**** **ISO/IEC JTC 1/SC 29/WG 2 N19**

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

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

**Document type:** Output Document

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

**Status:** Approved

**Date of document:** 2020-10-16

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

**Expected action:** none

**Action due date:** none

**No. of pages:** 50 (Including the cover page)

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

**Committee URL:** <https://isotc.iso.org/livelink/livelink/open/jtc1sc29wg2>

**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 N19**

**Online – October 2020**

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

**Source: WG 2 MPEG Technical requirements**

**Status: Approved**

**Serial number: 19843**

**Editor(s): M. Rafie (GTI), Yuan Zhang (China Telecom),**

**Shan Liu (Tencent)**

**Table of Contents**

[1 Introduction 4](#_Toc53661938)

[2 Test Conditions 4](#_Toc53661939)

[3 Anchor Definition and Requirements 4](#_Toc53661940)

[3.1 NN Tasks …………………. 6](#_Toc53661941)

[3.2 Network architectures 6](#_Toc53661942)

[3.3 Datasets …………………. 6](#_Toc53661943)

[4 Evaluation Methods and Procedures 9](#_Toc53661944)

[4.1 Proposed Processing Pipelines 10](#_Toc53661945)

[4.2 Pre/Post-Processing Data (image) Conversion 10](#_Toc53661946)

[4.2.1 Processing Pipeline for Downscaled Resolution 11](#_Toc53661947)

[4.3 Anchor Reference Compression / Decompression 11](#_Toc53661948)

[4.4 Anchor Performance Curve Generation 12](#_Toc53661949)

[5 Evaluation Approach for Machine and Human Consumption 12](#_Toc53661950)

[5.1 Complexity Measurement 14](#_Toc53661951)

[6 References 15](#_Toc53661952)

[7 Appendix A: Anchor Generation Results 17](#_Toc53661953)

[7.1 Object Detection 17](#_Toc53661954)

[7.1.1 Sharp vs. Tencent (Open Images) 17](#_Toc53661955)

[7.1.2 ITRI vs. Foxconn (COCO) 19](#_Toc53661956)

[7.1.3 Nokia vs. (ETRI) (CityPersons) 23](#_Toc53661957)

[7.1.4 Konkuk Univ vs. ETRI (FLIR) 25](#_Toc53661958)

[7.2 Object Segmentation 35](#_Toc53661959)

[7.2.1 Foxconn vs. ITRI (COCO) 35](#_Toc53661960)

[7.2.2 Ericsson vs. Tencent (KITTI) 38](#_Toc53661961)

[7.2.3 Nokia vs. ETRI (CityScape) 42](#_Toc53661962)

[7.3 Object Tracking (Video) 45](#_Toc53661963)

[7.3.1 Shanghai Jiao Tong Univ (MOT20) 45](#_Toc53661964)

[8 Appendix B: Anchor Runtime Measurement 50](#_Toc53661965)

# Introduction

The MPEG activity on Video Coding for Machines (VCM) aims to standardize a bitstream format generated by compressing either a video stream or 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 such as VVC.
* Use a common bitstream to support single or multiple tasks. The decompressed bitstream should be general enough to be used 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.
* For hybrid machine/human vision use cases, a common bitstream should be used for machine and human consumption; additional bitstream(s) is optional for the reconstruction of the compressed bitstream for human consumption
* The bitrate of the additional compressed bitstream shall be less than the bitrate of the bitstream at similar quality as measured by PSNR, which is the output of the VVC encoding of the unprocessed video.

MPEG VCM has identified a set of relevant use cases and related requirements [1], focusing on the machine-vision; and the hybrid machine/human-vision use cases. This document contains information on how to provide evidence for these use cases. [2]

* 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

Decoded video/feature shall be tested for one or more key tasks for a specific use case and compare the performance results to current anchors. 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 but need to be reported in detail. In some cases, the encoder may know the task-specific neural networks at the decoder side. In this document, framework refers to the used datasets and software packages.

# Anchor Definition and Requirements

VVC/H.266 codec with software version VTM-8.2 is used as the reference for the performance evaluation of MPEG-VCM encoder. Table 1 shows the tasks along with their metrics, datasets, benchmarks, and training/testing description.

Table *Training and test conditions, key metrics, datasets, benchmarks for various tasks*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task** | **Metrics** | **Datasets** | **Benchmarks** | **Training/Testing** |
| Object Detection | [mAP](https://mc.ai/the-confusing-metrics-of-ap-and-map-for-object-detection/)  vs  BPP/Rate | COCO [compressed]  (image) | <http://cocodataset.org/#detection-leaderboard> | For COCO, use 2017 Val set for evaluation and 2017 Train in the case of retraining. |
| CityPersons  [uncompressed]  (image) | <https://www.cityscapes-dataset.com/benchmarks/> | For CityScapes, use defined train and validation sets |
| Open Images  Compressed  (Image) | <https://storage.googleapis.com/openimages/web/index.html> | Version 6. |
| FLIR Thermal Dataset  (image) | <https://www.flir.com/oem/adas/adas-dataset-form/> | Ability to sense thermal infrared radiation or heat |
| Object Segmentation | [mAP](https://mc.ai/the-confusing-metrics-of-ap-and-map-for-object-detection/)  vs  BPP/Rate | COCO  [compressed]  (image) | <http://cocodataset.org/#detection-leaderboard> | (see above) |
| [CityScapes](https://www.cityscapes-dataset.com/) [uncompressed]  (image) | <https://www.cityscapes-dataset.com/benchmarks/> | (see above) |
| [KITTI](http://www.cvlibs.net/datasets/kitti/index.php)  (image) | <http://www.cvlibs.net/datasets/kitti/eval_object.php> | We recommend using the predefined splits. |
| Open Images  Compressed  (Image) | <https://storage.googleapis.com/openimages/web/index.html> | Version 6. |
| Object Tracking | MOTA  vs  Rate | MOT20  [compressed]  (video) | <https://arxiv.org/pdf/1906.04567.pdf> | Dataset split is available from the Tracking Challenge, available on their website. |
| Hybrid Machine / Human Vision | BD-rate  (mAP/MOTA, SSIM, PSNR vs. BPP/bitrate) | Any combination of listed Datasets  (video) | See above | See above |

## NN Tasks ………………….

For the MPEG-VCM performance evaluation, the anchors are generated for the following key NN tasks (see Appendix A):

* Object Detection (still image)
  + Open Images
    - Sharp vs. Tencent (crosschecker)
  + COCO
    - ITRI vs. Foxconn (crosschecker)
  + FLIR
    - Konkuk Univ vs. ETRI (crosschecker)
  + CityPersons
    - Nokia vs. (ETRI) (crosschecker)
* Object Segmentation (still image/video)
  + COCO
    - Foxconn vs. ITRI (crosschecker)
  + CityScapes
    - Nokia vs. ETRI (crosschecker)
  + KITTI
    - Ericsson vs. Tencent (crosschecker)
* Object Tracking (video)
  + MOT20
    - Shanghai Jiao Tong Uni (no crosschecker)

## Network architectures

* Object Detection: Faster R-CNN X101-FPN (part of Facebook AI Research’s Detectron)

* Object Segmentation: R50-FPN Cityscapes (part of Facebook AI Research’s Detectron), PreMOVOS

* Object Tracking: JDE-1088x608
* Optional requirement
  + As an additional performance data, other network architectures are allowed to be used per tasks. However, it is up to the proponents to provide comparable performance data with the anchors specified in this document

## Datasets ………………….

The datasets to be used for anchor generation evaluation framework and VCM standardization process shall be of high quality, available and downloadable, sufficiently modern, have permissive licenses, and support adequate pre-trained models.

* Datasets must be of high quality such that a VVC encoder can create several coded versions of the sequence with noticeable degradation in quality. It is recommended to use uncompressed datasets where possible.
* In some datasets, most images are compressed with quality factors around 96 and 90, while some images were even compressed with quality factor around 80. It is recommended to build up a subset of the dataset by removing the lower quality images.
* Datasets shall be capable of generating anchors per requirements
* Datasets must be available and downloadable
* Datasets shall be sufficiently modern
* Datasets that support pre-trained models

Proponents are invited to look into the datasets and raise a flag in case of encountering issues such as copyrights, etc. (Note: It is intended to have these datasets to be downloadable from the MPEG site). Table 2 shows the recommended datasets for VCM anchor generation.

