The Last Byte: Learning Just Enough for Machine-Oriented Image Compression

Wuyuan Xie[1], Zhenming Li[1], Ye Liu[2], Jian Jin[3], Yun Song[4], Miaohui Wang[5]*

[1]College of Computer Science & Software Engineering, Shenzhen University
[2]School of Automation, Nanjing University of Posts and Telecommunications
[3]Alibaba-NTU Joint Research Institute, Nanyang Technological University
[4]School of Computer Science and Technology, Changsha University of Science and Technology
[5]Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen University
wuyuan.xie@gmail.com, wang.miaohui@gmail.com

Abstract

Just recognizable distortion (JRD) has been introduced for image compression for machines, aiming to quantify the maximum coding distortion that can be tolerated by a specific perception model, thereby defining the upper bound of machine vision redundancy (MVR). However, existing JRD-based redundancy estimation methods face three key challenges: *limited dataset annotation accuracy*, *low prediction efficiency*, and *insufficient perception accuracy*, all of which hinder their practical deployment. To address these limitations, we propose a new *MVRNet*, a frame-wise efficient JRD prediction method that generates the optimal encoding quantization map in a single inference pass. Furthermore, we refine the annotation standard for JRD datasets based on experimental insights, enhancing the precision of recognizable redundancy measurement. Compared to state-of-the-art methods, *MVRNet* achieves a superior balance between bitrate reduction and perception accuracy in JRD-guided compression, while offering up to a 40,000× speed improvement, demonstrating its practicality and efficiency for real-world applications.

1 Introduction

With the rapid advancement of deep learning, machine vision models have achieved significant breakthroughs in many tasks such as *image classification*, *object detection*, and *semantic segmentation*. Object detection, a core computer vision task, has been successfully applied to multiple scenarios [Vahab et al. 2019] such as intelligent driving, intelligent security, intelligent access control, and intelligent factories. However, widely-deployed object detection systems generate massive visual data (*e.g.*, images, videos, and point clouds), which brings huge challenges to data storage and transmission [Sheng et al. 2024; Lin et al. 2023].

Modern image and video compression techniques (*e.g.*, *H.264/AVC* [Wang, Ngan, and Xu 2014], *H.265 /HEVC* [Wang, Ngan, and Li 2016], and *H.266/VVC* [Wang et al. 2021]) are optimized for the human visual system (HVS) but fail to consider the distinct perceptual features of machine vision models [Choi and Bajić 2022; Liu et al. 2023]. While advanced methods such as *pre-smoothing* [Wu et al. 2017], *prediction residual adjustment* [Wang et al. 2016], and *quality factor mapping* [Wang et al. 2022] improve compression efficiency for human perception, they are not directly transferable to machine perception due to fundamental differences in visual redundancy estimation. Machine vision models exhibit (1) fine-grained sensitivity, where small pixel variations can significantly impact prediction accuracy, and (2) task-driven perception, focusing on object-relevant regions rather than global visual quality.

To optimize compression for machine vision, just recognizable distortion (JRD) quantifies the maximum distortion tolerable without degrading machine perception accuracy [Zhang et al. 2021, 2024]. Existing JRD estimation methods typically adopt full-reference, reduced-reference, or no-reference paradigms, each balancing accuracy and practicality. However, existing JRDs on machine vision redundancy (MVR) estimation faces three major challenges:

- **Limited Dataset Annotation Accuracy**. Existing annotations often employ overly coarse object-background segmentation, leading to the omission of significant object regions and inaccuracies in MVR measurement. Furthermore, insufficient exploration of encoding parameters hinders precise visual redundancy estimation.
- **Low Prediction Efficiency**. The estimation process encounters dual bottlenecks: (1) multiple inference steps per object significantly increase computational overhead, and (2) object-wise annotation methods require numerous predictions per frame to determine its overall redundancy, substantially reducing computation efficiency.
- **Insufficient Perception Accuracy**. While reduced-reference and full-reference paradigms are commonly used, no-reference JRD suffers from high perception errors due to the absence of machine vision-specific feature information in compressed visual data (*e.g.*, image, video, or point cloud).

To address the challenges mentioned above, we make the following contributions: (1) we refine JRD annotation standards and construct a more comprehensive dataset for practical JRD estimation; (2) we propose a machine vision-driven JRD prediction method that generates a JRD-based quantization map using a single inference on the original frame, eliminating the need for reference information and signif-

*Corresponding author: Miaohui Wang

icantly improving efficiency; and (3) we incorporate machine vision characteristics, leveraging the original frame and confidence maps from the detection model to enhance JRD prediction accuracy. Experimental results show that, when applied to machine-oriented compression, our approach achieves the highest perception accuracy and up to a 40,000× speed improvement at equivalent bitrates compared to state-of-the-art methods, demonstrating superior efficiency and effectiveness.

2 Related Work

We highlights the distinction between human-oriented just noticeable difference (JND) and machine-oriented JRD.

