenge that requires new insights into the representation of images and videos. These factors have sparked a growing interest in video coding research, especially concerning scenarios, where the decoded data serves as input to machine vision tasks. Hence the need to develop specialized coding tools, named Video Coding for Machine (VCM), adapted to the requirements of machine vision processing (Fig. 1).

In this paper, we address the abovementioned need by proposing a new innovative compression tool for VCM. The proposed tool employs retargeting of the video frames that are processed based on the Regions of Interest (RoI), corresponding to the individual objects detected in the frames.

2 State of the art

The approach to image and video compression optimized for machine vision multi-tasks has gained academic attention in recent years [9, 11–15]. The solutions found in the literature focus on: extracting information about key points and transmitting this information without compression [11], using compression methods optimized to preserve object descriptors [9, 12], encoding common representations of features and textures [13, 14], reducing signal energy before compression without impairing the ability to perform machine tasks after decoding [15], and usage of Regions of Interest (RoIs).

Most of RoI-based video coding methods found in literature, e.g., [16–18], do not relate to video coding for machines, but to optimization of human perceptual quality. The main approaches are: usage of neural networks (e.g., [19]), quality control (e.g., [20, 21]) and image size resolution. In the first approach, RoIs are used to construct neural-network-based motion or texture models, e.g., as in [19], which is very loosely related to the scope of VCM considered in this paper.

The second approach employs RoIs for quality control or bit-allocation control. For example, in work [16], a bit allocation scheme based on RoI was, particularly aimed at conversational video encoding. Once the RoI regions are detected, various coding parameters, such as QP, coding modes, the number of reference frames, motion vector accuracy, and motion estimation search range, are adaptively adjusted at the macroblock (MB) level according to the importance of each MB and the target bit rate. Alternatively, in the paper [17], the bit rate across different regions was adapted using

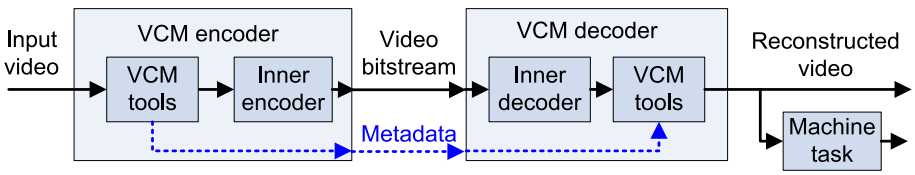

Fig. 1 General architecture of video coding for machines (VCM)

a just-noticeable-difference (JND) model. Similarly, work [18] proposed a foveated JND (FJND) model that integrates visual sensitivity and foveation characteristics of the human visual system (HVS). This approach, although successful for human perception optimization has one disadvantage: information about quality control, e.g., through changes of QP values for each block/macroblock/CTU, has to be transmitted which negatively impacts encoding performance [20]. An alternative approach can be found in work [21], where a saliency map is employed for quality/bit-allocation control. This, however, implies the need to transmit the saliency map to the decoder or infer it from previous frames, which is a disadvantage. Nevertheless, none of the mentioned methods have been applied in the context of VCM.

The third approach to the usage of RoIs for video compression consists of the usage of scaling or retargeting. Retargeting techniques generally involve removing certain areas or applying non-uniform scaling to reduce the overall resolution while preserving the integrity of key image regions. These methods can sometimes lead to exaggerated object sizes and distorted distances, but the most problematic outcomes are unnatural artifacts in the image. Since retargeting is primarily used for image presentation, research in this area mainly focuses on minimizing these distortions. For example, work [22] decomposes images into objects and background. If the background does not fit the target size, the areas originally occupied by objects are inpainted, and the background is scaled. The objects are then reinserted, preserving their spatial arrangement. Objects may also be scaled according to their importance if necessary. Both [22, 23] suggest that simple techniques like image cropping work well when there is a single object or a concentrated group of objects in the image, while uniform scaling is more effective for images with relatively low bandwidth. A combination of these two methods, adapted for sequences, was proposed by [24].

According to papers [23, 25], the main retargeting techniques beyond uniform scaling or cropping include seam carving [26, 27], non-homogeneous warping [28], scale-and-stretch [29, 30], multi-operator methods [31], and shift-map [32].

Recent work in retargeting has primarily focused on optimizing spatial and temporal control, as seen in papers [33–35]. Current approaches often use convolutional neural networks to select the best retargeting method for each image [36, 37], determine optimal control for traditional retargeting methods [38–40], or directly retarget the image [41–43].

The disadvantage of the abovementioned methods is that mostly they were proposed for human perception purposes, sometimes for still images, and not for VCM. Moreover, the mentioned techniques were tested in a variety of different conditions (still image/video coding, different data sets, and different machine tasks for evaluation). In summary, this reduces the suitability of these methods for use in industrial, standardized solutions.

3 MPEG video coding for machines

The need for a standardized solution for Video Coding for Machines has been recognized by ISO/IEC MPEG. In July 2019, the MPEG Video Coding for Machines Ad-Hoc group initiated the development of video coding standards tailored for "highly-efficient video compression and representation for intelligent machine-vision or hybrid

machine/human-vision applications" [44]. It should be noted that in 2021, also JPEG, to be precise JPEG AI called for proposals of learning-based coding standards [45]. In 2021, the MPEG VCM has established initial requirements and use cases related to various application areas, such as surveillance, intelligent transportation, smart city, and smart industry, where increasing use of machine processing algorithms is expected [46, 47]. These areas of use set the stage for developments, where specialized VCM codecs may have key applications. Very wide area of applications and the complexity of the problem led experts to divide the VCM standardization process into two independent tracks: first devoted to visual features compression approaches and the second devoted to classical video compression methods improvement. In the second track, Call for Proposals (CfP) was released in April 2022 [48], with updated requirements and use cases. The CfP concluded in October 2022, yielded numerous responses from prestigious research centers (Alibaba, Institute of Computing Technology, Chinese Academy of Sciences (CAS-ICT), China Telecom, Ericsson, Electronics and Telecommunications Research Institute (ETRI), Nokia, OP Solutions, Tencent, V-Nova) and universities (City University of Hong Kong, China, Hong Kong, Florida Atlantic University (FAU), USA, Konkuk University, Myongji University, Republic of Korea, Poznan University of Technology (PUT), Poland, Wuhan University, Zhejiang University, China [49]. The scientific and technical contributions received led to the initiation of MPEG WG4 activity, where further technological and scientific advancements are pursued through collaboration among researchers and competition between proposed tools. Consequently, current research in this domain revolves around the endeavors of the MPEG VCM group. Therefore, it can be claimed that they represent the state-of-the-art in video coding for machines.

