

**ISO/IEC JTC 1/SC 29/WG 11**

**Coding of moving pictures and audio**

**Convenorship: UNI (Italy)**

**ISO/IEC JTC 1/SC 29/WG 11 w19080**

**Document type: Approved WG 11 document**

**Title: Evaluation Framework for Video Coding for Machines**

**Status: Approved**

**Date of document: 2020-01-17**

**Source: Requirements**

**Expected action:**

**No. of pages: 5**

**Email of convenor: leonardo@chiariglione.org**

**Committee URL: mpeg.chiariglione.org**

**INTERNATIONAL ORGANIZATION FOR STANDARDIZATION**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

**ISO/IEC JTC1/SC29/WG11**

**CODING OF MOVING PICTURES AND AUDIO**

**ISO/IEC JTC1/SC29/WG11 MPEG2018/w19080**

**January 2020, Brussels, Belgium**

**Title Evaluation Framework for Video Coding for Machines**

**Source MPEG Requirements**

**Status:**

# Introduction

The MPEG activity on Video Coding for Machines (VCM) aims to standardize a bitstream format generated by compressing both a video stream and previously extracted features. The bitstream should enable multiple machine vision tasks. VCM shall be able to

* Efficiently compress the bitstream; the size of the compressed features shall be less than the encoded video stream using state-of-the-art video compression technologies like HEVC.
* Use the bitstream to support single or multiple tasks. Features should be general enough to be usable for different scenarios, for example object detection and segmentation.
* Support varying performance for multiple tasks as measured by the appropriate metrics. This performance level may depend on the application.
* Allow the reconstruction of the compressed bitstream for human consumption. This can be achieved with an additional bitstream.

MPEG VCM has identified a set of relevant use cases and related requirements [1], focusing on the machine-to-machine communication in intelligent transportation and the hybrid machine and human consumption for surveillance and smart city use cases. This document contains information on how to provide evidence for these use cases. It contains details about

* Datasets: which datasets should be used for which sub-tasks, where these datasets can be obtained, how the datasets are split into training and validation data
* Metrics: which metric shall be used for which sub-tasks, how these metrics are calculated, what to compare performance results against

# Test Conditions

There are two separate tests that can be performed either in parallel or in a single framework. Framework refers to the used datasets and software packages. Regarding the general feature extractor, proponents are asked to test one or more key tasks for a specific use case with the same feature extraction backbone and compare the performance results to current benchmarks. Retraining the shared backbone is permitted using joint training or other approaches in the case of two or more key tasks. Modifications and training of the task-specific networks are allowed. In some cases, the encoder may know the task-specific neural networks at the decoder side.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Metrics | Datasets | Benchmarks | Training/Testing |
| Object Detection | [mAP](https://mc.ai/the-confusing-metrics-of-ap-and-map-for-object-detection/) | COCO [compressed] | <http://cocodataset.org/#detection-leaderboard> | For COCO, use 2014 Val set for evaluation and 2014 Train in the case of retraining. |
| [CityScapes](https://www.cityscapes-dataset.com/) [uncompressed] | <https://www.cityscapes-dataset.com/benchmarks/> | For CityScapes, use defined train and validation sets |
| [ImageNet](http://www.image-net.org/) [compressed] | <https://kobiso.github.io/Computer-Vision-Leaderboard/imagenet.html> | For Imagenet, use the training and validation data as published from ILSVRC 2014. |
| [Visdrone](http://www.aiskyeye.com/) | <http://www.aiskyeye.com/views/getInfo?loc=14> |  |
| Object Segmentation | [mAP](https://mc.ai/the-confusing-metrics-of-ap-and-map-for-object-detection/) | COCO |  | (see above) |
| [CityScapes](https://www.cityscapes-dataset.com/) [uncompressed] |  | (see above) |
| [BD100K](https://bdd-data.berkeley.edu/) | <https://arxiv.org/abs/1805.04687> | We recommend using the predefined splits. |
| [KITTI](http://www.cvlibs.net/datasets/kitti/index.php) | <http://www.cvlibs.net/datasets/kitti/eval_object.php> | We recommend using the predefined splits. |
| Image Enhancement | PSNR | REDS |  | Train and Validation subsets are publically available. |
| Object Tracking | MOTA | MOT19 | <https://arxiv.org/pdf/1906.04567.pdf> | Dataset split is available from the Tracking Challenge, available on their website. |
| Event Recognition | Accuracy, [mAP](https://mc.ai/the-confusing-metrics-of-ap-and-map-for-object-detection/) | [UCF101](https://www.crcv.ucf.edu/data/UCF101.php) |  | The datasets, actions, and splits are available from their website for UCF101 |
| [Kinetics600](https://deepmind.com/research/open-source/kinetics) |  | There is a standard test set on Kinetics-600 |
| Event Prediction | Accuracy | [UCF101](https://www.crcv.ucf.edu/data/UCF101.php) |  | (see above) |
| [Kinetics600](https://deepmind.com/research/open-source/kinetics) |  | (see above) |
| Anomaly Detection | Accuracy, [mAP](https://mc.ai/the-confusing-metrics-of-ap-and-map-for-object-detection/) | [UCF Real World](https://www.crcv.ucf.edu/projects/real-world/), Street Scene |  | There is a dataset split into Anomaly Test and Anomaly Train for both anomaly detection and event detection. |
| Event Search | Accuracy | [Oxford 5k](https://www.robots.ox.ac.uk/~vgg/data/oxbuildings/) |  | The images are available from their website. |