2.1 Just Noticeable Difference (JND)

JND quantifies the maximum stimulus variation that the HVS cannot perceive, serving as a key indicator of human visual redundancy (HVR). Traditional JND estimation methods [Lin and Ghinea 2021; Pan et al. 2024] typically adopted nonlinear additive masking models [Yang et al. 2005] to combine masking effects such as luminance adaptation [Liu et al. 2010], contrast masking [Chen and Wu 2019], pattern masking [Wu et al. 2017], and structural sensitivity [Wang et al. 2016]. These JND models aim to simulate the perceptual limits of the human eye by leveraging the masking characteristics of the HVS. However, these hand-crafted integration rules often struggle to capture the complex coupling between multiple masking mechanisms and structural redundancy.

Currently, both knowledge-driven [Shen et al. 2020; Jiang et al. 2022; Wang et al. 2022] and data-driven [Xie et al. 2023; Jiang et al. 2024] have emerged to better model the intricate correlation between human perception and the JND threshold under diverse conditions. These advances have significantly enhanced our understanding of the HVR modeling and have laid a stronger foundation for practical applications such as image coding, watermarking, and quality assessment. However, there is a huge difference between human perception and machine perception. These HVS-oriented JND models cannot be directly applied to the MVR prediction to improve the accuracy of vision tasks.

2.2 Just Recognizable Distortion (JRD)

Given that JND reflects the perceptual limit of HVR in human vision, it is natural to ask whether machine vision systems exhibit analogous perceptual thresholds. Jin *et al*. [Jin et al. 2021] initiated this inquiry by qualitatively and quantitatively exploring JND-like behavior in machine vision. Building on this, Zhang *et al*. [Zhang et al. 2021] introduced the concept of JRD and constructed one of the earliest JRD datasets for image classification and object detection using public datasets such as COCO. However, their approach employed a reduced-reference method based on a single-level compression, and the annotation was restricted to *the person class*, limiting its applicability.

To address this limitation, Zhang *et al*. [Zhang et al. 2024] expanded the dataset to include multiple object categories and compression levels, adopting a full-reference paradigm for finer-grained prediction. However, the strict confidence threshold (*e.g*., 0.9) used for object filtering excluded approximately 34.8% of valid objects, compromising generalization in real-world scenarios. To overcome this, Liu *et al*. [Liu et al. 2024] proposed a no-reference framework using generative adversarial networks to synthesize pseudo-JRD images, with JRD predicted via residual learning. Additionally, Zhang *et al*. [Zhang et al. 2025] introduced the concept of machine satisfaction rate, inspired by user satisfaction in human perception, to estimate the MVR information for machine vision models.

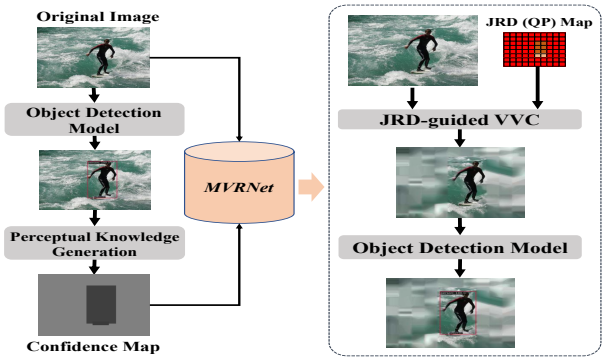

Figure 1: Illustrations of machine vision redundancy (MVR) annotation and JRD-based compression.

Despite these promising directions, existing JRD prediction methods still face several critical challenges [Yang et al. 2024], including *coarse annotation*, *limited scalability*, and *inefficiencies* in prediction. These shortcomings constrain their applicability in practical deployments and require more effective solutions to model, predict, and reduce MVR information in machine vision.

3 Rethinking JRD-based Compression

In coding for machines, JRD refers to the maximum compression distortion that can be tolerated without affecting vision task accuracy [Duan et al. 2020]. This section presents our no-reference JRD framework, as shown in Figure 1.

3.1 Problem Formulation

Existing methods for JRD prediction primarily focus on estimating the maximum MVR of object regions. These approaches typically employ full-reference or reduced-reference strategies, using several distortion levels of object patches to infer the JRD levels. Specifically, for a given object patch $R^i_{ori}(i \in [1, n])$ of the i-th object in the original image, with a set of encoded versions $\mathbb{R}^i_{enc} = \{R^i_{enc,1}, \ldots, R^i_{enc,j}, \ldots\}$, the goal is to use a prediction model $\mathcal{F}_{jrd}(\cdot)$ to estimate the fine-grained JRD level $\hat{L}^i_{jrd}$, which quantifies the maximum tolerable compression distortion that preserves machine recognition accuracy:

$$\hat{L}^i_{jrd} = \mathcal{F}_{jrd}(R^i_{ori}, \mathbb{R}^i_{enc}). \quad (1)$$

The predicted fine-grained JRD levels $\{\hat{L}^1_{jrd}, \hat{L}^2_{jrd}, \ldots, \hat{L}^n_{jrd}\}$, along with the corresponding object detection boxes,

Figure 2: Relationship between the confidence score and the object region JRD level (*i.e.*, coding unit-level QPs).

are then used to construct a JRD-based QP map for the entire image. Although existing methods [Zhang et al. 2024] can achieve promising accuracy, they have notable drawbacks:

- The reliance on reference-based paradigms requires multiple compression loops of object regions, increasing computational overhead.
- Per-image JRD prediction requires separate processing for each detected object, resulting in high time and memory complexity.