The current architecture of VCM (Fig. 2) includes several key components important for adapting video content for machine-based applications [50]. These components are spatial and temporal resampling, region of interest (RoI) encoding, and bit depth truncation.

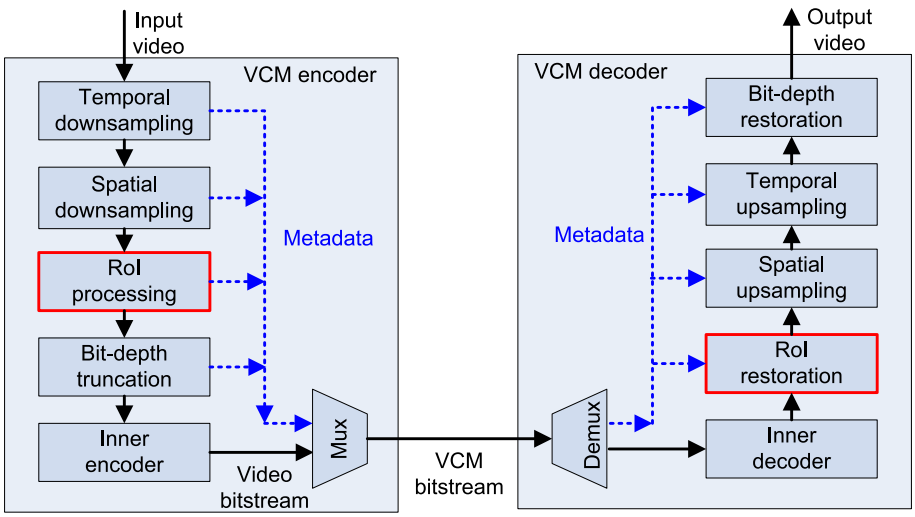

Fig. 2 Current architecture of VCM, with the scope of this paper—RoI-based tools—outlined in red

The spatial resampling reduces the resolution of video frames, striving to achieve a balance between minimizing bit rate and maintaining sufficient detail for effective machine vision recognition. This step allows also to reduce the computational complexity of the whole encoding process.

The temporal resampling modifies the video frame rate by skipping some frames in the encoder and interpolating them in the decoder. In other words, the current encoder may process every second, fourth, or eighth frame from the video sequence. On the decoder side, missing frames are interpolated using dedicated neural networks.

The Region of Interest (RoI) coding prioritizes [51] areas in the video frame relevant to machine-based tasks. This ensures that these areas are encoded with higher precision compared to less relevant areas.

The RoI-based simplification technique adopted to MPEG VCM prior to this paper, which serves as reference in the experiments has been introduced by work [52]. First, the detection of objects is performed resulting in so-called bounding boxes (RoIs). Second, bounding boxes are enlarged by a 20-pixel margin. Next, overlapping boxes or boxes that are close enough to each other are grouped. Finally, tracking of objects is involved to ensure temporal consistency of RoIs. Any information outside RoIs, so-called background, is shaded in grey.

The last tool, the bit depth truncation, adaptively reduces the dynamic range of the luma component samples, e.g., from 10 to 9 bits at the decoder side.

In the end, such processed image or video sequence is fed to the so-called Inner Codec, which may be any video encoder. In this way, VCM aims to optimize video encoding for machine-learning tasks by reducing redundant information at the encoding stage and reconstructing it later in the decoding process. Therefore, the VCM architecture seeks to optimize the efficiency of video encoding, with the hope of enabling efficient use in machine vision processing tasks.

Of course, the VCM standardization process is still ongoing (August 2024), and the so-called cooperation phase is in progress. Therefore, existing tools may be further optimized and improved as well as new tools may be added to the described VCM structure.

4 The proposed technique

The idea of the proposed new technique for Video Coding for Machines is presented in (Fig. 3). The core concept of the proposal is that content within Regions of Interest (RoIs) should be encoded with a high number of pixels, whereas the remaining regions can be downsized and represented with fewer pixels to conserve bitrate. This objective is attained by the usage of RoI detection and retargeting of the image prior to the encoding, which alters the proportions between regions within the image to prioritize RoIs. Consequently, the size of the transmitted image (within the Inner Codec) is different (smaller) than the original size of the input/output image. The original proportions are restored by inverse retargeting in the decoder, which has access to the location of the RoIs, as their bounding boxes are transmitted in the bitstream as metadata (marked by blue arrows in Fig. 1).

The overview of the proposed approach is described below, while the implementation details are shown later in Sect. 5.

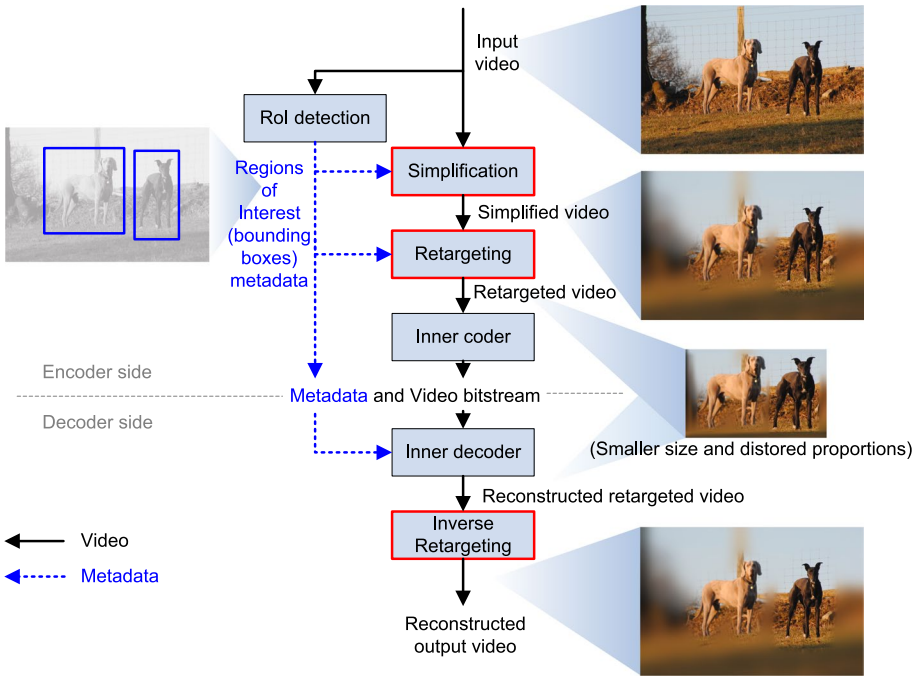

Fig. 3 Idea of the proposed method. Distinctive blocks of the proposal (retargeting and simplification) are outlined in red

- **RoI detection** Regions of Interest are detected at the encoder side prior to the other processing steps. Detection is accomplished using a neural network tailored to the machine vision task to be performed at the decoder. The descriptions of RoIs consist mainly of bounding boxes, but for efficient encoder control, it may also include identification of classes and/or measure of importance level. In general, each RoI is, therefore, assigned a value indicating its spectral redundancy, such as the scaling factor.

