**** **ISO/IEC JTC 1/SC 29/WG 2 N00215**

**ISO/IEC JTC 1/SC 29/WG 2**

**MPEG Technical requirements   
Convenorship: SFS (Finland)**

**Document type:** Output Document

**Title: Call for Evidence for Video Coding for Machines**

**Status:** Approved

**Date of document:** 2022-07-22

**Source:** ISO/IEC JTC 1/SC 29/WG 2

**Expected action:** none

**Action due date:** none

**No. of pages:** 19 (Including the cover page) + 1 XLS (reporting template)

**Email of Convenor:** igor.curcio@nokia.com

**Committee URL:** [**https://sd.iso.org/documents/ui/#!/browse/iso/iso-iec-jtc-1/iso-iec-jtc-1-sc-29/iso-iec-jtc-1-sc-29-wg-2**](https://sd.iso.org/documents/ui/#!/browse/iso/iso-iec-jtc-1/iso-iec-jtc-1-sc-29/iso-iec-jtc-1-sc-29-wg-2)

**INTERNATIONAL ORGANISATION FOR STANDARDISATION**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

**ISO/IEC JTC 1/SC 29/WG 2**

**MPEG TECHNICAL REQUIREMENTS**

**ISO/IEC JTC 1/SC 29/WG 2 N00215**

**Online – July 2022**

**Title: Call for Evidence for Video Coding for Machines**

**Source: WG 2 MPEG Technical requirements**

**Status: Approved**

**Serial number: 21818**

**Editor(s): Yuan Zhang (China Telecom)**

**Chris Rosewarne (Canon)**

**Shan Liu (Tencent)**

**Christopher Hollmann (Ericsson)**

**Table of Contents**

[1 Introduction 5](#_Toc109420473)

[2 Who may Participate? 5](#_Toc109420474)

[3 Code of Conduct and Rules of Engagement 5](#_Toc109420475)

[4 Source Code and IPR 5](#_Toc109420476)

[5 Definitions 6](#_Toc109420477)

[6 Documents of CfE Package 6](#_Toc109420478)

[7 Submission Process 6](#_Toc109420479)

[7.1 CfE Timeline Table 6](#_Toc109420480)

[7.2 Register your participation 6](#_Toc109420481)

[7.3 Mandatory Equipment, Software, and Data Components 7](#_Toc109420482)

[7.4 Access to Test Material 7](#_Toc109420483)

[7.5 Conduct Objective Evaluations 7](#_Toc109420484)

[7.6 Details of the Submission 8](#_Toc109420485)

[7.6.1 Target rate points 8](#_Toc109420486)

[7.6.2 Informative reference point 10](#_Toc109420588)

[7.7 Submission of response documentation 11](#_Toc109420589)

[7.8 Evaluate CfE Submissions 12](#_Toc109420590)

[8 Call Administrator 12](#_Toc109420591)

[9 Email reflector 13](#_Toc109420592)

[10 References 13](#_Toc109420593)

[Appendix A: Detailed description of test datasets 14](#_Toc109420594)

[**A.1 Dataset 1: Tencent Video Dataset (TVD)** 14](#_Toc109420595)

[**A.2 Dataset 2: OpenImages v6** 14](#_Toc109420596)

[Appendix B: Anchor metrics 15](#_Toc109420597)

[B.1 Bitrate measurement 15](#_Toc109420598)

[B.2 Task: Object Tracking 15](#_Toc109420599)

[B.3 Task: Object segmentation / Object detection 15](#_Toc109420600)

[B.4 Runtime Measurement 16](#_Toc109420601)

[Appendix C: Inference and Training information 17](#_Toc109420602)

[C.1 Inference information 17](#_Toc109420603)

[C.2 Training information 17](#_Toc109420604)

[Appendix D Anchor generation environment 18](#_Toc109420605)

[D.1 Object Tracking on TVD (feature anchor) 18](#_Toc109420606)

[D.2 Instance Segmentation on OpenImages (feature anchor) 18](#_Toc109420607)

[D.3 Object Detection on OpenImages (feature anchor) 18](#_Toc109420608)

[D.4 Object Tracking on TVD (informative video anchor) 19](#_Toc109420609)

[D.5 Instance Segmentation / object detection on OpenImages (informative anchor) 19](#_Toc109420610)

**Table of Tables**

[Table 1. CfE Timeline 6](#_Toc109420611)

[Table 2. Rate points for TVD videos for object tracking (overall) 8](#_Toc109420612)