**(Note 1:** Dataset licensing issue and how it relates to MPEG-VCM to be clarified.)

Table *. The following datasets are recommended to be used for anchor generation*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Dataset** | **Image /**  **Video** | **Compressed /**  **Uncompressed** | **Benchmarks** | **Comments** |
| COCO | Image | Compressed | <http://cocodataset.org/#detection-leaderboard> | For COCO, use 2017 Val set for evaluation and 2017 Train in the case of retraining  80 object classes  “The annotations in this dataset along with this website belong to the COCO Consortium and are licensed under a [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/legalcode).” (see Note 1.)  “The COCO Consortium does not own the copyright of the images. The use of the images must abide by the [Flickr Terms of Use](https://www.flickr.com/creativecommons/). The users of the images accept full responsibility for the use of the dataset, including but not limited to the use of any copies of copyrighted images that they may create from the dataset.” |
| CityScapes | Image | Uncompressed | <https://www.cityscapes-dataset.com/benchmarks/> | For Cityscapes, use defined train and validation sets  “This Cityscapes Dataset is made freely available to academic and non-academic entities for non-commercial purposes such as academic research, teaching, scientific publications, or personal experimentation.” (see Note 1.) |
| CityPersons | Image | Uncompressed | <https://www.cityscapes-dataset.com/benchmarks/> | For Citypersons, use defined train and validation sets  “CityPersons, a new set of person annotations on top of the Cityscapes dataset.” for pedestrian detection” (see Note 1.) |
| [KITTI](http://www.cvlibs.net/datasets/kitti/index.php) | Image | Uncompressed | <http://www.cvlibs.net/datasets/kitti/eval_object.php> | We recommend using the predefined splits  “All datasets and benchmarks on this page are copyright by us and published under the [Creative Commons Attribution-NonCommercial-ShareAlike 3.0](http://creativecommons.org/licenses/by-nc-sa/3.0/) License. This means that you must attribute the work in the manner specified by the authors, you may not use this work for commercial purposes and if you alter, transform, or build upon this work, you may distribute the resulting work only under the same license.” (see Note 1.) |
| DAVIS 2016 / 2017 | Video | Compressed | <https://davischallenge.org/> | We recommend using the semi-supervised mode for higher accuracy.  Creative Commons Attributions 4.0 License (non-commercial use) (see Note 1.) |
| MOT20 | Video | Compressed | <https://arxiv.org/pdf/1906.04567.pdf> | Dataset split is available from the Tracking Challenge, available on their website. |
| UCF101 | Video | Compressed | <https://www.crcv.ucf.edu/data/UCF101.php> | Action recognition  13320 videos from 101 action categories. (see Note 1.) |
| HiEve | Video | 14 Uncompressed | <http://humaninevents.org/data.html?title=1> | Human in Event  19 training videos, 13 test videos and a total of 32 videos  No issues with licensing – can provide license/permission to MPEG-VCM  Annotation of the bounding box, ID, key points and behavior, dense pose. Can be used for multi-target tracking  Proponent agreed to provide permission (licensing free) for MPEG-VCM activities at MPEG131 (by 2020-07-03 - Received) |
| SFU-HW-Objects-v1 | Video | Uncompressed |  | Labeled video data  Object labeled dataset on raw video sequences  Already being used for MPEG (HEVC) – appears to be fine for standardization activity  Can be used for compression and object detection simultaneously  Need to be investigated to see whether it large enough to be used for VCM activities  The dataset is provided under the Creative Commons license BY 4.0 (CC BY 4.0)  Proponent agreed to provide permission (licensing free for labels) for MPEG-VCM activities at MPEG131 (by 2020-07-03) |
| Open Images | Image | Compressed | <https://storage.googleapis.com/openimages/web/index.html> | 9M images, 600 classes, Modern.  It contains a total of 16M bounding boxes for 600 object classes on 1.9M images, making it the largest existing dataset with object location annotations.  Listed as using CC BY 2.0 and CC BY 4.0 Licensing terms |
| FLIR Thermal Dataset | Image | Compressed | <https://www.flir.com/oem/adas/adas-dataset-form/> | Ability to sense thermal infrared radiation or heat  “1.1 License. As between you and us, we or our licensors own and reserve all right, title, and interest in and to the Image Data. We grant you a limited, revocable, non-exclusive, non-sublicensable, non-transferable license to access and use the Image Data in the field of neural network development for automotive and other autonomous vehicle applications, and for general non-commercial educational and research purposes. Except as provided in this Section 1, you obtain no rights under this Agreement from us or our licensors to the Image Data, including the right to reproduce, redistribute or make derivative works of the Image Data” |

# Evaluation Methods and Procedures

The evaluation procedure and metrics are described in section 2 above. The metrics consist of two parts, one relating to feature extraction and one relating to compression of processed or unprocessed video. The metrics and anchors for feature extraction will be considered later.

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 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. A 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 a comparison to the released compression ratio of VVC on the unprocessed video.

For human consumption use cases, proponents shall report BD-rate curves. BD-rate should be calculated in the way as other standardization groups, e.g. JVET [3]. The performance reporting format is specified in the MPEG-VCM BD-rate and BD-AP/BD-Accuracy reporting template [4].

* Use case-specific performance metrics, with the key tasks and metrics as defined in Table 1. Proponents shall perform the evaluation themselves, with the experimental 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 a 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 R-CNN or YOLO.

## Proposed Processing Pipelines

Possible pipeline architectures for technology proposal are shown in Figures 1.a, 1.b, and 1.c. Anchors are generated on the pipeline architecture shown in Figure 1.a.

|  |
| --- |
|  |
| 1. Pipeline 1 |
|  |
| 1. Pipeline 2 |
|  |
| 1. Pipeline 3 |

Figure **1**. **Proposed processing pipeline** [5]

## Pre/Post-Processing Data (image) Conversion

For input data processing, it is suggested to use FFmpeg release 4.2.2. FFmpeg can be used for data format conversion, up/down-sampling, and resizing (cropping/padding/scaling) the image.

* + FFmpeg 4.2.2
    - Resolution: Scaling / resolution (100%, 75%, 50%, 25%):
* Scale PNG image to new resolution:

ffmpeg -i input.png -vf “scale=NEW\_WDT:NEW\_HGT“ output.png

for 100%: -vf “pad=ceil(iw/2)\*2:ceil(ih/2)\*2”

for 75%: -vf "scale=ceil(iw\*3/8)\*2:ceil(ih\*3/8)\*2"

for 50%:   -vf "scale=ceil(iw\*/4)\*2:ceil(ih\*/4)\*2"

for 25%: -vf "scale=ceil(iw\*/8)\*2:ceil(ih\*/8)\*2"

* + - Format conversion: Convert PNG 🡨🡪 YUV:

ffmpeg -i input.png -f rawvideo -pix\_fmt yuv420p *-dst\_range 1* output.yuv

ffmpeg -f rawvideo -pix\_fmt yuv420p10le -s WDTxHGT -*src\_range 1 -i* input.yuv -frames 1 -pix\_fmt rgb24 output.png

### Processing Pipeline for Downscaled Resolution

There are two possible processing pipelines [6] for the downscaled ground-truth image as are shown in Figure 1. For anchor generations, an alternative processing pipeline shown in Figure 2.b is recommended.

|  |
| --- |
|  |
| 1. Original Pipeline |
|  |
| 1. Alternative pipeline – Upscaled decoded image |

Figure . Possible pipelines for downscaled resolutions

## Anchor Reference Compression / Decompression

VTM8.2 encoder is used to compress generated YUV files under the default “All Intra” configuration and ConformaceWindowMode set to 1. ConformaceWindowMode is equal to 1 represent padding width and height of processing image automatic to multiple of minimal CU size.

A total of six evenly-spaced QPs are recommended, respectively (17 optional), 22, 27, 32, 37, 42, and 47.

* VVC: Reference software VTM-8.2
* JVET Common Test Conditions (CTC-420) with Random Access (RA) condition for videos
* JVET Common Test Conditions (CTC-420) with All Intra AI condition for images
* encoder -c cfg/encoder\_intra\_vtm.cfg -i input.yuv -o reconstruct.yuv -b compress.vvc -q QP --ConformanceWindowMode=1 -wdt WDT -hgt HGT -f 1 -fr 1 --InternalBitDepth=10
* decoder -b compress.vvc -o decode.yuv

## Anchor Performance Curve Generation

To generate the performance curves of the anchors, mAP/MOTA vs BPP/bitrate curves are to be produced with the following specifications.

* + Specify max. endpoint as the uncompressed performance (mAP/MOTA)
  + Specify min. endpoint performance (mAP/MOTA)
  + Generate performance (mAP/MOTA v.s. BPP) curves for each task based on the 4x resolutions and QPs within the range of min/max endpoints
  + Specify the minimum threshold (i.e., below which the performance results are not acceptable). This threshold is subject to the discretion of the anchor provider.
  + The performance curve is the Pareto-Front created from the generated 4 curves above. Refer to [7] for additional information regarding the creation of the Pareto-Front curve
* Accuracy measurement – mAP/MOTA is used to measure the accuracy performance
  + mAP, [AP@[0.5:0.95](mailto:AP@[0.5:0.95)]
  + MOTA
* Compression efficiency measurement – BPP/bitrate is used to measure the cost for storage/transmission of the generated bitstream for VCM
  + BPP calculation: BPP is calculated with respect to the original image resolution (not the downscales image)
* For machine/human-vision tasks, BD-rate curves (mAP/MOTA, SSIM, PSNR vs. BPP/bitrate) should be calculated in the way as other standardization groups, e.g. JVET [3].
* The performance curves shall be compared against the anchor performance curves using the BD-rate and BD-PSNR functions in the MPEG-VCM reporting template [4].