Therefore, let us define the scaling factor S_r, representing the down-sampling ratio that can be applied to a given RoI r without loss of important information.

- **Simplification** The regions in video frames are simplified according to their importance represented by scaling factors S_r. The high-importance RoIs are left intact (are not filtered), whereas the regions that are not of interest (outside of RoIs, e.g., background) are low-pass filtered. Regions of negligible importance (e.g., areas of background on peripherals of the image) can be entirely discarded, such as being grayed out. Such a graying-out technique is characteristic of the RoI-based tool currently adopted in VTM [52].

This simplification allows RoIs to be encoded with a higher bitrate (as compared to other regions) and results in improved quality in the decoded video. As a result, it generally enhances the precision of machine vision tasks performed on the decoded video.

- **Retargeting** The proportions of the regions within the image are adjusted depending on their importance and scaling factors S_r. High-importance RoIs are allocated a higher number of pixels, while regions that are not of interest (outside RoIs) are allocated fewer pixels. This adjustment is carried out using a rectangular retargeting grid, created based on the bounding boxes of RoIs, as described below. This grid is used to perform retargeting at the encoder, and inverse retargeting at the decoder.

The retargeting grid is constructed in the exactly same manner in the encoder and in the decoder, on the basis of the positions of RoI bounding boxes, in the following steps (Fig. 4):

1. All edges of the RoI bounding boxes are used to instantiate horizontal and vertical lines, which split the area of the original (input) image into fields demarcated by a rectangular grid (in the coordinate space of the original image).
2. Each field inside the retargeting grid, located at position x, y, is assigned with the scaling factor $FS_{x,y}$, equal to the scaling factor S_c, corresponding to RoI c, collocated to position x, y. In the case of overlapping of RoIs, a smaller value is taken (1):

$$FS_{x,y} = \min_{c \in \{colocatedtox,y\}}(S_c) \tag{1}$$

3. For each column x of fields in the retargeting grid, the minimal column scaling factor CS_y is found (2):

$$CS_x = \min_y\left(FS_{x,y}\right) \tag{2}$$

4. For each row y of fields in the retargeting grid, the minimal row scaling factor RS_y is found as follows (3):

$$RS_y = \min_x\left(FS_{x,y}\right) \tag{3}$$

5. 5. Each field inside retargeting grid, at position x, y, is scaled down, according to CS_x (horizontally) and RS_y (vertically). This yields grid coordinates in the retargeted image.

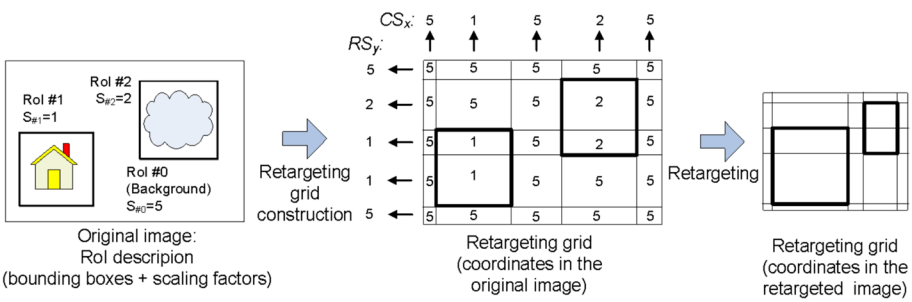

Fig. 4 Construction of the retargeting grid, on the example of a simple image with two RoIs (RoI #1, #2) and a background (RoI #0), assigned with exemplary scaling factors: $S_{\#0} = 5$, $S_{\#1} = 1$, and $S_{\#2} = 2$

Correspondence of the field coordinates (in the original and in the retargeted image) is used to perform image retargeting and inverse retargeting. It can be noted that this involves solely scaling of rectangular regions, which is a simple affine transformation.

5 Implementation

For the sake of experimental verification, the idea of the proposal has been implemented in Video Coding for Machines Reference Software (VCM-RS) [53]. In particular, implementation has been done on top of the existing RoI-based tool [52], which involves RoI detection and graying-out of background regions. To provide a fair evaluation of the gains of the proposal, the already existing processing steps have been left unchanged. In particular:

- RoI detection. The same neural networks as in the reference technique [52] are used by default, i.e., YOLO [54] for object tracking, and Detectron2 (Faster R-CNN) [55] for object tracking. In the results section, we also present alternative results with exchanged (alternative) detection networks.
- RoI accumulation. Inside a particular group of pictures (GOP), detected objects are accumulated in time to create regions of interest that are broader and more consistent in time (Fig. 5).

Because the VCM framework does not include object class indication or level of importance of detected RoIs, in the current version of the tool, all RoI are assigned with scale factors $S_r = 1$.

- All already adopted VCM techniques and tools enabled, e.g., temporal resampling and bit truncation tool.
- VVC [3, 4] was used as the Inner Codec.

An example of performed processing is illustrated in Fig. 6. Simplification is performed over closed areas, found between accumulated RoI regions. Therefore, the closing operation can occur between any RoIs detected in any frames within the accumulation period, e.g., RoIs related to the same object but in different frames, or between different objects. Closing is done for the maximal distance of 160 pixels, which is attained with a closing filter (dilation followed by erosion) with a rectangular kernel of the size of 321 pixels. For simplification, a simple low-pass FIR filter is used, with 25 coefficients and a normalized bandwidth of $B = 0.01$, designed using the Lanchos approximation [56].