[Table 3. Rate points for TVD videos for object tracking (per-sequence) 8](#_Toc109420613)

[Table 4. Rate points of instance segmentation on OpenImages dataset 9](#_Toc109420614)

[Table 5. Rate points of (optional) object detection on OpenImages dataset 9](#_Toc109420615)

[Table 6. Informative reference points for TVD video dataset for object tracking 10](#_Toc109420616)

[Table 7. Informative reference points for TVD video dataset per-sequence for object tracking 10](#_Toc109420617)

[Table 8. Informative reference points for OpenImages dataset for instance segmentation and object detection 11](#_Toc109420618)

# Introduction

In 2019 MPEG started an investigation into the area of video coding for machines. The focus of this exploration was to study the case where videos are compressed not to be looked at and evaluated by humans, but rather machine vision algorithms. These algorithms can serve different purposes such as object detection, segmentation, or tracking. As video compression standards such as HEVC or VVC are developed and optimized towards the human visual system, the existing standards may not be optimal for applications where the video is analysed by machines or where features are being compressed.

To better coordinate this study, WG 2 created a Video Coding for Machines (VCM) Ad-hoc Group (AhG) to investigate the Requirements and Use cases for Video Coding for Machines [1]. The following technologies require evidence:

* Efficient video feature compression technologies

As detailed in the use cases in [1], more video is produced than can be consumed and watched by humans. With video being by far the largest share of traffic on the internet, encoding video that is supposed to be analysed by machines in a manner that is optimized for humans seems like a wasteful idea, including using existing video compression standards to compress features for distributed network architectures. Over the past years, MPEG has studied and seen evidence that videos specifically compressed for machine vision tasks achieve a better machine vision task performance to bitrate ratio than compressing videos with the VVC Test Model (VTM) and executing the machine vision task on the decoded videos.

This call for evidence (CfE) is the start of a process that has the creation of a new international standard for Video Coding for Machines, specifically targeting feature compression, as its goal.

This call requires proponents to submit technology suitable for compressing features. **Note that it is mandatory to provide results for machine vision task of object tracking for video dataset.** Due to the scarcity of suitably licensed video datasets, an image dataset is also used in this CfE. The exact submission requirements are listed in Section 7.6.

This document contains detailed information about the setup of this call, general rules for conduct, the planned timeline for both this call and the development of a standard, detailed information on what needs to be submitted, and the next steps following the evaluation of responses to the call.

# Who may Participate?

Respondents that respond to this call may include any persons whether they are or are not accredited delegates of ISO/IEC JTC1/SC29/WG2. However, all respondents are required to attend the meetings at which their respective responses are evaluated.

A one-time invitation may be extended to respondents to participate in the evaluation process if the respondent is not an accredited delegate of ISO/IEC JTC1/SC29/WG2. Information for how to join National Body committees and to become an accredited delegate for ISO/IEC JTC1/SC29/WG2 is available at [How to Get Involved](https://www.iso.org/get-involved.html).

# Code of Conduct and Rules of Engagement

All participants shall be required to familiarize themselves with relevant [ISO Policies and Procedures](https://www.iso.org/resources.html), including in particular [ISO Code of Conduct](https://www.iso.org/publication/PUB100397.html), [ISO Declaration for Participants in ISO Activities](https://www.iso.org/declaration-for-participants-in-iso-activities.html), [ISO Privacy and Copyright](https://www.iso.org/privacy-and-copyright.html) policy, and [ISO Policy on Communication of Committee Work](https://www.iso.org/publication/PUB100382.html), and to consent to be bound by these policies.

# Source Code and IPR

There is no obligation of the proponent to release source code, provided experts are still able to conclude that technical evidence to commence standardization work exists.

Furthermore, respondents are advised that this Call is being made subject to the common patent policy of ITU-T/ITU-R/ISO/IEC (refer to [www.itu.int/ITU-T/dbase/patent/patent-policy.html](http://www.itu.int/ITU-T/dbase/patent/patent-policy.html) or Appendix I of [ISO/IEC Directives Part 1](http://isotc.iso.org/livelink/livelink?func=ll&objId=4230455&objAction=browse&sort=subtype)).

# Definitions

The definitions for terms associated with this Call for Evidence can be found in section 1 of [1].

# Documents of CfE Package

The CfE package consists of the following documents:

* N00190 Requirements and Use Cases for Video Coding for Machines

This document describes the use cases where responses can be applied and what requirements the final standard needs to fulfill.