To address these challenges, we propose a no-reference MVR prediction framework that directly predicts a JRD-based QP map from a single input image. Formally, for a given image I_{ori}, the prediction model $\mathcal{F}_{mvrp}(\cdot)$ estimates its JRD-based QP map $\mathbf{M}_{mvrp}$ without requiring any other additional inputs:

$$\mathbf{M}_{mvrp} = \mathcal{F}_{mvrp}(I_{ori}). \tag{2}$$

However, directly predicting $\mathbf{M}_{mvrp}$, particularly in object regions (*i.e.*, predicting the JRD of objects), remains highly challenging. The primary difficulty lies in the lack of task-relevant cues in the original image, which hinders the model's ability to generalize effectively.

To address this issue, we propose incorporating perceptual characteristic knowledge derived from pre-trained machine vision models. This additional information, denoted as $I_{knowledge}$, captures features relevant to the JRD prediction task and serves as an auxiliary information to improve the accuracy of $\mathcal{F}_{mvrp}$. The model is thus designed to jointly utilize the original image I_{ori} and the task-specific knowledge $I_{knowledge}$ to predict $\mathbf{M}_{mvrp}$, formulated as follows:

$$\mathbf{M}_{mvrp} = \mathcal{F}_{mvrp}(I_{ori}, I_{knowledge}). \tag{3}$$

It enhances the model capacity to capture both low-level visual cues and high-level perceptual knowledge, thereby improving the reliability of no-reference JRD prediction.

3.2 Perceptual Knowledge Analysis

To improve the accuracy of JRD prediction in object regions, we analyze the recognition behavior of machine vision models, extract perceptual characteristics related to object-level JRD, and generate machine perceptual knowledge to embed into $\mathcal{F}_{mvrp}$.

PCC: Object Size / Image Status	Small	Medium	Large	Overall
Normal (7539 images)	0.756	0.749	0.824	0.718
Abnormal (2012 images)	0.484	0.506	0.491	0.513

Table 1: Pearson correlation coefficient (PCC) between confidence and JRD-based QP map.

Analysis of Machine Perception Characteristics. For machine vision models trained on specific tasks such as *object detection*, perception is inherently task-driven. We analyze model outputs to uncover latent correlations between these outputs and the JRD levels of detected objects. This enables the extraction of structured perceptual knowledge reflecting recognition certainty, which can enhance JRD prediction for object regions.

In the context of *object detection*, the core objective is to localize and identify objects. Similar to human perception, where recognition certainty varies with familiarity, viewpoint, and context, machine detection models also show varying levels of confidence across object instances. Humans can recognize familiar objects under degraded conditions due to prior knowledge, while unfamiliar objects require higher visual fidelity. Based on this analogy, we investigate whether machine recognition certainty correlates with detection robustness under compression distortions.

To characterize machine perception, we use the primary outputs of deep neural network (DNN)-based detectors: 1) *bounding box coordinates*, 2) *class labels*, and 3) *confidence scores*. While coordinates and labels reflect spatial and semantic accuracy, the confidence score quantifies model uncertainty and indicates the strength of detection belief.

More importantly, we observe that confidence scores encode meaningful perceptual knowledge relevant to $\mathbf{M}_{mvrp}$. Through empirical analysis, we identify a key pattern: *objects assigned higher confidence scores in the original image tend to maintain correct recognition even under higher degrees of distortion, indicating a greater tolerance to compression*. That is, objects with higher initial confidence typically correspond to higher JRD values. This relationship is illustrated in Figure 2.

To validate this observation, we have examined the correlation between the confidence scores and the JRD levels across the whole dataset. Across all detected objects, the Pearson correlation coefficient (PCC) is 0.512. At the coding unit (CU) level, after filtering abnormal cases (*e.g.*, images with a single object or uniform JRD), the average PCC rises to 0.718 (Table 1), showing a strong positive correlation. By object size, large objects exhibit the highest PCC (0.824), while medium and small objects are around 0.75. Since large objects dominate the dataset and span more CUs, this correlation is particularly useful for improving JRD prediction.

In summary, confidence scores serve as a robust proxy for the perceptual certainty of detection model. Their strong correlation with JRD levels across object sizes highlights their utility as perceptual knowledge. Incorporating

this confidence-based knowledge into the MVR prediction framework can therefore substantially enhance the accuracy of no-reference JRD estimation.