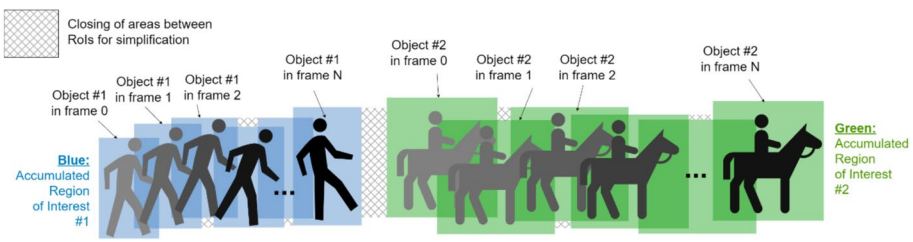

Fig. 5 RoI accumulation and closing for the area of simplification

Fig. 6 Exemplary image from TVD-01_1 video. The images are aligned so that RoIs are located as in the original image: **a** original frame, **b** reference [52] method, **c** proposed: simplified image, and **d** proposed: image retargeted from the original resolution 1920×1080 to 1920×448

In general, each RoI class can have a specified optimal size for representation, e.g., allowing high performance in machine vision tasks. This size differs per class, e.g., more pixels are required to detect a cat (complex fur) than a hippo (smooth skin). If the size of the detected RoI is greater than the optimal size for representation for a given class, its resolution can be reduced seamlessly. In the current implementation, however, all RoIs are initially assigned with scale factor $S_r = 1$, which corresponds to the original

resolution of RoI. This, however, is further modified by image resolution adjustment as described below.

In the current implementation of the encoder, the retargeting possibilities are analyzed independently for each "accumulation period", which corresponds to the intra period (to meet Random Access requirements). Therefore, for each such period of frames, different retargeted resolution is found. To encode the whole video sequence using a common retargeted resolution, the biggest retargeted resolution is found among all "accumulation periods". In addition, this resolution is aligned (rounded up) to 64×64, which is the CTU size used in the Inner Codec. Nevertheless, the common retargeted resolution is limited not to exceed the original resolution. Because the common retargeted resolution may be bigger than a particular retargeted resolution for a particular accumulation/intra-period, the retargeting grid is adjusted (scaled) to fit the common retargeted resolution.

In the example presented in Fig. 6, the resolution of the original sequence 1920×1080 was retargeted to 1920×448. This yields in significant reduction (more than 50%) of the number of pixels that are processed by the Inner Codec (Figs. 1, 3), which, as an additional benefit of the proposal, may lead to a proportional reduction of computational complexity of the encoder and the decoder.

6 Experiments/evaluation

6.1 Evaluation data sets

The presented method, as well as all the others applied for Video Coding for Machines (VTM), is required to be evaluated according to Common Test Conditions (CTC) [57]. The CTC precisely defines conditions, steps, and restrictions for evaluation within VCM by accurate definition of the testing environment (evaluation pipeline) and rules for comparison with the anchor technique. This detailed procedure outcomes in the consistent evaluation of the tested method for two machine learning tasks: object detection/segmentation and tracking. The CTC restricts the use of certain models for each machine task. In the object detection task, the Faster R-CNN [55] X101-FPN is employed, which is a part of Detectron 2 [58] developed by Facebook AI Research. For the tracking machine task, the JDE-1088×608 [59] network is employed.

Evaluation of our method, according to CTC, was performed in three configurations: all-intra (AI), random access (RA), and low delay (LD). Each configuration aims to test the method in different aspects that are important in video coding for machines. The AI configuration, based only on the currently processed frame, is dedicated to the use case, where the video sequence must be processed rapidly. It is important to mention that it is the only scenario, where task performance may be evaluated on images. The evaluation in the RA configuration is meant to check the method in situations, where the focus is put on quick access to the encoded frame. The LD configuration is designed to investigate the method's efficiency when latency restrictions are imposed on coding.

The CTC precisely defines the data sets that should be used in the evaluation of each machine-learning task. In Table 1, we presented a brief description of data sets. The CTC defines two mandatory data sets for evaluation: the SFU-HW-objects-v1 (SFU) [60] data set and the tencent video data set (TVD) [61]. The first data set is composed of sequences known from previous MPEG standardization groups: JVC-VC and JVET. The SFU data set contains 14 sequences, each assigned to classes {A, B, C, D, O} based on

Table 1 Data sets used in the evaluation, according to CTC [57]

Data set name	Class	Number of sequences	Resolution	Frame rate	Bit depth	Machine task
SFU [60]	A	1	2560×1600	30	8	Object detection/segmentation
	B	4	1920×1080	24, 50 or 60	8	
	C	4	832×480	30, 50 or 60	8	
	D	4	416×240	30, 50 or 60	8	
	O	1	1920×1080	24	8	
TVD [61]	–	7	1920×1080	50	8 or 10	Object tracking

resolution. The O class (1 sequence) is not mandatory in evaluation. One should mention that the frame rate of sequences within SFU is not the same. The second mandatory data set, the TVD, consists of 3 sequences in 1920×1080 resolution divided into 7 parts: TVD1-01 to TVD-03-3 (Table 3). Each sequence has a frame rate of 50, but the bit depth differs.

For each sequence, the CTC precisely indicates the number of frames to be encoded and six Quantization Parameters (QPs). In addition, the evaluation process should be performed on specified sequence fragments. The encoding should be done on sequences in YUV420p format. For the TVD data set, the conversion should be performed using ffmpeg software [62].

6.2 Evaluation metrics

In VCM, traditional quality assessments have given way to novel evaluation metrics designed to measure the efficiency of encoding and decoding techniques for machine tasks. The mean Average Precision (mAP) serves as a key metric for assessing how well an object detection algorithm can pinpoint and correctly label various objects, focusing on its success rate and accuracy [63]. Meanwhile, the Multiple Object Tracking Accuracy (MOTA) metric evaluates the effectiveness of tracking algorithms in consistently recognizing the same objects over successive frames, taking into account challenges such as overlooked detections, incorrect positives, and errors in object identification [64].

The Bjøntegaard Delta rate (BD-rate) model, a benchmark for evaluating coding efficiency, has significantly influenced the evolution of image and video coding standards across key standardization bodies, including the Joint Photographic Experts Group (JPEG), the Moving Picture Experts Group (MPEG), and the Video Coding Experts Group (VCEG). Initially introduced in 2001 [65] and subsequently refined [66], the BD-rate model has undergone enhancements to address complex coding scenarios [67]. The most recent iteration [68] introduces capabilities for analyzing beyond four operating points and specifies calculations within determined ranges. Given that the performance measured in machine tasks is no longer PSNR but mAP and MOTA, we have substituted mAP and MOTA for PSNR in calculating the BD-rate.