* N00215 Call for Evidence on Video Coding for Machines (this document)

This document contains details about the submission process and the guidelines to follow. Furthermore, it describes who may participate and what the rules for participation are.

# Submission Process

## CfE Timeline Table

Each entry in Table 1 is described in the section below. WG2 is the SC 29 working group WG2 MPEG Technical Requirements. Unless stated otherwise, deadlines refer to a specific day at 23:59 UTC.

Table 1. CfE Timeline

|  |  |  |  |
| --- | --- | --- | --- |
| **Meeting** | **Date** | **Who** | **Action** |
| 8 | Jul 2022  (18-22) | WG2 | Issue Call for Evidence package |
|  | Aug 2022 05 | WG2 | Verified version of the test material is available |
|  | Aug 2022  26 | Respondent | Register |
|  | Oct 2022 14 | Respondent | Submit response as a contribution to the 9th WG 2 meeting |
| 9 | Oct 2022 (24-28) | WG2 | Discuss the Call for Evidence submissions and produce the CfE response report. |

## Register your participation

Respondents must register on or before the date shown in the CfE Timeline Table 1, an intention to participate in the CfE. Registering an intent is not binding and registered parties are not required to make a submission. However, parties that do not register will not be able to make a submission. Each organization shall only register once. Register by sending an email to the people detailed in section 8. This email should indicate:

* Company name
* Contact name and contact email address
* The number of submissions the respondent plans to submit
* The envisioned scope of the submission(s) (e.g., will it process all or only some content types, will it meet all or only some requirements). This envisioned scope is not binding and is not a restriction on submissions but is rather for planning purposes only.

## Mandatory Equipment, Software, and Data Components

The use of the following software and data components is required to regenerate the anchors used in the CfE. Where applicable, the same software and configuration shall be used for the proposed technology as for the anchors, e.g., for converting YCbCr files to PNG FFmpeg shall be used.

* FFmpeg, version 4.2.2: [https://ffmpeg.org/releases](https://ffmpeg.org/releases/)
* VTM, version 12.0: <https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM>

## Access to Test Material

Respondents can obtain access to a verified version of the test material by the date shown in the CfE Timeline Table 1 above. Although the datasets are currently publicly available, a verified collection of the data will be made available together with the anchor results. The purpose of this verified collection of datasets is to be independent of changes made to the datasets, as MPEG does not own these datasets.

An overview of the datasets to be used can be found in Appendix A in this document.

Datasets and the corresponding scripts to generate the anchor results will be included in the test material. The test material can be accessed at <ftp://mpeg.org,> and the username and password will be communicated by the VCM Co-chairs list in Chapter 8 to interested parties directly upon email request.

## Conduct Objective Evaluations

Respondents must provide the objective measurement of the following parameters. Details on how to measure the following metrics are given in Appendix B.

The following objective results are to be reported:

* Machine vision task performance (per-sequence and overall MOTA for object tracking, mAP@0.5 for instance segmentation and (optional) object detection)
* For video dataset(s): Per-sequence and average of per-sequence bitrate (kbps)
* For image dataset(s): BPP calculated from total compressed bits and total pixels of the dataset
* Runtime: Encoding, decoding and task time (part 1, part 2) (optional)

For each dataset and task, objective results are to be presented as follows:

* Tabular presentation of Bitrate/BPP/MOTA/mAP/Runtime
* BD-rate, BD-[MOTA/mAP] (mAP/MOTA vs. bitrate/BPP) gain compared to feature anchor
* RD curve showing task performance vs bitrate/BPP for the result and for corresponding feature anchor and informative anchor.

A template for the submission of this information is given in the attached excel document. The respondent shall include a summary of the objective results in the contribution document describing the proposed technology. This shall include a description of the any software libraries, hardware and software environment where the objective evaluation is performed.

The performance for the machine vision tasks shall be evaluated against the anchors produced using the following neural networks:

* TVD (object tracking - JDE -1088x608 [3] DarkNet-53 split point) (mandatory)
* OpenImages V6 (instance segmentation - Mask R-CNN X101-FPN [2] P-layer split point) (mandatory)
* OpenImages V6 (object detection - Faster R-CNN X101-FPN [2] P-layer split point) (optional)

## Details of the Submission

Respondents must submit their response as an input document to the 9th WG2 meeting by the date shown in the CfE Timeline Table 1.