# Evaluation Approach for Machine and Human Consumption

The evaluation process of video test data set for machine consumption and human consumption using VVC as an anchor is shown in Figure 3 and Figure 4, respectively.

|  |
| --- |
|  |

Figure . Evaluation approach of video test data using VVC anchor - machine consumption

|  |
| --- |
|  |

Figure . Evaluation approach of video test data using VVC anchor - human consumption

The feature data type and format information are beneficial for compression experts to know the properties and limitations of the feature data types and formats to increase the quality of their proposals. The list of feature data types and formats is recommended to be as exhaustive as possible and to include all relevant information such as allowed values and data ranges.

Table 3 shows an overview of different data types for different tasks required by the use cases described in [1].

Table . *Feature data types and their description for various tasks*

|  |  |  |
| --- | --- | --- |
| **Task** | **Type of data** | **Description** |
| Object detection | List of bounding boxes | Maximum number of bounding boxes: TBD  Each bounding box has four attributes:   * pos\_x (integer): offset from left picture edge: 0 to MAX\_PIC\_WIDTH * pos\_y (integer): offset from top picture edge: 0 to MAX\_PIC\_HEIGHT * size\_x (integer): width from left edge of bounding box: 1 to MAX\_PIC\_WIDTH * size\_y (integer): height from top edge of bounding box: 1 to MAX\_PIC\_HEIGHT |
| Object segmentation | Matrix | Matrix size: INPUT\_WIDTH x INPUT\_HEIGHT  All elements of the matrix are either a single integer value or a list of three integer values (for different color formats). The range of the values depends on the chosen bit depth. |
| Object tracking | List of bounding boxes | Maximum number of bounding boxes: TBD  Each bounding box has five attributes:   * pos\_x (integer): offset from left picture edge: 0 to MAX\_PIC\_WIDTH * pos\_y (integer): offset from top picture edge: 0 to MAX\_PIC\_HEIGHT * size\_x (integer): width from left edge of bounding box: 1 to MAX\_PIC\_WIDTH * size\_y (integer): height from top edge of bounding box: 1 to MAX\_PIC\_HEIGHT   box\_id (integer): identifier for each box to allow tracking through multiple frames |

## Complexity Measurement

Encoder/decoder runtime has served as a reasonable proxy for computational complexity in JVET / VVC standardization projects. It is recommended the runtime metric [8] to be reported for anchors generation in Machine vision and Human vision tasks.

The complexity of the codec shall be independent of pre-processing methods and NN tasks used. For the recommended method to measure the codec complexity see Appendix B.

# References

|  |  |
| --- | --- |
| [1] | m55597, "Use cases and draft requirements for Video Coding for Machines," Online, Oct 2020. |
| [2] | L. Tsung-Hua, N. Yu-Chieh, (Foxconn) and L. Deng-Rung, "m55242-[VCM] Anchor result for object segmentation on COCO," Oct 2020. |
| [3] | K. Andersson, F. Bossen, J.-R. Ohm, A. Segall, R. Sjöberg, J. Ström and G. J. Sullivan, "Summary information on BD-rate experiment evaluation practices. JVET-Q2016 (w19168)," January 2020.. |
| [4] | S. Liu, W. Gao, X. X. (Tencent), S.-P. Wang, C.-C. L. (ITRI) and T.-H. L. (Foxconn), "m55583\_[VCM] Common test conditions, evaluation methodology and reporting template for VCM," Oct 2020. |
| [5] | L. Yuchao Shao, "m54366 [VCM] Coding Experiments of End-to-end Compression Network in VCM," June 2020. |
| [6] | C. Hollmann, L. Litwic and M. Damghanian, "m55038-[VCM] Anchor results for KITTI dataset," Oct 2020. |
| [7] | H. Zhang, N. Le, R. Ghaznavi-Youvalari, F. Cricri, H. R. Tavakoli and M. Hannuksela, "m55572-[VCM] On computing the Pareto front for VCM anchors," Oct 2020. |
| [8] | K. Misra, P. Cowan and T. J. a. A. Segall, "m54474 [VCM] Comments on the VCM evaluation framework," June 2020. |
| [9] | T. Ji, P. Cowan, K. Misra and S. Andrew, "m54859-[VCM] Anchor results for object detection on Open Images," Oct 2020. |
| [10] | W. Gao, W. Wang and S. Liu, "m55256-[VCM] Crosscheck for m54859 - Anchor results for object detection on Open," Oct 2020. |
| [11] | W. Sheng-Po, K. Erh-Chung, L. C. Chieh and C.-L. Lin, "m55317 [VCM] Anchor generation results for object detection on COCO dataset," Oct. 2020. |
| [12] | T.-H. Li, Y.-C. Nien, (Foxconn) and D.-R. Liu, "[VCM] Crosscheck of m54861 (Anchor generation results for object detection on COCO dataset)," Oct 2020. |
| [13] | N. Le, R. Ghaznavi-Youvalari, A. Zare, H. Zhang, F. Cricri, H. R. Tavakoli, E. Aksu and M. Hannuksela, "[VCM] VCM anchors for object detection and object segmentation tasks," Oct 2020. |
| [14] | K. Yoon, S. Kim, Y. L. University), H. Lim, H.-G. Choo, H. K. Lee and J. Seo(ETRI), "m55138-[VCM] Anchor results for object detection in IR image," Oct 2020. |
| [15] | H. Lim, H.-G. Choo, H. K. Lee, W.-S. Cheong and J. S. (ETRI), "m55457-[VCM] Crosscheck of m55138 (Anchor results on object detection for FLIR dataset)," Oct 2020. |
| [16] | S.-P. Wang, E.-C. Ke, C.-C. Lin and C.-L. L. (ITRI), "m55398-[VCM] Crosscheck of m55242 (Anchor results for object segmentation on COCO dataset)," Oct 2020. |
| [17] | W. Gao, W. Wang and S. Liu, "m55257-[VCM] Cross-check for m55038 Anchor results for KITTI dataset," Oct 2020. |
| [18] | J. Yun, S. Kwak, W.-S. Cheong and J. S. (ETRI), "[VCM] Anchor crosscheck for object segmentation on Cityscapes dataset (crosscheck of m54886)," Oct 2020. |
| [19] | K. Dong, W. L. Tao Wang and A. Zhang, "m55270-Anchor generation for MOT20," Oct 2020. |

# Appendix A: Anchor Generation Results

## Object Detection

### Sharp vs. Tencent (Open Images)

|  |  |  |
| --- | --- | --- |
| **Sharp** [9] **, Tencent (Crosschecker)** [10] | | |
| Network model: Faster R-CNN X101-FPN | | Dataset: OpenImages |
| Test Condition: | | * Format conversion: FFmpeg 4.2.2 * Codec: VTM 8.2 * Scaling resolution: 100%, 75%, 50% and 25% * QPs: 22, 27, 32, 37, 42, 47 * Average precision: mAP, AP@[0.5:0.95]; * Compression ratio: BPP |
| A screenshot of a cell phone  Description automatically generated | | |
| Test Results: | * Average precision vs. BPP * Max. mAP (uncompressed) is 45.861 | |
| Figure - Updated Rate-Performance of Anchor result for different scaling factors | | |
| Table – Updated mAP and bpp for different QPs and scaling factors   |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | |  | Scale: 100% | | Scale: 75% | | Scale: 50% | | Scale: 25% | | | QP | mAP | bpp | mAP | bpp | mAP | bpp | mAP | bpp | | 22 | 45.2596 | 0.7682 | 45.1107 | 0.4675 | 44.4228 | 0.2522 | 39.5117 | 0.0863 | | 27 | 44.6953 | 0.4427 | 44.1699 | 0.2754 | 42.6831 | 0.1535 | 36.115 | 0.0548 | | 32 | 43.2653 | 0.2449 | 41.5874 | 0.1557 | 39.0697 | 0.0883 | 30.9118 | 0.0328 | | 37 | 40.0281 | 0.1285 | 37.4829 | 0.0836 | 33.5822 | 0.0479 | 23.2994 | 0.0186 | | 42 | 33.6812 | 0.0651 | 30.7869 | 0.0433 | 25.4618 | 0.0253 | 14.8521 | 0.0103 | | 47 | 24.5483 | 0.0315 | 20.4969 | 0.0215 | 14.9104 | 0.0129 | 7.5669 | 0.0057 | | | |