A long-standing issue that has been raised during the VCM investigation relates to the non-monotonic behavior of these R-D data. It is observed that the metric values used by VCM in terms of mAP and MOTA are relatively noisy.

The noise of metric values boils down to several major aspects: (1) measurement error due to limited accuracy of the object detection or object tracking methods used in the

VCM framework, (2) cross-platform discrepancy of machine task results, which may happen for the same implementation and bit-exact bitstream, due to different hardware architectures or floating-point precision, and (3) the intrinsic averaging mechanism performed in mAP and MOTA calculations.

During the exploration of Video Coding for Machines (VCM), a recurrent challenge has been the non-linear response observed in Rate-Distortion (R-D) data, particularly manifested through the variability in mAP and MOTA metrics. This variability, or "noise," in the metrics can be attributed to several key factors:

1. Errors in measurement stemming from the inherent limitations in the accuracy of object detection and tracking methodologies within the VCM framework;
2. Disparities in machine task outcomes across different platforms, even with identical implementation and bitstream, attributable to variations in hardware architecture or floating-point computation precision; and
3. The fundamental nature of the averaging processes employed in calculating mAP and MOTA metrics. These elements collectively contribute to the observed fluctuations, underlining the complexity of accurately assessing performance within the VCM paradigm.

Aligned with the VCM CTC guidelines, we've employed a curve-fitting technique to manage the irregularities observed in metric values across different scenarios. Opting for a cubic polynomial function offers the adaptability needed to accurately represent the R-D curve, particularly in the mid to high-quality spectrums, described as (4):

$$f(x) = a + bx + cx^2 + dx^3 \tag{4}$$

where a, b, c, and d signify the coefficients of the cubic equation. To refine the curve's alignment with R-D characteristics specific to machine tasks, we've imposed additional conditions. We ensure the curve's first derivative is non-negative to maintain a monotonically increasing trend. Moreover, by setting the second derivative to be non-positive, we preserve the function's concavity, allowing it to smoothly transition between linear and saturated regions. This tailored curve-fitting method has been applied to all our data, resulting in a recalibrated data set.

6.3 Coding performance

In this section, we presented the evaluation of our method according to the procedure described in CTC. For the default configuration of RoI detection, we have used the same neural networks as in the reference technique [52], i.e., YOLO [54] for object tracking, and Detectron2 (Faster R-CNN) [55] for object tracking. For the sake of completeness of the results, we also present results for the alternative configuration, where the RoI detection networks are exchanged.

First, we collated results for the object detection task in the default configuration, where Faster R-CNN is used, for all encoding scenarios. In Table 2, we presented results for bitrate reduction (BD-RATE) while preserving the same (constant) mAP quality. Please note that the highlighted "All" row, reports BD-Rate results averaged for all individual sequences in the data set, and not the mathematical average over presented

Table 2 Object detection task—comparison of Bitrate Reduction (BD-Rate), for All Intra (AI), Low Delay (LD) and Random Access (RA) Scenarios, and different RoI detection networks Faster R-CNN [55] X101-FPN and Detectron2 [58]

Case	End-to-end BD-rate [%] over mAP					
	Faster R-CNN (default as in [52] and [57])			Detectron 2 (alternative)		
	AI	LD	RA	AI	LD	RA
Class A	−18.00	−9.57	−29.41	−12.43	−7.32	−22.74
Class B	−26.32	3.49	−10.99	−13.93	−0.66	−5.43
Class C	−22.34	−6.48	−4.84	−12.64	−2.58	−3.33
Class D	−10.02	−6.15	5.61	−4.55	−2.75	2.43
All	**−19.44**	**−3.55**	**−5.41**	**−10.59**	**−3.13**	**−2.33**
Class O (not mandatory)	−21.91	1.76	−20.82	−19.88	2.43	−17.55

Table 3 Object tracking task—comparison of Bitrate Reduction (BD-Rate), for All Intra (AI), Low Delay (LD) and Random Access (RA) Scenarios, and different RoI detection networks: Detectron2 [58] and Faster R-CNN [55] X101-FPN

Case	End-to-End BD-Rate [%] over MOTA					
	Detectron 2 (default as in [52] and [57])			Faster R-CNN (alternative)		
	AI	LD	RA	AI	LD	RA
TVD-01-1	−61.77	−32.36	−57.58	−47.66	−28.53	−55.08
TVD-01-2	−65.02	−14.19	−37.51	−54.81	−8.89	−30.13
TVD-01-3	−85.80	−31.16	−59.10	−69.64	−27.85	−51.97
TVD-02-1	−33.09	16.81	17.15	−31.06	21.52	24.53
TVD-03-1	−50.64	−6.31	−8.22	−36.85	−4.78	2.62
TVD-03-2	−42.20	−2.81	−17.83	−38.99	−0.17	−11.53
TVD-03-3	−60.93	−8.74	−20.86	−47.37	−8.14	−18.29
All	**−57.06**	**−11.25**	**−26.28**	**−46.63**	**−8.12**	**−19.98**

class-wise BD-Rate results. In the All Intra (AI) encoding scenario our method yields the best results of a 19.44% bitrate reduction (over constant mAP) for the average calculated over all sequences. In this scenario, we report improvement versus the anchor method for all data set classes. In configurations, where Inter-coding is utilized—Random Access (RA) and Low Delay (LD) our method still delivers improvements. Results in the RA configuration evaluation, averaged over all sequences, indicate a 5.41% bitrate reduction (over constant mAP). For the LD scenario, a 3.55 bitrate reduction over const mAP was observed. Analysis of class-specific results showed, that in two cases (the LD in class B and the RA in class D) our method reported worse results, but they do not exceed a 6% bitrate (over constant mAP).

It can be seen that usage of the alternative network for RoI detection for the object detection task (Detectron 2) yields considerably worse results. This can be explained by the mismatch between the detection network and the machine task in the decoder.

The evaluation results for the object tracking machine task are provided in Table 3. The results are presented as bitrate reduction BD-RATE over the same task performance, expressed as MOTA.

Similarly, as before, the best results are observed in the default RoI detection network configuration, which for the object tracking task is Detectron 2. Here we observe even bigger improvements in efficiency compared to anchor. In the AI scenario, our method reported a 57.05% smaller bitrate on average (over all sequences) for the same object tracking result. What is more, in the rest of the CTC configuration we observe improvements too. Evaluation in the RA scenario demonstrated an average 26.28% reduction bitrate (over constant MOTA). The smallest improvements are delivered in the LD configuration, where usage of our method benefits with 11.25% bitstream reduction (over preserved MOTA quality) on average over all sequences. Here we note that only one sequence: TVD-02-1 was found as encoded worse than the anchor.