Respondents shall not manually select nor hardcode the encoder and decoder algorithm or parts of the encoder and/or decoder algorithm according to the testing and validation dataset, test data, or machine tasks. The proposed encoding and decoding methods shall support different network models, e.g., Mask R-CNN and JDE -1088x608, and various machine tasks, e.g., instance segmentation and object tracking.

Respondents shall submit results for each bitrate point for each of the datasets: TVD (object tracking) and OpenImages V6 (instance segmentation).

Respondents are encouraged to use publicly available datasets with permissible license terms to be used for standardization activities for training – respondents need to specify the training datasets they used.

A diagram describing the pipeline of responses can be found in Figure 1.

VCM encoder

VCM decoder

Neural Network Task (part 2)

Video

Bit stream

Reconstructed Feature

Feature Encoding

Neural Network Task (part 1)

Feature Decoding

Figure 1 Pipeline of response

### Target rate points

For each dataset and task, respondents are to provide results targeting six rate points. The bitrate or BPP shall not exceed the values defined in Table 2, Table 3, Table 4, and Table 5 for the indicated datasets and tasks. In order to produce a reliable BD-rate measurement, the MOTA/mAP should cover a similar range to that of the results in the anchors.

The video sequences shall be coded in a random-access configuration, meaning no more than 32 frames of structural delay, e.g., 32 pictures “group of pictures (GOP)”, and random-access intervals of 64 pictures or less for a video sequence with a frame rate of 50 frames per second.

Table 2. Rate points for TVD videos for object tracking (overall)

|  |  |
| --- | --- |
| **Rate point** | **Bitrate(kbps)** |
| Rate point 1 | 8254.629 |
| Rate point 2 | 6075.859 |
| Rate point 3 | 4393.071 |
| Rate point 4 | 2520.689 |
| Rate point 5 | 1664.718 |
| Rate point 6 | 1106.308 |

Table 3. Rate points for TVD videos for object tracking (per-sequence)

|  |  |  |
| --- | --- | --- |
| **Sequence** | **Rate point** | **bitrate (kbps)** |
| TVD-01 | Rate point 1 | 10039.353 |
| Rate point 2 | 7133.127 |
| Rate point 3 | 4870.162 |
| Rate point 4 | 2466.427 |
| Rate point 5 | 1428.580 |
| Rate point 6 | 755.897 |
| TVD-02 | Rate point 1 | 7403.181 |
| Rate point 2 | 5018.684 |
| Rate point 3 | 3309.629 |
| Rate point 4 | 2669.620 |
| Rate point 5 | 1654.542 |
| Rate point 6 | 1287.730 |
| TVD-03 | Rate point 1 | 6192.655 |
| Rate point 2 | 5004.977 |
| Rate point 3 | 4075.076 |
| Rate point 4 | 2549.853 |
| Rate point 5 | 1971.010 |
| Rate point 6 | 1507.273 |

Table 4. Rate points of instance segmentation on OpenImages dataset

|  |  |
| --- | --- |
| **Rate point** | **BPP** |
| Rate point 1 | 1.377 |
| Rate point 2 | 0.946 |
| Rate point 3 | 0.650 |
| Rate point 4 | 0.443 |
| Rate point 5 | 0.295 |
| Rate point 6 | 0.195 |

Table 5. Rate points of (optional) object detection on OpenImages dataset

|  |  |
| --- | --- |
| **Rate point** | **BPP** |
| Rate point 1 | 1.355 |
| Rate point 2 | 0.931 |
| Rate point 3 | 0.641 |
| Rate point 4 | 0.437 |
| Rate point 5 | 0.292 |
| Rate point 6 | 0.194 |

### Informative reference point

To facilitate comparison of the performance of feature compression technology against state-of-the-art technology, an informative reference point is provided.

The anchor results in Table 6 and Table 7 for video compression provide an informative reference point against the case of not performing feature compression for the video dataset. The results in Table 6 are the frame-averaged bitrates and MOTA scores. The results in Table 7 are the pre-sequence bitrates and MOTA scores. The anchor results in Table 8 provide an informative reference point against the case of not performing feature compression for instance segmentation and object detection.