Perceptual Knowledge Generation. To leverage the aforementioned perceptual knowledge for JRD prediction, we design a perceptual knowledge generation module $\mathcal{F}_{knowledge}$ that constructs confidence-based knowledge representations $I_{knowledge}$ from the detection outputs of a machine vision model, which is defined as follows:

$$I_{knowledge} = I_{conf} = \mathcal{F}_{knowledge}\left(\mathcal{F}_{det}(I_{ori})\right), \quad (4)$$

where I_{conf} denotes the confidence map. $\mathcal{F}_{det}$ denotes an object detection model, and the Faster R-CNN [Ren et al. 2016] is used for simplicity in the annotation stage.

The process for generating the confidence map proceeds as follows: We first input the original image into an object detection model to obtain detection results. Based on the remaining bounding boxes and their associated confidence scores, we construct a confidence map that reflects the model certainty across the spatial extent of the image. However, to facilitate joint input with the original image into a neural network, the confidence map is defined as a two-dimensional array with the same spatial resolution (*i.e.*, height and width) as the original image, thus avoiding coordinate scaling. Each pixel in the confidence map initially takes a value of 1, indicating maximum certainty.

For each valid detection box, we generate a confidence sub-map based on the box position and confidence score. These sub-maps are sequentially merged into the global confidence map using a pixel-wise minimum operation. This ensures that areas covered by multiple overlapping boxes reflect the lowest (most uncertain) confidence value among them. The resulting confidence map encodes the spatial distribution of perceptual certainty and serves as an informative auxiliary input for improving JRD prediction performance.

3.3 Establishment of The Proposed *MVRSet*

To better train our *MVRNet*, we further construct a promising MVR dataset, namely *MVRSet*. Specifically, we have randomly sampled 10,000 images from the COCO2017 dataset [Caesar, Uijlings, and Ferrari 2018] as source images and employed Faster R-CNN [Ren et al. 2015] with a ResNet-101 backbone, pre-trained on COCO2017, for object detection. The model weights are not fine-tuned for compression artifacts. Each image is compressed using the VVC standard (VTM-22.2 reference software) at 64 quantization parameter (QP) levels ranging from 0 to 63, yielding a total of 640,000 distorted images. All compressed images are re-evaluated using the same detection model to obtain detection results across varying compression levels.

To facilitate machine vision redundancy prediction, annotations are created to quantify visual redundancy in both background and object regions. As illustrated in Figure 3, the annotation process involves three key steps:

- **Background Redundancy Annotation**: Non-essential regions are identified by filtering the original detection outputs to isolate background areas unrelated to the task. Compression is applied, and detection performance is monitored to ensure degradation remains within acceptable bounds.
- **Foreground Object JRD Annotation**: Compressed variants of each object-containing region are analyzed to determine the JRD point, *i.e.*, the maximum tolerable distortion before the detection output becomes unacceptably degraded.
- **JRD-based QP Map Generation**: The per-object redundancy maps are aggregated to produce a unified QP map $\mathbf{M}_{mvrp}$ for each image. This map serves as a ground truth for training models in end-to-end machine vision redundancy prediction.

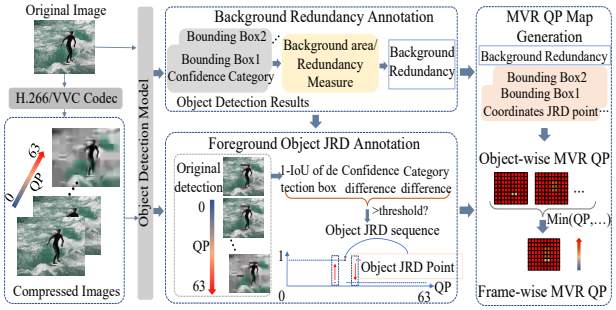

Figure 3: Pipeline the enhanced *MVRSet* dataset. For each original image, 64 distorted versions are generated, followed by a three-step annotation process using an object detection model: (1) background redundancy annotation, (2) object-region JRD labeling, and (3) generation of the corresponding visual redundancy QP map.

The resulting JRD-based QP map $\mathbf{M}_{mvrp}$, incorporating background processing, yields a comprehensive visual redundancy QP for the entire image. A typical annotation example is provided in Figure 2.

Table 2 summarizes both the count and average JRD of objects by size. As seen, medium and large objects are more prevalent and exhibit higher average JRD values, with large objects reaching QP=31 and small ones QP=22. While the overall object-level average is QP=28, the CTU-level average is QP=0, reflecting the dominance of large objects in both frequency and spatial coverage.

4 Proposed JRD-driven *MVRNet* Model

In this section, we present a JRD-based QP prediction framework that integrates machine perception knowledge, with a focus on the proposed MVR network (*MVRNet*) designed specifically for the no-reference prediction. *MVRNet* fuses the confidence map with the original image to predict the MVR-induced QP map.