The majority of these datasets have publicly defined training and validation sets. In the case this is not available, we will release a training and testing split for comparison. This list is not exhaustive, and proponents are free to use their own datasets for each of the key tasks.

The input images and labels for training and testing are directly taken from the dataset for specific use cases as listed above. This leads into a general feature extractor such as a convolutional neural network, which converts the images or video into a stream of unprocessed or processed video. The resulting features are then fed into different machines, whose results are calculated with respect to the appropriate metric. Proponents are asked to report this result along with the current state of the art on the chosen group of tasks, which will be released by MPEG-VCM. Comparison will be made regarding the performance across the different tasks in the group measured by the relevant metric.

Regarding the compression of processed or unprocessed video, proponents are asked to test the compression ratio on the processed or unprocessed video. This compression ratio should be given as comparison to the released compression ratio of HEVC on the unprocessed video. For human consumption use cases, proponents shall report BD-rate.

MPEG-VCM will release a stream of feature maps taken from video for several use cases based on current state of the art computer vision, along with the performance measured across different datasets. MPEG-VCM will also release the compression ratio and performance graphed for HEVC with a common profile. Proponents are asked to test their compression coding on this stream, and shall report the bitrate of the compressed stream to the provided stream and report this ratio. Proponents are also asked to report the performance on the chosen database in the case of lossy coding.

For comparison, proponents are asked to report their performance on key tasks for a subset of the bits-per-pixel (bpp) [0.1, 0.2, 0.5, 1.0].

# Evaluation Methods and Procedures

The evaluation procedure and metrics are described in section 4 above. The metrics consist of two parts, one relating to feature extraction and one relating to compression of processed or unprocessed video:

* Use case specific perforrmance metrics, with the key tasks and metrics as defined above. Proponents shall perform the evalution themselves, with the experiment conditions described in [1].
* Compression efficiency, runtime complexity and memory consumption of compression/decompression (measurement is independent of the use case). Proponents shall perform the evaluation themselves based upon a provided unprocessed or processed video. In the case of processed video, the output may come from common neural network or general feature extraction methods regarding the specific key tasks. As an example, these common neural networks backbones may be VGG, ResNet, Inception and the specific frameworks depend on the key tasks. For detection and segmentation, an example may be Mask RCNN or YOLO.

# References

1. N18768, Use cases and requirements for Video Coding for Machines, Geneva, SW, October 2019.