Table 6. Informative reference points for TVD video dataset for object tracking

|  |  |  |
| --- | --- | --- |
| **Reference point** | **bitrate (kbps)** | **MOTA (object tracking)** |
| Reference point 1 | 4756.92 | 51.10% |
| Reference point 2 | 2148.92 | 49.90% |
| Reference point 3 | 1002.07 | 46.80% |
| Reference point 4 | 472.07 | 40.00% |
| Reference point 5 | 207.59 | 32.70% |
| Reference point 6 | 87.30 | 26.10% |

Table 7. Informative reference points for TVD video dataset per-sequence for object tracking

|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence** | **Reference point** | **bitrate (kbps)** | **MOTA(object tracking)** |
| TVD-01 | Reference point 1 | 3544.371199 | 37.3% |
| Reference point 2 | 1584.968259 | 35.7% |
| Reference point 3 | 731.1409347 | 30.7% |
| Reference point 4 | 329.2842596 | 21.9% |
| Reference point 5 | 139.2548007 | 12.8% |
| Reference point 6 | 58.17240493 | 3.3% |
| TVD-02 | Reference point 1 | 693.5037862 | 55% |
| Reference point 2 | 358.2855385 | 52.2% |
| Reference point 3 | 190.4754646 | 51.1% |
| Reference point 4 | 102.5553176 | 43.5% |
| Reference point 5 | 40.32515881 | 34.9% |
| Reference point 6 | 18.69057154 | 31.4% |
| TVD-03 | Reference point 1 | 10032.89425 | 68.5% |
| Reference point 2 | 4503.496998 | 67.9% |
| Reference point 3 | 2084.580128 | 67.1% |
| Reference point 4 | 984.3581748 | 63% |
| Reference point 5 | 443.2046287 | 58.1% |
| Reference point 6 | 185.0438772 | 54.9% |

Table 8. Informative reference points for OpenImages dataset for instance segmentation and object detection

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Reference point** | **BPP (instance segmentation)** | **mAP [%] (instance segmentation)** | **BPP (object detection)** | **mAP [%] (Object Detection)** |
| Reference point 1 | 0.841 | 80.536 | 0.863 | 78.929 |
| Reference point 2 | 0.493 | 80.197 | 0.509 | 77.989 |
| Reference point 3 | 0.277 | 78.775 | 0.287 | 77.263 |
| Reference point 4 | 0.147 | 75.653 | 0.153 | 73.963 |
| Reference point 5 | 0.074 | 69.917 | 0.078 | 68.842 |
| Reference point 6 | 0.036 | 57.773 | 0.037 | 58.021 |

## Submission of response documentation

Respondents shall submit the following information as a contribution to the MPEG meeting indicated in the CfE Timeline Table 1:

* A written description of the technology having sufficient detail to permit technical discussions
* Objective test results, including machine vision task performance, bitrate/BPP, and (optional) runtime measurement as indicated in Appendix B
* Description of how the requirements in [1] are met
* Description of the training details if the response contains components that are learned
* Description of the software libraries, hardware (CPU and GPU types) and software environment where the objective evaluation is performed.
* Reporting template attached to this document with all mandatory information filled out accordingly

Upon entering the data in the attached reporting template, several different performance measurements will be automatically calculated. This includes BD-rate and BD-mAP/MOTA. The results will be summarized in the “Summary” sheet of the template. The summarized results shall be reported in the contribution.

A template for reporting the training details can be found in Appendix C.

Respondents that are WG2 members shall register and upload an input contribution to the WG2 meeting as indicated in section 7 prior to the contribution upload deadline. The contribution shall contain all documentation mentioned above.

Respondents that are not WG2 members shall email the documents to the convenor of WG2 two weeks before the 9th WG2 meeting, so that the documents can be registered and uploaded by the convenor. The documents should be written in Microsoft Word. A template for input contributions can be found at the FTP site where the test data is available (see section 7.4). The Convenor of WG2 will extend a one-time-only invitation to the WG2 meeting so that a non-member respondent can present their contributions and participate in the selection process.

All respondents are urged to become WG2 members (see Section 2).

## Evaluate CfE Submissions

At the WG2 meeting indicated in the CfE Timeline Table 1, submissions will be evaluated by the WG2 experts. It is strongly urged that respondents have experts familiar with the submitted technology attend to allow discussions on details of the submissions.

Submissions shall be evaluated considering all submitted information including objective results.

The metrics for evaluating the submissions can be found in Section 7.5.