Inspired by the efficient bilateral network architecture commonly used in semantic segmentation tasks [29], we design a dual-path neural network that processes the original image I_{ori} and the corresponding confidence map I_{conf} in parallel, as illustrated in Figure 4. The overall architecture consists of two main components: the efficient feature extraction module (EFEM) $\mathcal{F}_{extract}$, and the efficient feature fusion module (EFFM) $\mathcal{F}_{fuse}$. The former independently

Properties \ Object size	Small	Medium	Large	Overall
Pixel area size	<32×32	32×32∼96×96	>96×96	-
Number (objects)	6993	15759	18197	40949
Average JRD (objects)	22	27	31	28
Number (coding units)	12674	59224	402971	474869
Average JRD (coding units)	22	27	31	30

Table 2: Statistics of object size and coding unit (CU)-related values in our *MVRSet*.

extracts multi-scale features from both inputs, while the latter integrates the extracted representations to produce a compact and informative feature map, as formulated by

$$\mathbf{M}_{mvrp} = \mathcal{F}_{mvrp}(I_{ori}, I_{conf}) = \mathcal{F}_{fuse}(\mathcal{F}_{extract}(I_{conf}, I_{ori})). \quad (5)$$

4.1 Efficient Feature Extraction Module

The EFEM consists of two networks: the task path $\mathcal{F}_{task}$ and the pixel path $\mathcal{F}_{pixel}$, which are used to extract the multi-scale task features $\mathbf{F}_{task} = \{\mathbf{F}^1_{task}, \mathbf{F}^2_{task}\}$ in the confidence map and the multi-scale pixel features $\mathbf{F}_{pixel} = \{\mathbf{F}^1_{pixel}, \mathbf{F}^2_{pixel}\}$ of the original image, respectively.

$$\mathbf{F}_{task}, \mathbf{F}_{pixel} = \mathcal{F}_{task}(I_{conf}), \mathcal{F}_{pixel}(I_{ori}). \quad (6)$$

The task path $\mathcal{F}_{task}$ is composed of a stacked 7×7 convolutional layer $\mathcal{F}_{Conv7}$, multiple Res18 sub-blocks $\mathcal{F}_{Res18}$, and two channel attention modulation modules (CAMM) $\mathcal{F}_{CAMM}$. On the other hand, the pixel path $\mathcal{F}_{pixel}$ is composed of a stacked 3×3 convolution layer $\mathcal{F}_{Conv3}$, multiple MBConv sub-blocks $\mathcal{F}_{MBConv}$ and two CAMM. Among them, the convolution layer is used to extract shallow features $\mathbf{F}_{conf}$ and $\mathbf{F}_{ori}$, respectively:

$$\mathbf{F}_{conf}, \mathbf{F}_{ori} = \mathcal{F}_{Conv7}(I_{conf}), \mathcal{F}_{Conv3}(I_{ori}). \quad (7)$$

Consequently, multi-scale task and pixel features $\mathbf{F}_{task}$ and $\mathbf{F}_{pixel}$ are extracted as follows:

$$\begin{cases} \mathbf{F}^1_{task}, \mathbf{F}^2_{task} = \mathcal{F}_{CAMM1}(\mathcal{F}^6_{Res18}(\mathbf{F}_{conf})), \mathcal{F}_{CAMM2}(\mathcal{F}^2_{Res18}(\mathbf{F}^1_{task})) \\ \mathbf{F}^1_{pixel}, \mathbf{F}^2_{pixel} = \mathcal{F}_{CAMM3}(\mathcal{F}^5_{MBConv}(\mathbf{F}_{ori})), \mathcal{F}_{CAMM4}(\mathcal{F}^2_{MBConv}(\mathbf{F}^1_{pixel})) \end{cases}, \quad (8)$$

where $\mathcal{F}^N_{Res18}$ means stacking N network blocks. The network structure and function of each component are given as follows:

- The Res18 module is based on the Resnet18 [He et al. 2016] and is built by stacking multiple 3×3 convolution layers $\mathcal{F}_{Conv3}$ and batch normalization (BN) layers $\mathcal{F}_{BN}$. It is mainly used to extract multi-scale task features from confidence map inputs. The processing of the input feature $\mathbf{F}_{in}$ can be expressed as:

$$\mathcal{F}_{Res18}(\mathbf{F}_{in}) = \mathcal{F}_{BN}(\mathcal{F}_{Conv3}(\mathcal{F}_{BN}(\mathcal{F}_{Conv3}(\mathbf{F}_{in})))) + \mathbf{F}_{in}. \quad (9)$$

- The MBConv module is inspired by [Tan and Le 2019], and consists of a 3×3 convolution layer, a deep-wise convolution layer $\mathcal{F}_{DwConv}$, a squeeze block (SE) [Hu, Shen, and Sun 2018] $\mathcal{F}_{SE}$, and a random dropout layer $\mathcal{F}_{Dropout}$. Residual connections are also introduced. The processing of the input feature $\mathbf{F}_{in}$ can be expressed as:

$$\mathcal{F}_{MBConv}(\mathbf{F}_{in}) = \mathcal{F}_{Dropout}(\mathcal{F}_{Conv3}(\mathcal{F}_{SE}(\mathcal{F}_{DwConv}(\mathcal{F}_{Conv3}(\mathbf{F}_{in}))))) + \mathbf{F}_{in}. \quad (10)$$