### ITRI vs. Foxconn (COCO)

|  |  |  |
| --- | --- | --- |
| **ITRI** [11]**, Foxconn (Crosschecker)** [12] | | |
| Network model: Faster R-CNN X101-FPN | | Dataset: COCO 2017 |
| Test Condition: | | * Format conversion: FFmpeg 4.2.2 * Codec: VTM 8.2 * Scaling resolution: 100%, 75%, 50% and 25% * QPs: 22, 27, 32, 37, 42, 47, 52 * Average precision: AP, AP50, AP75, APs, APm, APl * Compression ratio: BPP |
|  | | |
| Test Results: | * Average precision vs. BPP * Large target: APl * Medium target: APm * Small target: APs | |

|  |
| --- |
| Anchor generation result of object detection on COCO 2017 val. dataset |
| Average precision for large targets (APl) |
| Average precision for medium targets (APm) |
| Average precision for small targets (APs) |
| |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Scale** | **QP** | **AP** | **AP50** | **AP75** | **APs** | **APm** | **APl** | **bpp** | | 100% | 22 | 41.5045 | 61.7009 | 45.0437 | 26.1824 | 44.4832 | 52.7675 | 1.5963184 | | 27 | 40.4758 | 60.6376 | 43.7608 | 24.3928 | 43.4715 | 52.3682 | 1.0053233 | | 32 | 37.7885 | 57.2687 | 40.6742 | 21.8702 | 40.4005 | 50.2088 | 0.5800124 | | 37 | 32.298 | 50.3367 | 34.2423 | 16.7662 | 34.5564 | 44.7244 | 0.307784 | | 42 | 24.0074 | 38.7239 | 25.4864 | 10.5384 | 26.0294 | 35.9852 | 0.1466404 | | 47 | 13.3754 | 22.2007 | 13.6436 | 3.3998 | 13.1983 | 23.5616 | 0.0617786 | | 52 | 4.9413 | 8.5412 | 4.8654 | 0.5739 | 3.8083 | 10.4418 | 0.0265837 | | 75% | 22 | 39.6599 | 59.5928 | 43.2245 | 23.0789 | 42.7946 | 52.804 | 0.8306602 | | 27 | 38.0783 | 57.6917 | 40.8564 | 21.6167 | 41.0959 | 51.0536 | 0.5245268 | | 32 | 34.3469 | 52.9217 | 36.5852 | 18.409 | 37.1146 | 46.8496 | 0.3080359 | | 37 | 28.0185 | 44.4437 | 29.7056 | 12.6252 | 30.1045 | 41.4095 | 0.164811 | | 42 | 19.1538 | 31.2732 | 20.0212 | 6.6961 | 20.208 | 31.3728 | 0.0810905 | | 47 | 9.3642 | 15.9783 | 9.3189 | 1.5974 | 8.4252 | 18.3333 | 0.0372565 | | 52 | 3.1372 | 5.494 | 3.1085 | 0.2618 | 1.9217 | 7.0911 | 0.0174967 | | 50% | 22 | 34.9565 | 53.6583 | 37.1223 | 17.0158 | 38.1277 | 49.3218 | 0.3925818 | | 27 | 32.6094 | 50.2893 | 34.7585 | 15.8032 | 35.3038 | 47.0289 | 0.2531014 | | 32 | 27.6571 | 43.7859 | 29.0775 | 11.6163 | 29.8321 | 42.4014 | 0.1507692 | | 37 | 20.6278 | 33.2851 | 21.48 | 7.0188 | 21.3607 | 34.0829 | 0.0822794 | | 42 | 12.1992 | 20.3155 | 12.3755 | 2.3539 | 11.6503 | 22.9786 | 0.0422947 | | 47 | 5.2227 | 9.0581 | 5.2285 | 0.5339 | 3.7093 | 11.3101 | 0.0206868 | | 52 | 1.6082 | 2.8585 | 1.5732 | 0.042 | 0.6539 | 3.6317 | 0.0103904 | | 25% | 22 | 20.1592 | 32.6318 | 20.8702 | 4.5236 | 20.2629 | 35.3964 | 0.1202626 | | 27 | 17.4383 | 28.5217 | 18.0351 | 3.2577 | 17.2756 | 31.5521 | 0.0802068 | | 32 | 13.4498 | 22.2494 | 13.8305 | 1.9013 | 12.4723 | 26.4705 | 0.0496675 | | 37 | 8.4106 | 14.3478 | 8.5317 | 0.84 | 6.8332 | 17.3023 | 0.0287331 | | 42 | 3.9971 | 7.0998 | 3.9173 | 0.2405 | 2.6848 | 9.0565 | 0.0160057 | | 47 | 1.5447 | 2.7375 | 1.5408 | 0.0136 | 0.669 | 3.5466 | 0.0087154 | | 52 | 0.4819 | 0.9142 | 0.43 | 0.0006 | 0.1625 | 1.097 | 0.0051481 | |
|  |

### Nokia vs. (ETRI) (CityPersons)

|  |  |  |
| --- | --- | --- |
| **Nokia** [13] **(ETRI) (Crosschecker)** | | |
| Network model: Faster R-CNN X101-FPN | | Dataset: COCO train2017, CityPersons (Eval) |
| Test Condition: | | * Format conversion: FFmpeg 4.2.2 * Codec: VTM 8.2 * Scaling resolution: 100%, 75%, 50% and 25% * QPs: 22, 27, 32, 37, 42, 47 * Average precision: AP [.5:.05:.95], Range: [0,100] * Compression ratio: BPP |
| Processing flow for anchor generation | | |
| Test Results: | * Average precision vs. BPP * Test results need to be confirmed by a crosschecker (ETRI) | |
| Rate-Performance of Anchors for object detection task | | |
|  | | |
| Table 1. BPP and AP results for different QPs and resolutions in Object detection task   |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | |  | Resolution 100% | | Resolution 75% | | Resolution 50% | | Resolution 25% | | | QP | BPP | mAP | BPP | mAP | BPP | mAP | BPP | mAP | | 22 | 0.3698 | 25.3551 | 0.2485 | 24.7935 | 0.1510 | 23.7719 | 0.0587 | 20.2158 | | 27 | 0.2045 | 24.3321 | 0.1436 | 23.9869 | 0.0900 | 22.7785 | 0.0367 | 18.4755 | | 32 | 0.1138 | 23.1985 | 0.0819 | 22.2892 | 0.0519 | 20.2584 | 0.0217 | 14.9729 | | 37 | 0.0633 | 21.4129 | 0.0457 | 19.5308 | 0.0287 | 16.5814 | 0.0120 | 9.0977 | | 42 | 0.0348 | 17.3505 | 0.0249 | 14.8551 | 0.0153 | 10.8454 | 0.0063 | 4.3328 | | 47 | 0.0183 | 11.5329 | 0.0129 | 8.6519 | 0.0078 | 4.8076 | 0.0031 | 1.3825 | | | |