# Call Administrator

This Call for Evidence is issued by WG2. The WG2 convenor serves as the administrator:

Igor Curcio

Convenor, WG2 MPEG Technical Requirements

[igor.curcio@nokia.com](mailto:igor.curcio@nokia.com)

For any questions related to this Call for Evidence or associated evaluation procedures please contact the co-chairs of the VCM AhG:

|  |  |
| --- | --- |
| Chris Rosewarne  Canon  [chris.rosewarne@canon.com.au](mailto:chris.rosewarne@canon.com.au) | Yuan Zhang  China Telecom  [zhangy666@chinatelecom.cn](mailto:zhangy666@chinatelecom.cn) |
| Shan Liu  Tencent  [shanl@tencent.com](mailto:shanl@tencent.com) | Christopher Hollmann  Ericsson  [christopher.hollmann@ericsson.com](mailto:christopher.hollmann@ericsson.com) |

# Email reflector

For communication, usage of the reflector is encouraged: [mpeg-vcm@lists.aau.at](mailto:mpeg-vcm@lists.aau.at)

You can subscribe to the VCM reflector on the following webpage: <https://lists.aau.at/mailman/listinfo/mpeg-vcm>

Important information such as scheduling for AhG, BoG, or WG2 meetings may be shared via this reflector.

WG2 is using the following reflector: [mpeg-req@lists.aau.at](mailto:mpeg-req@lists.aau.at)

You can subscribe to the WG2 reflector on the following webpage: <https://lists.aau.at/mailman/listinfo/mpeg-req>

# References

|  |  |
| --- | --- |
| [1] | "N00190 Use cases and draft requirements for Video Coding for Machines," Online. |
| [2] | "Detectron2, https://github.com/facebookresearch/detectron2," Facebook AI Research. |
| [3] | "Towards-Real-Time-MOT, https://github.com/Zhongdao/Towards-Realtime-MOT," Online. |
| [4] | M. Everingham, L. Van Gool, C. K. Williams, J. Winn and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *International Journal of Computer Vision,* vol. 88, no. 2, pp. :303-338., 2010. |
| [5] | M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn and A. Zisserman, "The Pascal Visual Object Classes Challenge: A Retrospective," *International Journal of Computer Vision,* vol. 111, no. 1, pp. 98-136., 2015. |
| [6] | K. Bernardin, A. Elbs and A. Stiefelhagen, "Multiple Object Tracking Performance Metrics and Evaluation in a Smart Room Environment," Sixth IEEE International Workshop on Visual Surveillance in Conjunction with ECCV, May 2008. |

**Appendix A: Detailed description of test datasets**

**A.1 Dataset 1: Tencent Video Dataset (TVD)**

The Tencent Video Dataset (TVD) consists of 166 images for object detection and segmentation as well as 3 video sequences for object tracking. The three video sequences, TVD-01, TVD-02 and TVD-03 are used for the CfE for video coding for machines. TVD-01, TVD-02 and TVD-03 have 3000, 636 and 2334 frames respectively, and the resolution of the videos is 1920x1080. The files are in MP4 format. The dataset is provided and labeled by Tencent.

Detailed information can be found on <https://multimedia.tencent.com/resources/tvd>.

**A.2 Dataset 2: OpenImages v6**

OpenImages V6 is a large-scale dataset, consists of 9 million training images, 41,620 validation samples, and 125,456 test samples. Note that all images are already compressed.

In this Call for Evidence, a subset of the OpenImages dataset is used. A total of 5000 images were selected for object detection and object segmentation. While there is an overlap between the two subsets, these are not identical.

More information on the dataset can be found on <https://storage.googleapis.com/openimages/web/index.html>.

For the machine vision task performance, mAP@0.5 shall be used.

The dataset is available using the following license text:

*The annotations are licensed by Google LLC under the* [*CC BY 4.0*](https://creativecommons.org/licenses/by/4.0/) *license. The images are listed as having a* [*CC BY 2.0*](https://creativecommons.org/licenses/by/2.0/) *license.* ***Note:*** *while we tried to identify images that are licensed under a Creative Commons Attribution license, we make no representations or warranties regarding the license status of each image and you should verify the license for each image yourself.*

**Appendix B: Anchor metrics**

## B.1 Bitrate measurement

For image datasets, bits per pixel (BPP) shall be used. BPP is the number of bits occupied by each pixel, which is defined by:

“Total pixels” refers to the total number of pixels overall images at their original resolution.

For video sequences, the bitrate shall be measured in kilobits per second (kbps). This is defined as:

Here *fps* denotes the number of frames per second and *frames* denote the number of encoded frames.

## B.2 Task: Object Tracking

For the object tracking task, Multiple Object Tracking Accuracy (MOTA) [6] shall be used to measure performance.