Similarly, as in the case of object detection, usage of the alternative RoI detection network also brings slight deficiency. In the case of the object detection task, the alternative RoI detection network is Detectron 2. Just as before, this can be explained by the mismatch between the detection network and the machine task in the decoder.

In Tables 2 and 3, it can be observed that the coding efficiency on different classes varies significantly, especially for LD and RA scenarios. For example, in the case of object detection, Table 2, default RoI detection network, for the LD scenario BD-Rates vary between −9.57% gain to 3.59% loss. In the RA scenario BD-Rates vary between −29.41% gain to 5.61% loss. Similarly, in Table 3, object tracking machine task, default RoI detection network, for the LD scenario BD-Rates vary between −32.36% gain to 16.81% loss, and in the RA scenario: between −59.10% gain to 17.15% loss. This can be attributed to the high dependence of the proposed method on the distribution of Regions of Interest in the video. Because the retargeting is performed in a rectangular grid, particular unfortunate placement of objects may disallow efficient stretching of RoI areas, thus disabling compression gain while still having cost related to signalling of RoIs. This could be investigated in our future research, where we intend to propose a more efficient RoI detection and optimization algorithm.

7 Conclusions

This paper describes the retargeting coding tool [69] that was proposed and accepted for the prospective ISO/IEC international standard for Video Coding for machines. The tool has evolved from the RoI-based preprocessing and retargeting tools proposed originally as a response to CfP by Poznan University of Technology [70, 71]. This tool has been later redesigned, improved and merged with ideas and implementations already adopted to VCM-RS, provided by the other participants in the VCM group, i.e., ETRI and Myongji University [50].

Extensive experimental validation was performed both by the authors and the cross-checkers from the MPEG VCM experts group [72–74]. The provided results demonstrated that the respective extension of the VCM Test Model resulted in a significant improvement in coding performance. For most cases, average BD-Rate bitrate reductions of 3–57% have been calculated using the monotonic RD curves derived from the curve fitting method specified in the CTC, i.e., these bitrate reductions are demonstrated by keeping roughly unchanged the average mean precision of object detection and tracking. The thoughtful consideration within the MPEG group resulted in the adoption of the proposed tool for the upcoming VCM video coding technology.

In addition, due to the reduced size of the frames encoded by the Inner Codec, the encoding and decoding times are vastly reduced. We avoid citations of the exact numbers for these reductions as the time measurement methodology settled within the VCM group is very vulnerable to the computing environment. Further works are planned with the aim of increasing the flexibility and coding efficiency of the tool by allowing more specific values of the retargeting ratios.

Abbreviations

AVC	Advanced video coding
BD-rate	Bjøntegaard Delta rate
CfP	Call for proposals
CTC	Common test conditions
HEVC	High efficiency video coding
JPEG	Joint photographic experts group
LD	Low delay
mAP	Mean average precision
MOTA	Multiple object tracking accuracy
MPEG	Moving picture experts group
QP	Quantization parameter
RA	Random access
RoI	Region of interest
VCEG	Video coding experts group
VCM	Video coding for machines
VCM-RS	Video Coding for Machines Reference Software
VTM	Versatile video coding test model
VVC	Versatile video coding

Acknowledgements
Not applicable

Author contributions
SR invented the proposed method. All authors participated in the design of the proposed method, implementation, performance measures, experiments, and writing of the manuscript. All authors read and approved the final manuscript.

Funding
Not applicable.

Data availability
The source code for the proposal is available for MPEG experts on the MPEG reference software repository: https://git.mpeg.expert/MPEG/Video/VCM/VCM-RS

Declarations

Competing interests
Not applicable.

Received: 1 March 2024 Accepted: 29 August 2025
Published online: 24 October 2025

References

1. G.J. Sullivan, J. Ohm, W.J. Han, T. Wiegand, Overview of the high efficiency video coding (HEVC) standard. IEEE Trans. Circuits Syst. Video Technol. **22**(12), 1649–1668 (2012)
2. ITU-T Rec. H.265 | ISO/IEC IS 23008-2, High efficiency coding and media delivery in heterogeneous environment – Part 2: High efficiency video coding
3. J. Chen, Y. Ye, S. Kim, Algorithm description for Versatile Video Coding and Test Model 3 (VTM3). Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, Doc. JVET L1002, Macao, October 2018
4. ISO/IEC DIS 23090–3 (2020) / ITU-T Recommendation H.266 (08/2020), Versatile video coding
5. B. Bross, Y.-K. Wang, Y. Ye, S. Liu, J. Chen, G.J. Sullivan, J.-R. Ohm, Overview of the versatile video coding (VVC) standard and its applications. IEEE Trans. Circuits Syst. Video Technol. **31**(10), 3736–3764 (2021)
6. T. Wiegand, G.J. Sullivan, G. Bjontegaard, A. Luthra, Overview of the h. 264/AVC video coding standard. IEEE Trans. Circuits Syst. Video Technol. **13**(7), 560–576 (2003)
7. D. Xu, R. Chellappa, L. Van Gool et al., Guest editorial: special issue on deep learning for video analysis and compression. Int. J. Comput. Vis. **129**, 3171–3173 (2021). https://doi.org/10.1007/s11263-021-01530-3