### Konkuk Univ vs. ETRI (FLIR)

|  |  |  |
| --- | --- | --- |
| **Konkuk Univ** [14]**, ETRI (Crosschecker)** [15] | | |
| Network model: Faster R-CNN X101-FPN | | Dataset: FREE FLIR Thermal dataset |
| Test Condition: | | * Format conversion: FFmpeg 4.2.2 * Codec: VTM 8.2 * Scaling resolution: 100%, 75%, 50% and 25% * QPs: 22, 27, 32, 37, 42, 47 * Average precision: mAP[.5:.05:.95] * Compression ratio: BPP |
| Figure 1 Processing Flow of Anchor generation for object detection on FLIR dataset | | |
| Test Results: | * Average precision vs. BPP * 600 images (300 RGB images and 300 IR images) from FLIR Dataset | |
| Anchor generation results of pre-trained object detection network in thermal images | | |
| Anchor generation results of pre-trained object detection network in thermal images   |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Scale** | **QP** | **mAP** | **AP50** | **AP75** | **APs** | **APm** | **APl** | **BPP** | | **original** |  | 29.647 | 56.441 | 27.172 | 13.556 | 41.307 | 66.916 | 6.861 | | **100%** | 17 | 27.224 | 52.399 | 23.707 | 11.444 | 38.829 | 65.254 | 2.476 | | 22 | 29.383 | 56.417 | 26.955 | 13.314 | 40.703 | 68.281 | 1.807 | | 27 | 29.097 | 55.864 | 26.454 | 13.557 | 39.896 | 67.340 | 1.224 | | 32 | 27.799 | 52.789 | 25.349 | 13.565 | 38.547 | 61.477 | 0.306 | | 37 | 20.628 | 39.474 | 18.670 | 7.934 | 28.960 | 56.310 | 0.131 | | 42 | 10.096 | 19.125 | 8.544 | 2.412 | 15.290 | 33.322 | 0.065 | | 47 | 2.908 | 5.949 | 2.618 | 0.778 | 4.622 | 8.487 | 0.030 | | **75%** | 17 | 26.949 | 50.999 | 24.943 | 15.696 | 51.340 | 71.264 | 1.376 | | 22 | 26.144 | 50.329 | 24.431 | 15.242 | 49.094 | 70.098 | 0.888 | | 27 | 25.456 | 48.849 | 22.788 | 14.686 | 48.344 | 68.697 | 0.400 | | 32 | 21.933 | 42.046 | 19.224 | 12.463 | 42.420 | 61.537 | 0.189 | | 37 | 15.765 | 30.048 | 15.735 | 7.582 | 33.681 | 49.680 | 0.098 | | 42 | 5.645 | 12.116 | 4.755 | 1.852 | 14.017 | 22.353 | 0.049 | | 47 | 1.494 | 2.714 | 1.223 | 0.361 | 3.714 | 3.366 | 0.022 | | **50%** | 17 | 21.249 | 38.880 | 20.059 | 15.125 | 55.255 | 79.285 | 0.578 | | 22 | 21.140 | 38.001 | 20.044 | 15.106 | 55.260 | 82.104 | 0.352 | | 27 | 18.996 | 34.471 | 17.739 | 13.209 | 50.143 | 79.203 | 0.193 | | 32 | 15.567 | 28.998 | 15.550 | 10.052 | 44.901 | 73.014 | 0.107 | | 37 | 8.280 | 15.452 | 8.481 | 5.018 | 25.930 | 53.103 | 0.056 | | 42 | 2.010 | 4.652 | 1.416 | 1.131 | 7.092 | 14.262 | 0.027 | | 47 | 0.469 | 0.660 | 0.495 | 0.215 | 0.967 | 6.436 | 0.012 | | **25%** | 17 | 8.396 | 14.833 | 7.799 | 7.036 | 49.819 | NaN | 0.149 | | 22 | 7.870 | 13.977 | 8.728 | 6.535 | 49.033 | NaN | 0.100 | | 27 | 6.728 | 12.068 | 7.018 | 5.528 | 41.174 | NaN | 0.064 | | 32 | 4.840 | 9.680 | 5.222 | 4.036 | 29.287 | NaN | 0.038 | | 37 | 1.567 | 3.834 | 0.851 | 1.377 | 6.595 | NaN | 0.020 | | 42 | 0.473 | 0.762 | 0.660 | 0.103 | 2.335 | NaN | 0.010 | | 47 | 0.001 | 0.011 | 0.000 | 0.002 | 0.000 | NaN | 0.005 | | | |
| Anchor generation results of pre-trained object detection network in thermal images | | |
| Anchor generation results of pre-trained object detection network in RGB images   |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Scale** | **QP** | **mAP** | **AP50** | **AP75** | **APs** | **APm** | **APl** | **BPP** | | **original** |  | 22.248 | 41.889 | 20.122 | 7.271 | 32.948 | 61.818 | 15.403 | | **100%** | 17 | 21.829 | 42.400 | 19.138 | 7.087 | 31.971 | 63.684 | 1.556 | | 22 | 22.196 | 43.773 | 19.591 | 7.279 | 33.410 | 60.975 | 0.576 | | 27 | 20.685 | 41.125 | 17.716 | 5.972 | 31.985 | 57.737 | 0.236 | | 32 | 17.522 | 34.498 | 13.889 | 4.888 | 25.885 | 54.186 | 0.116 | | 37 | 11.528 | 21.961 | 10.238 | 2.804 | 17.024 | 36.155 | 0.064 | | 42 | 3.551 | 8.073 | 2.715 | 0.799 | 5.139 | 15.482 | 0.036 | | 47 | 0.847 | 1.733 | 0.675 | 0.187 | 0.628 | 6.455 | 0.019 | | **75%** | 17 | 22.447 | 42.564 | 21.490 | 12.566 | 46.057 | 62.778 | 0.881 | | 22 | 21.772 | 41.896 | 19.166 | 12.916 | 44.238 | 57.718 | 0.448 | | 27 | 18.660 | 37.070 | 15.331 | 10.570 | 38.103 | 55.017 | 0.194 | | 32 | 15.448 | 31.359 | 12.149 | 8.867 | 32.126 | 41.499 | 0.092 | | 37 | 8.265 | 17.100 | 6.639 | 4.222 | 17.092 | 30.616 | 0.050 | | 42 | 2.336 | 5.346 | 1.605 | 1.058 | 4.713 | 11.926 | 0.028 | | 47 | 0.468 | 0.857 | 0.519 | 0.293 | 0.573 | 4.657 | 0.015 | | **50%** | 17 | 19.371 | 36.962 | 16.757 | 13.314 | 53.801 | 63.209 | 0.373 | | 22 | 17.815 | 33.714 | 17.407 | 12.286 | 49.701 | 52.588 | 0.189 | | 27 | 14.869 | 29.301 | 12.201 | 10.215 | 41.666 | 51.469 | 0.097 | | 32 | 9.260 | 19.466 | 8.094 | 6.194 | 27.843 | 29.983 | 0.055 | | 37 | 3.556 | 8.547 | 2.139 | 2.176 | 11.708 | 23.069 | 0.031 | | 42 | 0.967 | 1.966 | 0.962 | 0.361 | 3.492 | 18.438 | 0.017 | | 47 | 0.188 | 0.330 | 0.132 | 0.000 | 0.380 | 9.653 | 0.009 | | **25%** | 17 | 8.008 | 16.107 | 6.308 | 6.916 | 39.897 | NaN | 0.101 | | 22 | 6.581 | 12.444 | 5.604 | 5.68 | 34.524 | NaN | 0.061 | | 27 | 3.516 | 7.806 | 2.384 | 2.946 | 21.575 | NaN | 0.038 | | 32 | 1.718 | 3.445 | 1.347 | 1.427 | 9.229 | NaN | 0.023 | | 37 | 0.667 | 1.172 | 0.693 | 0.494 | 6.116 | NaN | 0.013 | | 42 | 0.231 | 0.33 | 0.33 | 0.231 | 1.061 | NaN | 0.008 | | 47 | 0.044 | 0.141 | 0 | 0.017 | 0.264 | NaN | 0.004 | | | |
|  | | |
| Figure - Anchor generation results of fine-tune object detection network in thermal images | | |