The MOTA accounts for all object configuration errors made by the tracker, false positives, misses (true negative), mismatches, and overall frames.

where , , and are the number of false negatives, the number of false positives, the number of mismatch error (ID Switching between 2 successive frames), and the number of objects in the ground truth respectively at time .

For object detection and segmentation performance, mAP@0.5 shall be used. The performance for object tracking shall be measured in MOTA.

## B.3 Task: Object segmentation / Object detection

For both object detection and object segmentation, mean Average Precision (mAP) [4] [5] shall be used to measure the performance of the network.

For a given category of object, true positive , false positive , false negative , and true negative are defined with an Intersection over Union (IoU) threshold for that category, where true/false represents the output of the neural network, positive/negative represents the label in the ground truth.

Then, recall of the given IoU threshold is defined as the proportion of all true positive examples in all true positive and false negative examples corresponding to that IoU threshold:

The precision of the given IoU threshold is the proportion of all true positive examples which are from all positive examples:

A neural network of segmentation may achieve several pairs of recall and precision values corresponding to a certain IoU threshold and different confidence levels. For each recall value in the pairs, let takes the maximum precision value in all precision values for which the corresponding recall values are above the given recall value :

Average Precision (AP) of a given category of object is defined as the average value of for all recall values provided by the neural network, which can characterize the area of the entire precision-recall curve.

Mean Average Precision (mAP) is an averaged AP overall categories of objects and in a range of IoU thresholds. As an example, in MS COCO 2017 dataset, 10 IoU thresholds are taken at equal intervals from 0.50 to 0.95. In particular, AP50 and AP75 generally present the mAP when the IoU threshold is 0.50 and 0.75 respectively.

The following variants of mAP are used:

* [mAP@0.5](mailto:mAP@0.5): the mAP when the IoU threshold is 0.5
* [mAP@[0.5:0.05:0.95](mailto:mAP@[0.5:0.05:0.95)]: the average of 10 IoU thresholds at equal intervals from 0.50 to 0.95.

A different neural network may provide object detection or segmentation results with a different number of confidence level. This may affect the quality evaluation in VCM. So we recommend specifying the number of confidence levels provided by the neural network for each dataset/task. For example, the number of confidence levels provided by a new neural network should be the same as anchors. This will help to align potential responses to the CfE and easy comparison.

## B.4 Runtime Measurement

Runtime includes Encoding time (EncT), Decoding time (DecT) and Task time for part 1 and part 2 of the network (TaskT1, TaskT2) for complexity measurement. The proposed runtime measurements for a VCM solution are:

* **TaskT1**: Time needed to perform part 1 of the network (e.g., the backbone) to produce features.
* **EncT:** Time needed to convert feature input to bitstream.
* **DecT:** Time needed to convert bitstream to decoded features.
* **TaskT2**: Time needed to perform part 2 of the network (e.g., the head) to complete the task based on the decoded features.

For the purpose of reporting encoding and decoding running times, the anchor and proposal should be simulated on the same platform, e.g., similar CPU and GPU configuration, to have reliable time comparison.

**Appendix C: Inference and Training information**

## C.1 Inference information

The information described below is required to be provided for the inference process for both encoding and decoding processes.

* **Network Visualization:** Graphical representation of the neural network
* **Param. Number**: Total numbers of parameters in the neural network.
* **Param. Precision**: Bits for storing one parameter. Additionally, use “I” for indicating an integer parameter and use “F” to indicate a floating-point number. For example, if the proposed method uses 16-bit integer to represent a parameter, you can report this information as “16 (I)”.
* **MAC (Giga):** Number of multiply–accumulate (MAC) operations per pixel in the worst case for the inference stage, where the multiply–accumulate operation is a common step that computes the product of two numbers and adds that product to an accumulator. Since different size of input may influence the value, it is suggested to use 3840x2160 as the input frame size for unification.
* **Mem.T (MB):** Temporary memory. It denotes the memory used to store the output feature map for all intermediate layers (forward pass). Since different size of input may influence the value, it is suggested to use 3840x2160 as the input frame size for unification. For reporting Mem.T (MB) the calculation process is also suggested to be provided for crosschecking.
* **Patch Size**: The size of input to the neural networks during inference (patchW×patchH×patchT, e.g., 64x64x3) where applicable (e.g., when patch-wise processing of features is performed).

## C.2 Training information

When applicable, it is required to report and discuss the following information for the training process.