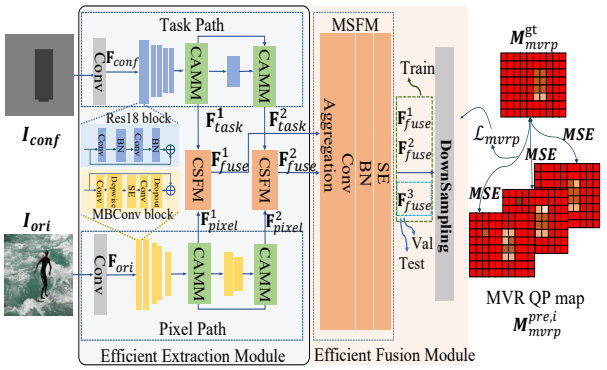

Figure 4: Pipeline of frame-wise efficient JRD prediction model (*MVRNet*).

- The CAMM module is composed of the global pooling layer $\mathcal{F}_{GAP}$, the 1×1 convolution layer $\mathcal{F}_{Conv1}$, the BN layer, and the Sigmoid layer $\mathcal{F}_{Sigmoid}$. It uses channel attention to further refine the input feature $\mathbf{F}_{in}$ as:

$$\mathcal{F}_{CAMM}(\mathbf{F}_{in}) = \mathcal{F}_{Sigmoid}(\mathcal{F}_{BN}(\mathcal{F}_{Conv1}(\mathcal{F}_{GAP}(\mathbf{F}_{in})))) \odot \mathbf{F}_{in}. \quad (11)$$

where $\odot$ denotes the element-wise multiplication.

4.2 Efficient Feature Fusion Module

The EFFM mainly includes the context-spatial fusion module (CSFM) $\mathcal{F}_{CSFM}$, the multi-scale fusion module (MSFM) $\mathcal{F}_{MSFM}$, and the down-sampling output module $\mathcal{F}_{DownSample}$, as defined by:

$$\mathcal{F}_{fuse}(\mathbf{F}_{task}, \mathbf{F}_{pixel}) = \mathcal{F}_{DownSample}(\mathcal{F}_{MSFM}(\mathcal{F}_{CSFM}(\mathbf{F}_{task}, \mathbf{F}_{pixel}))), \quad (12)$$

Further, the CSFM module consists of one concatenation layer $\mathcal{F}_{Concat}$, one 3×3 convolution layer, one BN layer, and one SE block. It is mainly used to fuse the contextual tasks and spatial pixel features of the two branches at different scales to obtain the fusion features $\mathbf{F}^1_{fuse}$ and $\mathbf{F}^2_{fuse}$, respectively.

$$\begin{cases} \mathbf{F}^1_{fuse} = \mathcal{F}_{CSFM1}(\mathbf{F}^1_{task}, \mathbf{F}^1_{pixel}) = \mathcal{F}_{SE}(\mathcal{F}_{BN}(\mathcal{F}_{Conv3}(\mathcal{F}_{Concat}(\mathbf{F}^1_{task}, \mathbf{F}^1_{pixel})))) \\ \mathbf{F}^2_{fuse} = \mathcal{F}_{CSFM2}(\mathbf{F}^2_{task}, \mathbf{F}^2_{pixel}) = \mathcal{F}_{SE}(\mathcal{F}_{Conv3}(\mathcal{F}_{Concat}(\mathbf{F}^2_{task}, \mathbf{F}^2_{pixel}))) \end{cases}, \quad (13)$$

where $\mathcal{F}_{CSFM1}$ and $\mathcal{F}_{CSFM2}$ are the head and tail CSFMs.

Meanwhile, the MSFM module consists of one aggregation layer, one 3×3 convolution layer, and one BN layer. The aggregation layer mainly includes one average pooling layer $\mathcal{F}_{AP2}$ with the kernel and step size of 2, and one concatenation layer. MSFM is used to fuse the multi-scale features output of the CSFM to adapt to objects with different sizes, and it generates the fused feature $\mathbf{F}^3_{fuse}$:

$$\mathbf{F}^3_{fuse} = \mathcal{F}_{BN}(\mathcal{F}_{Conv3}(\mathcal{F}_{Concat}(\mathcal{F}_{MSFM}(\mathcal{F}_{AP2}(\mathbf{F}^1_{fuse}), \mathbf{F}^2_{fuse})))). \quad (14)$$

Following the feature extraction and fusion, the combined features are progressively downsampled to a fixed spatial resolution using a downsampling module $\mathcal{F}_{DownSample}$ composed of a 3×3 convolution layer followed by an average pooling layer. This final output represents the predicted $\mathbf{M}_{mvrp}$. During training, we introduce three downsampling