| Table - Anchor generation results of fine-tune object detection network in thermal images   |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Scale** | **QP** | **mAP** | **AP50** | **AP75** | **APs** | **APm** | **APl** | **BPP** | | **original** |  | 40.463 | 77.767 | 36.784 | 30.923 | 46.984 | 64.680 | 6.861 | | **100%** | 17 | 39.126 | 75.336 | 35.734 | 29.596 | 45.512 | 66.563 | 2.476 | | 22 | 39.441 | 76.634 | 35.123 | 30.510 | 45.591 | 61.689 | 1.807 | | 27 | 39.651 | 76.694 | 35.633 | 30.886 | 46.051 | 62.608 | 1.224 | | 32 | 37.772 | 73.476 | 31.449 | 28.588 | 45.252 | 58.788 | 0.306 | | 37 | 33.747 | 69.220 | 27.693 | 24.555 | 41.793 | 52.623 | 0.131 | | 42 | 23.061 | 47.937 | 18.876 | 15.660 | 29.359 | 40.428 | 0.065 | | 47 | 10.616 | 24.111 | 7.591 | 6.067 | 13.970 | 24.376 | 0.030 | | **75%** | 17 | 39.215 | 75.005 | 34.668 | 32.942 | 52.722 | 65.191 | 1.376 | | 22 | 39.591 | 75.909 | 34.172 | 33.342 | 53.274 | 64.369 | 0.888 | | 27 | 38.793 | 74.983 | 32.565 | 32.565 | 52.461 | 64.137 | 0.400 | | 32 | 35.959 | 71.790 | 29.987 | 29.678 | 49.940 | 56.578 | 0.189 | | 37 | 30.794 | 62.316 | 25.339 | 24.943 | 44.431 | 42.681 | 0.098 | | 42 | 19.204 | 40.857 | 14.654 | 14.401 | 31.077 | 28.018 | 0.049 | | 47 | 7.412 | 16.740 | 5.540 | 4.403 | 14.958 | 15.600 | 0.022 | | **50%** | 17 | 34.744 | 66.315 | 31.096 | 30.827 | 57.169 | 75.333 | 0.578 | | 22 | 34.147 | 66.452 | 30.767 | 30.407 | 55.657 | 70.223 | 0.352 | | 27 | 32.945 | 63.719 | 28.882 | 29.321 | 52.820 | 68.895 | 0.193 | | 32 | 29.056 | 57.737 | 24.735 | 25.605 | 48.172 | 61.540 | 0.107 | | 37 | 21.003 | 43.937 | 17.726 | 18.509 | 35.864 | 52.551 | 0.056 | | 42 | 9.712 | 22.778 | 6.849 | 7.995 | 21.247 | 23.577 | 0.027 | | 47 | 2.355 | 5.656 | 1.642 | 1.695 | 6.853 | 7.648 | 0.012 | | **25%** | 17 | 15.614 | 31.236 | 13.243 | 15.020 | 35.299 | NaN | 0.149 | | 22 | 15.239 | 31.949 | 12.853 | 14.649 | 34.580 | NaN | 0.100 | | 27 | 14.155 | 28.747 | 11.132 | 13.682 | 31.517 | NaN | 0.064 | | 32 | 10.323 | 21.818 | 7.468 | 9.939 | 25.327 | NaN | 0.038 | | 37 | 5.675 | 12.441 | 4.206 | 5.332 | 15.415 | NaN | 0.020 | | 42 | 1.262 | 3.053 | 0.852 | 1.114 | 6.284 | NaN | 0.010 | | 47 | 0.099 | 0.306 | 0.041 | 0.084 | 0.480 | NaN | 0.005 | | | |
| Figure - Anchor generation results of fine-tune object detection network in RGB images | | |
| Table - Anchor generation results of fine-tune object detection network in RGB images   |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Scale** | **QP** | **mAP** | **AP50** | **AP75** | **APs** | **APm** | **APl** | **BPP** | | **original** |  | 23.687 | 50.122 | 18.960 | 9.755 | 36.875 | 64.969 | 15.403 | | **100%** | 17 | 23.619 | 49.092 | 19.482 | 9.427 | 36.478 | 63.023 | 1.556 | | 22 | 24.021 | 50.829 | 20.691 | 9.644 | 37.427 | 67.511 | 0.576 | | 27 | 21.694 | 46.376 | 17.395 | 8.181 | 34.091 | 63.118 | 0.236 | | 32 | 17.527 | 37.903 | 14.126 | 5.858 | 27.648 | 53.864 | 0.116 | | 37 | 12.371 | 26.087 | 9.574 | 3.220 | 18.587 | 45.408 | 0.064 | | 42 | 6.162 | 13.835 | 4.238 | 1.044 | 8.374 | 34.405 | 0.036 | | 47 | 2.072 | 4.609 | 1.411 | 0.345 | 2.170 | 15.488 | 0.019 | | **75%** | 17 | 23.607 | 50.581 | 19.667 | 14.814 | 48.066 | 61.895 | 0.881 | | 22 | 22.789 | 48.239 | 18.940 | 13.917 | 47.020 | 65.275 | 0.448 | | 27 | 20.456 | 42.648 | 17.353 | 11.955 | 42.807 | 57.762 | 0.194 | | 32 | 16.873 | 36.185 | 12.970 | 9.737 | 34.341 | 51.590 | 0.092 | | 37 | 10.888 | 24.376 | 8.455 | 4.802 | 24.606 | 41.278 | 0.050 | | 42 | 4.660 | 11.994 | 3.017 | 1.655 | 11.008 | 26.260 | 0.028 | | 47 | 1.387 | 3.094 | 1.086 | 0.384 | 2.780 | 10.379 | 0.015 | | **50%** | 17 | 22.332 | 46.682 | 18.543 | 16.847 | 57.523 | 60.761 | 0.373 | | 22 | 20.241 | 42.061 | 16.381 | 14.854 | 56.100 | 59.964 | 0.189 | | 27 | 17.273 | 36.219 | 14.313 | 12.127 | 50.129 | 51.112 | 0.097 | | 32 | 12.752 | 28.031 | 9.256 | 8.737 | 37.697 | 36.168 | 0.055 | | 37 | 6.098 | 13.539 | 4.026 | 2.895 | 25.648 | 32.260 | 0.031 | | 42 | 2.232 | 5.673 | 1.574 | 0.998 | 8.767 | 23.222 | 0.017 | | 47 | 0.687 | 1.662 | 0.271 | 0.228 | 2.769 | 14.455 | 0.009 | | **25%** | 17 | 11.173 | 24.065 | 9.022 | 9.799 | 51.142 | NaN | 0.101 | | 22 | 9.559 | 20.907 | 7.530 | 8.445 | 42.446 | NaN | 0.061 | | 27 | 7.933 | 16.615 | 6.330 | 6.840 | 40.354 | NaN | 0.038 | | 32 | 3.494 | 9.226 | 2.065 | 3.034 | 16.705 | NaN | 0.023 | | 37 | 1.300 | 2.997 | 1.099 | 0.890 | 12.408 | NaN | 0.013 | | 42 | 0.419 | 1.079 | 0.292 | 0.218 | 4.997 | NaN | 0.008 | | 47 | 0.180 | 0.330 | 0.071 | 0.165 | 1.446 | NaN | 0.004 | | | |
| Figure - Anchor results comparison according to neural network type in RGB, thermal images (Resolution 100% Only) | | |
| Table - Anchor results comparison according to neural network type in RGB, thermal images   |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | **Scale** | **QP** | **RGB** | | | **Thermal** | | | | **Pre-trained** | **Fine-tuned** | **BPP** | **Pre-trained** | **Fine-tuned** | **BPP** | | **mAP** | **mAP** | **mAP** | **mAP** | | **original** |  | 22.248 | 23.687 | 15.403 | 29.647 | 40.463 | 6.861 | | **100%** | 17 | 21.829 | 23.619 | 1.556 | 27.224 | 39.126 | 2.476 | | 22 | 22.196 | 24.021 | 0.576 | 29.383 | 39.441 | 1.807 | | 27 | 20.685 | 21.694 | 0.236 | 29.097 | 39.651 | 1.224 | | 32 | 17.522 | 17.527 | 0.116 | 27.799 | 37.772 | 0.306 | | 37 | 11.528 | 12.371 | 0.064 | 20.628 | 33.747 | 0.131 | | 42 | 3.551 | 6.162 | 0.036 | 10.096 | 23.061 | 0.065 | | 47 | 0.847 | 2.072 | 0.019 | 2.908 | 10.616 | 0.030 | | **75%** | 17 | 22.447 | 23.607 | 0.881 | 26.949 | 39.215 | 1.376 | | 22 | 21.772 | 22.789 | 0.448 | 26.144 | 39.591 | 0.888 | | 27 | 18.660 | 20.456 | 0.194 | 25.456 | 38.793 | 0.400 | | 32 | 15.448 | 16.873 | 0.092 | 21.933 | 35.959 | 0.189 | | 37 | 8.265 | 10.888 | 0.050 | 15.765 | 30.794 | 0.098 | | 42 | 2.336 | 4.660 | 0.028 | 5.645 | 19.204 | 0.049 | | 47 | 0.468 | 1.387 | 0.015 | 1.494 | 7.412 | 0.022 | | **50%** | 17 | 19.371 | 22.332 | 0.373 | 21.249 | 34.744 | 0.578 | | 22 | 17.815 | 20.241 | 0.189 | 21.140 | 34.147 | 0.352 | | 27 | 14.869 | 17.273 | 0.097 | 18.996 | 32.945 | 0.193 | | 32 | 9.260 | 12.752 | 0.055 | 15.567 | 29.056 | 0.107 | | 37 | 3.556 | 6.098 | 0.031 | 8.280 | 21.003 | 0.056 | | 42 | 0.967 | 2.232 | 0.017 | 2.010 | 9.712 | 0.027 | | 47 | 0.188 | 0.687 | 0.009 | 0.469 | 2.355 | 0.012 | | **25%** | 17 | 11.173 | 11.173 | 0.101 | 8.396 | 15.614 | 0.149 | | 22 | 9.559 | 9.559 | 0.061 | 7.870 | 15.239 | 0.100 | | 27 | 7.933 | 7.933 | 0.038 | 6.728 | 14.155 | 0.064 | | 32 | 3.494 | 3.494 | 0.023 | 4.840 | 10.323 | 0.038 | | 37 | 1.300 | 1.300 | 0.013 | 1.567 | 5.675 | 0.020 | | 42 | 0.419 | 0.419 | 0.008 | 0.473 | 1.262 | 0.010 | | 47 | 0.180 | 0.180 | 0.004 | 0.001 | 0.099 | 0.005 | | | |