8. N. Xu, W. Lin, X. Lu, Y. Wei, *Video object tracking: tasks, datasets, and methods. Springer synthesis lectures on computer vision (SLCV)* (Springer, Cham, 2024)
9. S. Ma, X. Zhang, S. Wang, X. Zhang, C. Jia, S. Wang, Joint feature and texture coding: toward smart video representation via front-end intelligence. IEEE Trans. Circuits Syst. Video Technol. **29**(10), 3095–3105 (2018)
10. Q. Zhang, S. Wang, X. Zhang, S. Ma, W. Gao, Just recognizable distortion for machine vision oriented image and video coding. Int. J. Comput. Vis. **129**(10), 2889–2906 (2021)
11. J. Chao, E. Steinbach, Keypoint encoding for improved feature extraction from compressed video at low bitrates. IEEE Trans. Multimedia **18**(1), 25–39 (2016)
12. L. Galteri, M. Bertini, L. Seidenari, A. Del Bimbo, Video compression for object detection algorithms. 24th International Conference on Pattern Recognition (ICPR), (2018), p. 3007–3012
13. L. Duan, J. Liu, W. Yang, T. Huang, W. Gao, Video coding for machines: a paradigm of collaborative compression and intelligent analytics. IEEE Trans. Image Process. **29**, 8680–8695 (2020)
14. K. Fischer, F. Brand, C. Herglotz, A. Kaup, Video Coding for Machines with Feature-Based Rate-Distortion Optimization. 22nd International Workshop on Multimedia Signal Processing (MMSP), (2020)
15. Y. Lee, S. Kim, K. Yoon, H. Lim, S. Kwak, H.-G. Choo, Machine-attention-based Video Coding for Machines. 2023 IEEE International Conference on Image Processing (ICIP), (2023), p. 2700–2704
16. Y. Liu, Z. Li, Y.C. Soh, Region-of-interest based resource allocation for conversational video communications of H.264/AVC. IEEE Trans. Circuits Syst. Video Technol. **18**(1), 134–139 (2008)
17. X. Yang, W. Lin, Z. Lu, X. Lin, S. Rahardja, E.P. Ong, S. Yao, Rate control for videophone using local perceptual cues. IEEE Trans. Circuits Syst. Video Technol. **15**(4), 496–507 (2005)
18. Z. Chen, C. Guillemot, Perceptually-friendly H.264/AVC video coding based on foveated just-noticeable-distortion model. IEEE Trans. Circuits Syst. Video Technol. **20**(6), 806–819 (2010)
19. M. Bosch, F. Zhu, E.J. Delp, Segmentation-based video compression using texture and motion models. IEEE J. Sel. Top. Signal Process. **5**(7), 1366–1377 (2011)
20. C. Cai, L. Chen, X. Zhang, Z. Gao, End-to-end optimized ROI image compression. IEEE Trans. Image Process. **29**, 3442–3457 (2020). https://doi.org/10.1109/TIP.2019.2960869
21. H. Hadizadeh, I.V. Bajić, Saliency-aware video compression. IEEE Trans. Image Process. **23**(1), 19–33 (2014). https://doi.org/10.1109/TIP.2013.2282897
22. V. Setlur, S. Takagi, R. Raskar, M. Gleicher, B. Gooch, Automatic image retargeting. Proceedings of the 4th International Conference on Mobile and Ubiquitous Multimedia, (2005)
23. M. Rubinstein, D. Gutierrez, O. Sorkine, A. Shamir, A comparative study of image retargeting. ACM Trans. Graph. **29**(6), 160:1-160:10 (2010)
24. F. Liu, M. Gleicher, Video retargeting: Automating pan and scan, (2006), p. 241–250. https://doi.org/10.1145/1180639.1180702.
25. R. Kharsa, R. Alzoubi, M. Alsmirat M. Al-Ayyoub, Image Retargeting Techniques Identification Using Supervised Deep Learning. 2023 Fourth International Conference on Intelligent Data Science Technologies and Applications (IDSTA), Kuwai, Kuwait, (2023), p. 15–20, https://doi.org/10.1109/IDSTA58916.2023.10317851.
26. S. Avidan, A. Shamir, Seam carving for content-aware image resizing. SIGGRAPH (2007). https://doi.org/10.1145/1276377.1276390
27. M. Rubinstein, A. Shamir, S. Avidan, Improved seam carving for video retargeting. ACM Trans. Graph. **27**(3), 1–9 (2008). https://doi.org/10.1145/1360612.1360615
28. L. Wolf, M. Guttmann, D. Cohen-Or, Non-homogeneous Content-driven Video-retargeting. Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on. 1, (2007), p. 1–6. https://doi.org/10.1109/ICCV.2007.4409010.
29. Y.-S. Wang, C.-L. Tai, O. Sorkine, T.-Y. Lee, Optimized scale-and-stretch for image resizing. ACM Trans. Graph. **27**(5), Article 118 (2008). https://doi.org/10.1145/1409060.1409071
30. D. Panozzo, O. Weber, O. Sorkine, Robust image retargeting via axis-aligned deformation. Comput. Graph. Forum. **31**, 229–236 (2012). https://doi.org/10.1111/j.1467-8659.2012.03001.x
31. M. Rubinstein, A. Shamir, S. Avidan, Multi-operator media retargeting. ACM Trans. Graph. **28**(3), Article 23 (2009). https://doi.org/10.1145/1531326.1531329
32. Y. Pritch, E. Kav-Venaki, S. Peleg, Shift-Map Image Editing. Proceedings of the IEEE International Conference on Computer Vision, (2009),p. 151–158. https://doi.org/10.1109/ICCV.2009.5459159
33. Z. Zhang, B. Kang, H. Li, Improved seam carving for content-aware image retargeting. 2013 IEEE Asia Pacific Conference on Postgraduate Research in Microelectronics and Electronics (PrimeAsia), Visakhapatnam, India, (2013), pp. 254–257, https://doi.org/10.1109/PrimeAsia.2013.6731216
34. H.C. Hsin, Saliency histogram equalisation and its application to image resizing. IET Image Process. **10**(10), 787–798 (2016)
35. H. Kaur, S. Kour, D. Sen, Video retargeting through spatio-temporal seam carving using Kalman filter. IET Image Proc. **13**, 1862–1871 (2019). https://doi.org/10.1049/iet-ipr.2019.0236
36. Y. Zhou, Z. Chen, W. Li, Weakly supervised reinforced multi-operator image retargeting. IEEE Trans. Circuits Syst. Video Technol. **31**(1), 126–139 (2020)
37. R. Kharsa, R. Alzoubi, M. Alsmirat, M. Al-Ayyoub, Image Retargeting Techniques Identification Using Supervised Deep Learning. In 2023 Fourth International Conference on Intelligent Data Science Technologies and Applications (IDSTA), (2023, October), p 15–20. IEEE.
38. E. Song, M. Lee, S. Lee, CarvingNet: content-guided seam carving using deep convolution neural network. IEEE Access **7**, 284–292 (2018)
39. J. Wu, R. Xie, L. Song, B. Liu, Deep feature guided image retargeting. In 2019 IEEE Visual Communications and Image Processing (VCIP), (2019, December), pp. 1–4. IEEE.
40. M. Ahmadi, N. Karimi, S. Samavi, Context-aware saliency detection for image retargeting using convolutional neural networks. Multimedia Tools Appl. **80**(8), 11917–11941 (2021)
41. D. Cho, J. Park, T.H. Oh, Y-W. Tai, I. Kweon, Weakly- and Self-Supervised Learning for Content-Aware Deep Image Retargeting, (2017), p. 4568–4577. https://doi.org/10.1109/ICCV.2017.488