* **Epoch**: The number of complete passes through the training data (e.g., 100)
* **Batch Size**: The number of samples processed before the model is updated. (e.g., 4Kx16frames)
* **Training Time**: CPU and/or GPU (e.g., 48h) and hardware such as CPU/GPU model and count (if different to that used for inferencing).
* **Learning Curve:** Plot of the training loss and validation loss (or similar) versus the number of epochs
* **Training Sets**: Training sets used. If a pre-trained model is used, the source of the pre-trained model and its training sets should be reported in detail.
* **Training Configuration per Rate-Distortion Point**: Any changes in the requested information used to generate different rate-distortion points

The following additional training information could help to better understand proposed neural network-based methods:

* **Number of Iterations:** number of gradient updates within an epoch
* **Patch Size**: size of input to the neural networks (patchW×patchH×patchT, e.g., 64x64x3) where applicable (e.g., when patch-wise training using features is performed).
* **Learning Rate**: The amount that the weights are updated during training (e.g., 5e-4)
* **Optimizer**: The algorithm used to change the attributes of proposed neural networks (e.g., ADAM)
* **Loss Function**: The function to calculate the model error during training and optimization (e.g., L1, L2, etc.)
* **Preprocessing**: (e.g., preprocessing procedure, normalization, cropping method, rotation, zoom etc.)

**Appendix D** **Anchor generation environment**

## D.1 Object Tracking on TVD (feature anchor)

Software package versions used to generate the object tracking feature anchor are as follows:

* CentOS 7.9.2009
* CUDA 11.3
* Nvidia driver 465.19.01
* Python 3.6.8
* Numpy 1.19.5
* PyTorch 1.7.1
* Pandas 1.1.5
* PIL 8.2.0
* GNU 'parallel' utility (for parallel VTM, inferencing execution)
* ffmpeg 4.2.2
* VTM-12.0 software (extracted and compiled under the `build` directory as 'VTM-12.0' subdirectory)

## D.2 Instance Segmentation on OpenImages (feature anchor)

Software package versions used to generate the instance segmentation feature anchor are as follows:

* Ubuntu 18.04.4 LTS
* Nvidia Driver version: 460.91.03
* CUDA: 11.2
* Python: 3.6.9
* Torch: 1.8.0
* Tensorflow: 2.6.0
* Detectron2: 0.4
* VTM: 12.0
* pandas: 1.1.5
* opencv-python: 4.5.1.48

Detectron2 and Tensorflow Object Detection API

* Install detectron2 following instructions from the webpage <https://github.com/facebookresearch/detectron2/releases/tag/v0.4>
* Install Tensorflow Object Detection API by following the webpage: <https://tensorflow-object-detection-api-tutorial.readthedocs.io/en/latest/install.html#tensorflow-object-detection-api-installation>

## D.3 Object Detection on OpenImages (feature anchor)

Software package versions used to generate the object detection feature anchor are as follows:

* Ubuntu 20.04.1 LTS
* Python 3.8.11
* Torch 1.9.0
* Detectron2 0.5
* Object-detection 0.1
* Pandas 1.3.3
* Numpy 1.21.2
* Opencv-python 4.5.3.56
* Pillow 8.3.1
* ffmpeg 4.4
* VTM 12.0

## D.4 Object Tracking on TVD (informative video anchor)

Software package versions used to generate the object tracking video anchor are as follows:

* Ubuntu 18.04.3 LTS
* Nvidia Driver version: 450.80.02
* CUDA: 11.0
* Python: 3.8.8
* Torch: 1.8.1
* Detectron2: 0.4
* VTM 12.0i
* FFMPEG: 4.2.2

## D.5 Instance Segmentation / object detection on OpenImages (informative anchor)

Software package versions used to generate the instance segmentation and object detection anchor are as follows:

* Ubuntu 18.04.4 LTS
* Nvidia Driver version: 440.100 or above
* CUDA: 10.2
* Python: 3.7.4
* Torch: 1.6.0
* Tensorflow: 2.7.0
* Detectron2: 0.2.1
* VTM: 12.0
* pandas: 1.2.4
* opencv-python: 4.5.2.52

Detectron2 and Tensorflow Object Detection API

* Install detectron2 0.2.1 following instructions from the webpage: <https://github.com/facebookresearch/detectron2/releases/tag/v0.2.1>
* Install Tensorflow Object Detection API by following the webpage: <https://tensorflow-object-detection-api-tutorial.readthedocs.io/en/latest/install.html#tensorflow-object-detection-api-installation>