## Object Segmentation

### Foxconn vs. ITRI (COCO)

|  |  |  |
| --- | --- | --- |
| **Foxconn** [2]**, ITRI (Crosscecker)** [16] | | |
| Network model: R50-FPN-3x COCO | | Dataset: COCO 2017 |
| Test Condition: | | * Format conversion: FFmpeg 4.2.2 * Codec: VTM 8.2 * Scaling resolution: 100%, 75%, 50% and 25% * QPs: 22, 27, 32, 37, 42, 47, 52 * Average precision: AP, AP50, AP75, APs, APm, APl * Compression ratio: BPP |
| Figure - Anchor generation flowchart | | |
| Test Results: | * Average precision vs. BPP * Compared the additional padding methods with the agreed padding method, including padding at the scaling stage or at the conversion YUV format stage, the neighbor padding method at the conversion format stage performs well. | |
| Figure - Anchor generation result of object segmentation on COCO 2017 val. dataset | | |
| Figure - Agreed padding method and additional padding method | | |
| Table - Accuracy results on COCO dataset for semantic segmentation (the agreed method)   |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Resolution | QP | AP | AP50 | AP75 | Aps | APm | APl | bpp | | 100% | 22 | 35.8954 | 56.9149 | 38.2892 | 17.3453 | 38.342 | 51.7019 | 1.5963184 | | 27 | 35.0086 | 55.785 | 37.1368 | 17.2117 | 37.188 | 50.9669 | 1.0053233 | | 32 | 32.7259 | 52.8685 | 34.642 | 15.4039 | 34.6604 | 49.0097 | 0.5800124 | | 37 | 28.3282 | 46.7013 | 29.735 | 11.6446 | 29.5772 | 44.3323 | 0.3077840 | | 42 | 21.3343 | 36.2714 | 21.7672 | 7.5664 | 22.1229 | 36.6821 | 0.1466404 | | 47 | 12.2341 | 21.85 | 11.9048 | 2.6973 | 11.4247 | 24.3233 | 0.0617786 | | 52 | 4.574 | 8.4714 | 4.308 | 0.3189 | 3.3906 | 10.8719 | 0.0265837 | | 75% | 22 | 34.3549 | 54.8968 | 36.7708 | 15.7018 | 36.4776 | 50.796 | 0.8306602 | | 27 | 33.0555 | 53.0177 | 35.3624 | 15.1245 | 34.8609 | 50.0527 | 0.5245268 | | 32 | 30.0056 | 49.0714 | 31.7018 | 12.415 | 31.5465 | 46.3752 | 0.3080359 | | 37 | 24.5255 | 40.8674 | 25.6141 | 8.7848 | 25.2681 | 40.9896 | 0.1648110 | | 42 | 17.0244 | 29.492 | 17.1511 | 4.8059 | 17.0492 | 31.7803 | 0.0810905 | | 47 | 8.812 | 15.953 | 8.3569 | 1.152 | 7.1893 | 19.4655 | 0.0372565 | | 52 | 2.9593 | 5.6201 | 2.5731 | 0.1632 | 1.7467 | 6.8863 | 0.0174967 | | 50% | 22 | 30.4602 | 49.5458 | 31.9967 | 11.3855 | 32.0252 | 48.3981 | 0.3925818 | | 27 | 28.5179 | 46.9001 | 30.0109 | 10.4357 | 29.618 | 46.2861 | 0.2531014 | | 32 | 24.556 | 41.339 | 25.5556 | 7.6508 | 25.1722 | 42.1428 | 0.1507692 | | 37 | 18.2083 | 31.4779 | 18.319 | 4.3462 | 18.0238 | 33.781 | 0.0822794 | | 42 | 11.0876 | 19.8769 | 10.8835 | 1.6025 | 9.9814 | 23.4225 | 0.0422947 | | 47 | 4.7676 | 9.0771 | 4.4888 | 0.3815 | 3.0838 | 11.3374 | 0.0206868 | | 52 | 1.4537 | 2.8463 | 1.2766 | 0.0462 | 0.4958 | 3.8725 | 0.0103904 | | 25% | 22 | 17.7768 | 30.6783 | 17.7163 | 2.9581 | 16.6493 | 34.6377 | 0.1202626 | | 27 | 15.6993 | 27.3466 | 15.6346 | 2.3916 | 14.4532 | 31.2724 | 0.0802068 | | 32 | 11.8796 | 21.1704 | 11.7559 | 1.2976 | 10.0654 | 24.88 | 0.0496675 | | 37 | 7.4785 | 13.8503 | 7.1387 | 0.5471 | 5.5108 | 17.4953 | 0.0287331 | | 42 | 3.7615 | 7.2329 | 3.3269 | 0.228 | 2.0676 | 9.5802 | 0.0160057 | | 47 | 1.457 | 2.8606 | 1.2551 | 0.0287 | 0.4543 | 3.714 | 0.0087154 | | 52 | 0.4518 | 0.9223 | 0.3848 | 0.0078 | 0.0748 | 1.0836 | 0.0051481 | | | |
|  | | |

### Ericsson vs. Tencent (KITTI)

|  |  |  |
| --- | --- | --- |
| **Ericsson** [6]**, Tencent (Crosscecker)** [17] | | |
| Network model: R50-FPN-3x COCO | | Dataset: KITTI |
| Test Condition: | | * Semantic segmentation * Instance segmentation * Format conversion: FFmpeg 4.2.2 * Codec: VTM 8.2 * Scaling resolution: 100%, 75%, 50% and 25% * InternalBitDepth=10 * QP: 22, 27, 32, 37, 42, 47 |
| Processing pipeline for anchor generation | | |
| Test Results: | * Average precision vs. BPP * Semantic segmentation * Instance segmentation | |
| Figure - Accuracy of semantic segmentation over bitrate | | |
| Table 1. Accuracy results of semantic segmentation   |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Accu-racy | Un-compressed | QP17 | QP22 | QP27 | QP32 | QP37 | QP42 | QP47 | | 100% | 84.25% | 83.99% | 83.78% | 83.41% | 82.24% | 80.00% | 74.72% | 62.67% | | 75% |  | 83.86% | 83.73% | 82.93% | 81.72% | 77.89% | 70.76% | 55.25% | | 50% |  | 83.58% | 83.29% | 82.32% | 79.22% | 73.90% | 61.77% | 39.99% | | 25% |  | 81.67% | 80.65% | 78.26% | 72.85% | 60.36% | 38.08% | 22.25% | | | |
| Table 2. Bitrate of compressed bitstreams in BPP relative to the original size of the image   |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 100% | QP17 | QP22 | QP27 | QP32 | QP37 | QP42 | QP47 | | 100% | 2.062 | 1.280 | 0.754 | 0.416 | 0.220 | 0.115 | 0.057 | | 75% | 1.214 | 0.776 | 0.472 | 0.269 | 0.147 | 0.077 | 0.038 | | 50% | 0.609 | 0.404 | 0.255 | 0.151 | 0.084 | 0.044 | 0.022 | | 25% | 0.178 | 0.124 | 0.082 | 0.051 | 0.030 | 0.016 | 0.008 | | | |
| Figure - Average Precision for instance segmentation over bitrate | | |
| Table 3. Average Precision results for instance segmentation   |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | AP | Un-compressed | QP17 | QP22 | QP27 | QP32 | QP37 | QP42 | QP47 | | 100% | 22.56% | 20.21% | 21.73% | 19.80% | 17.44% | 10.82% | 8.39% | 3.57% | | 75% |  | 18.30% | 15.31% | 16.46% | 11.88% | 9.31% | 5.53% | 1.94% | | 50% |  | 16.03% | 14.32% | 12.05% | 8.51% | 6.19% | 4.03% | 0.68% | | 25% |  | 5.42% | 5.11% | 4.39% | 2.12% | 0.94% | 0.42% | 0.08% | | | |
| Figure – Pareto-Front Semantic Segmentation | | |
| Figure – Pareto-Front Instance Segmentation | | |

### Nokia vs. ETRI (CityScape)

|  |  |  |
| --- | --- | --- |
| **Nokia** [13]**, ETRI (Crosschecker)** [18] | | |
| Network model: R50-FPN | | Dataset: COCO train2017, Cityscapes (Eval) |
| Test Condition: | | * Format conversion: FFmpeg 4.2.2 * Codec: VTM 8.2 * Scaling resolution: 100%, 75%, 50% and 25% * QPs: 22, 27, 32, 37, 42, 47 * Average precision: AP [.5:.05:.95], Range: [0,100] * Compression ratio: BPP |
|  | | |
| Test Results: | * Average precision vs. BPP | |

|  |
| --- |
| Figure - Rate-Performance of Anchors for Object Segmentation task |
|  |
| Table - BPP and AP results for different QPs and resolutions in Object Segmentation task   |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | |  | Resolution 100% | | Resolution 75% | | Resolution 50% | | Resolution 25% | | | QP | BPP | mAP | BPP | mAP | BPP | mAP | BPP | mAP | | 22 | 0.3698 | 34.4493 | 0.2485 | 32.5882 | 0.1510 | 29.5958 | 0.0587 | 17.1971 | | 27 | 0.2045 | 32.2868 | 0.1436 | 30.1543 | 0.0900 | 24.7600 | 0.0367 | 13.3322 | | 32 | 0.1138 | 28.3342 | 0.0819 | 24.8302 | 0.0519 | 19.5137 | 0.0217 | 8.4633 | | 37 | 0.0633 | 21.4306 | 0.0457 | 17.5714 | 0.0287 | 12.7333 | 0.0120 | 4.0694 | | 42 | 0.0348 | 14.2049 | 0.0249 | 10.7195 | 0.0153 | 6.6346 | 0.0063 | 1.7918 | | 47 | 0.0183 | 6.6726 | 0.0129 | 4.2834 | 0.0078 | 2.2601 | 0.0031 | 0.2973 | |
| Number of bits for generated bitstreams   |  |  |  |  |  | | --- | --- | --- | --- | --- | | QP | Res. 100 | Res. 75 | Res. 50 | Res. 25 | | 22 | 387779072 | 260606936 | 158336608 | 61599496 | | 27 | 214463120 | 150533112 | 94330944 | 38531384 | | 32 | 119281424 | 85906328 | 54386200 | 22707384 | | 37 | 66340504 | 47932560 | 30046696 | 12581944 | | 42 | 36531320 | 26105344 | 16029976 | 6618608 | | 47 | 19168192 | 13523944 | 8165024 | 3292960 | |