42. W. Tan, B. Yan, C. Lin, X. Niu, Cycle-ir: deep cyclic image retargeting. IEEE Trans. Multimedia **22**(7), 1730–1743 (2019)
43. Y. Mei, X. Guo, D. Sun, G. Pan, J. Zhang, Deep Supervised Image Retargeting. 2021 IEEE International Conference on Multimedia and Expo (ICME), Shenzhen, China, (2021), p. 1–6, https://doi.org/10.1109/ICME51207.2021.9428129
44. ISO/IEC, Conclusions of 127th meeting. ISO/IEC JTC 1/SC 29/WG 11, MPEG doc. N18540, July 2019
45. J. Ascenso, E. Upenik, White Paper on JPEG AI Scope and Framework. ISO/IEC JTC 1/SC 29/WG1, MPEG doc. N90049, 2021
46. ISO/IEC JTC1/SC29/WG2, Use cases and requirements for Video Coding for Machines. MPEG doc. N18, October 2020
47. ISO/IEC JTC1/SC29/WG2, Use cases and requirements for Video Coding for Machines. MPEG doc. N0043, January 2021
48. ISO/IEC JTC 1/SC 29/WG 2, Call for Proposals for Video Coding for Machines. MPEG doc. N191, April 2022
49. ISO/IEC JTC 1/SC 29/WG 2, CfP response report for Video Coding for Machines. MPEG doc. N248, October 2022
50. ISO/IEC JTC 1/SC 29/WG 4, Algorithm description of tools in VCM reference software. MPEG doc. N418, December 2023
51. H. Chen, Y. Xu, Video Coding for Machines Based on Motion Assisted Saliency Analysis. Lecture Notes in Computer Science book series, Springer LNCS, **14357**, (2023)
52. S.-K. Kim, M. H. Jeong, J. Y. Lee, H.-K. Lee, H.-G. Choo, S.-H. Jung, [VCM] CfP response: Region-of-Interest based video coding for machine. ISO/IEC JTC1/SC29/WG2 m60758, October 2022
53. Video Coding for Machines Reference Software, available for MPEG experts: https://git.mpeg.expert/MPEG/Video/VCM/VCM-RS
54. J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You Only Look Once: Unified, Real-Time Object Detection. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), (2016), p. 779–788
55. S. Ren, K. He, R. Girshick et al., Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. **39**(6), 1137–1149 (2016)
56. C.E. Duchon, Lanczos filtering in one and two dimensions. J. Appl. Meteorol. **18**(8), 1016–1022 (1979)
57. ISO/IEC JTC 1/SC 29/WG 04, Common test conditions for video coding for machines. MPEG doc. N419, October 2023
58. Y. Wu, A. Kirillov, F. Massa, et al. Detectron2, https://github.com/facebookresearch/detectron2
59. Z. Wang, L. Zheng, Y. Liu, et al. Towards real-time multi-object tracking. In European Conference on Computer Vision (ECCV), (2020), p. 107–122
60. H. Choi, E. Hosseini, S. R. Alvar, R. A. Cohen, I. V. Bajić, A. Karabutov, Z. Yin, E. Alshina, [VCM] Object labelled dataset on raw video sequences. ISO/IEC JTC1/SC29/WG11 MPEG doc. m54737, July 2020.
61. X. Xu, S. Liu, Z. Li, A Video Dataset for Learning-based Visual Data Compression and Analysis. In 2021 International Conference on Visual Communications and Image Processing (VCIP), Dec. 2021.
62. S. Tomar, Converting video formats with FFmpeg. Linux J. **2006**(146), 10 (2006)
63. M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, A. Zisserman, The PASCAL visual object classes (VOC) challenge. Int. J. Comput. Vis. **88**(2), 303–338 (2010). https://doi.org/10.1007/s11263-009-0275-4
64. K. Bernardin, R. Stiefelhagen, Evaluating multiple object tracking performance: the CLEAR MOT metrics. EURASIP J. Image Video Process. **2008**(1), 1–10 (2008)
65. G. Bjøntegaard, Calculation of average PSNR differences between RD-curves. ITU SG16 Doc. VCEG-M33, (2001)
66. G. Bjøntegaard, Improvements of the BD-PSNR model. ITU-T SG16 Q.6 document VCEG-AI11, (2023)
67. S. Akramullah, *Video quality metrics. In Digital video concepts, methods, and metrics* (Apress, Berkeley, 2014)
68. A. M. Tourapis, D. Singer, Y. Su, K. Mammou, Bd-rate/BD-PSNR excel extensions. ISO/IEC JTC1/SC29/WG11 M41482, (2017)
69. S. Różek, O. Stankiewicz, S. Maćkowiak, T. Grajek, M. Wawrzyniak, J. Stankowski, M. Lorkiewicz, D. Cywiński, J. Szekiełda, M. Domański, [VCM] Improved RoI preprocessing and retargeting for VCM. ISO/IEC JTC1/SC29/WG4, MPEG doc. m66523, (January, 2024)
70. M. Domański, O. Stankiewicz, S. Maćkowiak, S. Różek, T. Grajek, J. Szekiełda, D. Cywiński, J. Siejak, [VCM] Poznań University of Technology Proposals A and B in response to CfP on Video Coding for Machines. ISO/IEC JTC1/SC29/WG4, MPEG doc. m61519, (October, 2022)
71. S. Różek, O. Stankiewicz, S. Maćkowiak and M. Domański, Video Coding for Machines using Object Analysis and Standard Video Codecs. 2023 IEEE International Conference on Visual Communications and Image Processing (VCIP), Jeju, Republic of Korea, (2023), p. 1–5
72. D. Ding [VCM] crosscheck of m66523. ISO/IEC JTC1/SC29WG4 MPEG doc. m66778, (January, 2024)
73. Q. Li Fang, H. Wang, Y. Zhang (China Telecom), [VCM] Cross-check of m66523. ISO/IEC JTC1/SC29WG4 MPEG doc. m66769, (January, 2024)
74. H. Yang, S. Wang, C. Lin, [VCM] Crosscheck report for m66523. ISO/IEC JTC1/SC29WG4 MPEG doc. m66907, (January, 2024)

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.