Methods \ Errors	E_s	E_m	E_l	E_{object}	E_{bg}	E_{total}
JRDVCM	**7.529**	8.429	9.486	9.385	-	4.655
OWJRD	13.940	13.122	12.190	12.298	-	6.035
Proposed	8.250	**7.585**	**7.532**	**7.735**	0.679	**4.140**

Table 3: **Prediction error** across object sizes, object regions, background regions, and overall performance.

output branches to ensure effective supervision of the relatively shallow network. These branches independently predict JRD-based QP maps from the intermediate features of the two CSFMs and the final MSFM. Losses from each prediction branch are computed and backpropagated to jointly optimize the model parameters across all levels of the network, which is defined as:

$$\begin{cases} \mathbf{M}_{mvrp}^{pre,1} = \mathcal{F}_{DownSample}(\mathbf{F}_{fuse}^{1}) = \mathcal{F}_{AP8}(\mathcal{F}_{BN}(\mathcal{F}_{Conv3}(\mathbf{F}_{fuse}^{1}))) \\ \mathbf{M}_{mvrp}^{pre,2} = \mathcal{F}_{DownSample}(\mathbf{F}_{fuse}^{2}) = \mathcal{F}_{AP4}(\mathcal{F}_{BN}(\mathcal{F}_{Conv3}(\mathbf{F}_{fuse}^{2}))) \\ \mathbf{M}_{mvrp}^{pre,3} = \mathcal{F}_{DownSample}(\mathbf{F}_{fuse}^{3}) = \mathcal{F}_{AP4}(\mathcal{F}_{BN}(\mathcal{F}_{Conv3}(\mathbf{F}_{fuse}^{3}))) \end{cases}, \quad (15)$$

where $\mathcal{F}_{AP8}$ and $\mathcal{F}_{AP4}$ refers to the average pooling layer with the kernel size and stride of 8 and 4, respectively.

During the validation and testing phases, the training model only retains the last downsampled output network, using the output features of MSFM to predict the JRD-based QP map $\mathbf{M}_{mvrp}^{pre,3}$.

5 Experimental Results

5.1 Experimental Protocol

Implementation Details. We implement *MVRNet* using *PyTorch* [Paszke et al. 2017], initializing all model parameters with Kaiming initialization [He et al. 2015]. The proposed *MVRSet* is partitioned into training, validation, and test subsets in an 8:1:1 ratio. Since the original image sizes vary and both height and width are less than 640 pixels, we apply zero-padding to all input images and confidence maps to standardize them to 640×640 resolution. Prior to training and evaluation, perceptual knowledge generation and data preprocessing are performed. The model is trained using the Adam optimizer with default parameters on an *NVIDIA GeForce RTX 3090* GPU. We set the batch size to 16, the initial learning rate to *3e-4*, and apply a linear decay schedule over 200 epochs. The *Faster R-CNN* object detection model is deployed using the *MMDetection* framework [Chen et al. 2019] to evaluate the impact of redundancy prediction on downstream machine tasks.

During training, *MVRNet* generates three predicted JRD-based QP maps, corresponding to the outputs from two CAMM modules and one CSFM module. To improve convergence of the shallow layers, we define the training loss $\mathcal{L}_{mvrp}$ as the average of the mean squared error (MSE) between each of the three predicted QP maps and the ground truth. $\mathcal{L}_{mvrp}$ is defined by

$$\mathcal{L}_{mvrp} = \frac{1}{3}\sum_{i=1}^{3} MSE\left(\mathbf{M}_{mvrp}^{pre,i}, \mathbf{M}_{mvrp}^{gt}\right), \quad (16)$$

where $\mathbf{M}_{mvrp}^{pre,i}$ denotes the i-th predicted JRD-based QP map, $\mathbf{M}_{mvrp}^{gt}$ denotes the ground truth QP map, and $MSE(\cdot)$ denotes the MSE loss function.

Methods \ Steps	Pre-detect.	Confidence map	Image coding	Cropping	QP map predict.	Overall
JRDVCM	**0.062s**	-	34.93s	0.027s	1.20s	36.22s
OWJRD	**0.062s**	-	7110.28s	0.027s	5.83s	7116.19s
Proposed	**0.062s**	0.025s	-	-	**0.082s**	**0.17s**

Table 4: Computational complexity across five processing stages for different prediction methods.

Comparison Methods. To evaluate the performance of the proposed method, we compare it against four SOTA approaches: *SDVCM* [Fischer et al. 2021], *JRDVCM* [Zhang et al. 2021], *MAVCM* [Lee et al. 2023], and *OWJRD* [Zhang et al. 2024].