## Object Tracking (Video)

### Shanghai Jiao Tong Univ (MOT20)

|  |  |  |
| --- | --- | --- |
| **Shanghai Jiao Tong Univ** [19] **(no crosschecker)** | | |
| Network model: JDE-1088x608 | | Dataset: MOT20 training (MOT20-01, MOT20-02, MOT20-03, MOT20-05) |
| Test Condition: | | * Format conversion: FFmpeg 4.2.2 * Codec: VTM 8.2 * Scaling resolution: 100%, 75%, 50% and 25% * QPs: 22, 27, 32, 37, 42, 47, 52 * Average precision: MOTA (MOT20-01, MOT20-02, MOT20-03, MOT20-05) * Compression ratio: BPP |
| Test Results: | * Test results need to be confirmed by a crosschecker (ETRI) * MOTA vs bpp | |
| x100 resolution result of MOT20-01 video. MOTA baseline of the MOT20-01 video with JDE-1088x608 is 61.91 | | |
| x100 resolution result of MOT20-02 video. The MOTA baseline of the MOT20-02 video with JDE-1088x608 is 60.75 | | |
| x100 resolution result of MOT20-03 video. The MOTA baseline of the MOT20-03 video with JDE-1088x608 is 57.92 | | |
| x100 resolution result of MOT20-05 video. The MOTA baseline of the MOT20-05 video with JDE-1088x608 is 41.80 | | |
| Table of MOTA results of every video   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | MOTA | MOT20-01 | MOT02-02 | MOT20-03 | MOT20-05 | overall | | original | 61.91 | 60.75 | 57.92 | 41.80 | 49.19 | | 25-22 | -11.34 | -10.98 | -0.75 | -3.79 | -4.06 | | 25-27 | -7.54 | -7.51 | -0.35 | -2.47 | -2.66 | | 25-32 | -3.32 | -4.34 | -0.11 | -1.23 | -1.38 | | 25-37 | -0.99 | -1.65 | -0.01 | -0.40 | -0.48 | | 25-42 | -0.15 | -0.67 | 0.00 | -0.12 | -0.16 | | 25-47 | -0.07 | -0.19 | 0.00 | -0.02 | -0.04 | | 25-52 | -0.01 | 0.00 | 0.00 | 0.00 | 0.00 | | 50-22 | -66.46 | -62.72 | -56.02 | -37.04 | -46.31 | | 50-27 | -63.25 | -59.40 | -51.50 | -32.87 | -42.17 | | 50-32 | -57.46 | -53.20 | -43.27 | -25.13 | -34.54 | | 50-37 | -46.77 | -42.38 | -30.30 | -14.45 | -23.21 | | 50-42 | -34.37 | -29.31 | -14.76 | -5.80 | -11.99 | | 50-47 | -19.61 | -14.08 | -3.10 | -1.33 | -3.88 | | 50-52 | -4.11 | -3.71 | -0.18 | -0.22 | -0.75 | | 75-22 | -55.04 | -49.01 | -52.80 | -29.80 | -39.22 | | 75-27 | -53.67 | -47.18 | -49.89 | -28.13 | -37.19 | | 75-32 | -50.11 | -43.88 | -45.18 | -23.69 | -32.85 | | 75-37 | -44.65 | -37.73 | -36.60 | -16.73 | -25.57 | | 75-42 | -34.98 | -28.68 | -24.24 | -8.99 | -16.35 | | 75-47 | -22.86 | -16.68 | -9.67 | -2.75 | -6.91 | | 75-52 | -9.07 | -6.17 | -1.43 | -0.44 | -1.65 | | 100-22 | 59.65 | 58.23 | 53.62 | 31.87 | 27.14 | | 100-27 | 58.77 | 57.64 | 52.34 | 31.21 | 26.67 | | 100-32 | 57.15 | 55.86 | 49.06 | 28.42 | 24.81 | | 100-37 | 52.60 | 51.44 | 43.07 | 22.48 | 20.74 | | 100-42 | 46.39 | 44.37 | 32.86 | 11.17 | 13.23 | | 100-47 | 32.86 | 31.69 | 17.20 | 5.68 | 8.14 | | 100-52 | 16.05 | 14.96 | 4.11 | 0.27 | 2.48 | | | |
| Table of MOTP results of every video   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | MOTP | MOT20-01 | MOT02-02 | MOT20-03 | MOT20-05 | overall | | original | 74.97 | 75.88 | 71.67 | 73.19 | 73.19 | | 25-22 | 0.00 | 0.00 | 0.00 | 57.69 | 57.69 | | 25-27 | 0.00 | 0.00 | 0.00 | 55.45 | 55.45 | | 25-32 | 0.00 | 0.00 | 0.00 | 54.22 | 54.22 | | 25-37 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | | 25-42 | 0.00 | 0.00 | 0.00 | 52.56 | 52.56 | | 25-47 | 0.00 | 0.00 | 0.00 | 56.49 | 56.49 | | 25-52 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | | 50-22 | 0.00 | 59.40 | 55.24 | 56.86 | 58.04 | | 50-27 | 0.00 | 59.47 | 55.89 | 56.92 | 58.15 | | 50-32 | 0.00 | 59.50 | 55.72 | 57.01 | 58.09 | | 50-37 | 50.69 | 59.43 | 57.45 | 57.59 | 58.58 | | 50-42 | 51.73 | 59.23 | 58.43 | 57.75 | 58.77 | | 50-47 | 0.00 | 59.77 | 58.20 | 60.23 | 59.71 | | 50-52 | 0.00 | 60.57 | 66.95 | 57.98 | 60.37 | | 75-22 | 61.79 | 61.35 | 57.57 | 59.20 | 59.48 | | 75-27 | 61.41 | 61.46 | 57.77 | 59.33 | 59.59 | | 75-32 | 61.05 | 61.40 | 57.96 | 59.43 | 59.67 | | 75-37 | 61.17 | 61.62 | 58.29 | 59.80 | 60.01 | | 75-42 | 61.15 | 61.82 | 58.68 | 60.24 | 60.43 | | 75-47 | 60.59 | 62.10 | 59.04 | 60.14 | 60.83 | | 75-52 | 60.10 | 62.39 | 58.98 | 60.75 | 61.60 | | 100-22 | 74.82 | 75.20 | 71.11 | 73.37 | 73.96 | | 100-27 | 74.94 | 75.52 | 71.38 | 73.44 | 74.10 | | 100-32 | 75.01 | 75.70 | 71.49 | 73.49 | 74.22 | | 100-37 | 74.78 | 75.70 | 71.47 | 73.40 | 74.23 | | 100-42 | 74.74 | 75.66 | 71.40 | 73.21 | 74.40 | | 100-47 | 74.39 | 75.52 | 71.14 | 72.92 | 74.37 | | 100-52 | 73.65 | 75.32 | 71.40 | 73.48 | 74.99 | | | |

# Appendix B: Anchor Runtime Measurement

Runtime as a proxy for complexity in JVET / VVC standardization to be reported for anchors for Machine vision task and Human vision task. The proposed runtime methods for machine and human visions are:

* Machine vision tasks
  + EncTm: Time needed to convert RGB input to bitstream
  + DecTm: Time needed to convert bitstream to inference output
* Human vision
  + Encv: time needed to convert RGB input to bitstream
  + DecTv: Time needed to convert bitstream to YUV

The runtime methods for machine and human visions are shown in Figure 5 and Figure 6, respectively.

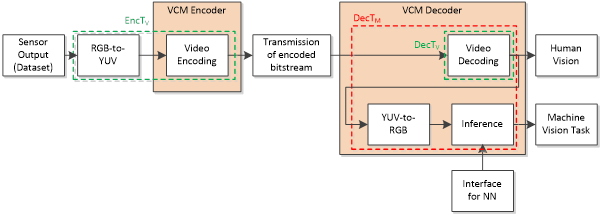


Figure . *Runtime measurements to be made for anchors*

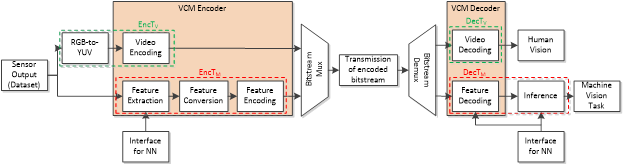


Figure . *Illustration of runtime measurements to be made for the proposed technology*