Evaluation Metrics. The overall prediction error, E_{total}, is computed to quantify the discrepancy between predicted and reference QP maps:

$$E_{total} = \frac{\sum_{x=0}^{H-1}\sum_{y=0}^{W-1}|\mathbf{M}_{mvrp}^{pre,i}(x,y) - \mathbf{M}_{mvrp}^{gt}(x,y)|}{W \times H}, \quad (17)$$

where $\mathbf{M}_{mvrp}^{pre,i}(x,y)$ and $\mathbf{M}_{mvrp}^{gt}(x,y)$ denotes the predicted and labeled visual redundancy QP values at the position of *(x, y)*, respectively. W and H denotes the corresponding length and width, respectively. $|\cdot|$ represents the absolute operation. The regional prediction error E_{area} is calculated as follows:

$$E_{area} = \frac{\sum_{(x,y)}|\mathbf{M}_{mvrp}^{pre,i}(x,y) - \mathbf{M}_{mvrp}^{gt}(x,y)|}{N}, (x,y) \in area, \quad (18)$$

where *area* indicates the corresponding background or object area, and N denotes the number of pixels in the corresponding area. In addition, we also calculate the prediction errors of large objects E_l, medium objects E_m, and small objects E_s, respectively.

Time complexity significantly affects the practical deployment of the model. We evaluate this by measuring the total processing time, which includes five main stages: 1) *object detection*, 2) *confidence map generation*, 3) *image compression*, 4) *cropping*, and 5) *QP map prediction*. This end-to-end measurement provides a comprehensive assessment of prediction efficiency.

5.2 Overall Performance

Prediction Performance. Table 3 shows that *MVRNet* outperforms both baselines, particularly in medium, large, and object-specific regions. For the background error metric E_{bg}, both *JRDVCM* and *OWJRD* lack predictions for background areas, resulting in missing values. In contrast, *MVRNet* achieves an E_{bg} of 0.679, a small error that has negligible impact on subsequent compression performance.

Computational Complexity. Table 4 provides the average runing time for each step. All methods exhibit comparable performance during the pre-detection stage. However, in the compression and cropping stage, our approach eliminates the need for these steps, unlike *JRDVCM* and *OWJRD*, which incur additional overhead due to its full-reference design. In the inference stage, both *JRDVCM* and *OWJRD* re-

Original Image	Confidence Map	E_s	E_m	E_l	E_{total}
✓	×	21.068	15.468	12.316	13.785
×	✓	**7.728**	7.872	8.226	8.330
✓	✓	8.250	**7.585**	**7.532**	**7.735**

Table 5: Ablation results of the perceptual knowledge.

CAMM	CSFM	E_s	E_m	E_l	E_{total}
×	×	8.605	7.931	8.277	8.437
✓	×	**7.702**	7.897	7.663	7.844
×	✓	8.304	7.750	7.982	8.137
✓	✓	8.250	**7.585**	**7.532**	**7.735**

Table 6: Ablation results of different modules.

quire multiple iterations and post-processing steps to generate the final $\mathbf{M}_{mvrp}$, whereas our method produces the full-frame $\mathbf{M}_{mvrp}$ in a single forward pass. As a result, our approach achieves a total runtime of just 0.17 seconds, approximately 213× faster than *JRDVCM* and over **40,000× faster than *OWJRD***.

5.3 Ablation Study

Impact of Perceptual Knowledge. To assess the influence of perceptual knowledge, we conduct ablations by selectively removing either the confidence map (task path) or the original image (pixel path) as input. Each variant is retrained and evaluated independently. The results, summarized in Table 5, clearly show that using only the original image leads to the weakest performance. It is consistent with the analysis in Section 3.2, which confirms the difficulty of directly inferring frame-wise redundancy from raw pixels in a no-reference setting.

Impact of CAMM and CSFM Modules. To evaluate the contributions of the CAMM and the CSFM, we conduct controlled ablation experiments by replacing each module with standard convolutional layers, followed by retraining and evaluation. The results are reported in Table 6. As seen, when CAMM is retained and CSFM is removed, performance improves due to more effective refinement of low-level features. Conversely, when CSFM is used alone, feature fusion across branches is enhanced, also resulting in performance gains. The best results are achieved when both CAMM and CSFM are employed, demonstrating that these components provide complementary benefits.

5.4 More Validations

Cross Datasets. As shown in Table 7, the proposed method consistently achieves the highest mAP@0.50 across COCO, ImageNet-VID, and BDD100K, while maintaining the lowest BPP. On COCO, it reaches 0.8640 mAP at 0.5314 BPP, outperforming *JRDVCM* and *OWJRD*. On ImageNet-VID and BDD100K, it achieves 0.6657 and 0.4623 mAP at just 0.2663 and 0.1920 BPP, respectively. Compared to *H.266/VVC* and other learned methods, the proposed approach offers significantly better detection performance at

Method	*COCO*		*ImageNet-VID*		*BDD100K*	
	BPP	mAP@.50	BPP	mAP@.50	BPP	mAP@.50
H.266/VVC	5.8720	0.7290	4.6360	0.6840	1.9859	0.5360
SDVCM	0.6128	0.7316	0.3629	0.5872	0.3352	0.4011
MAVCM	0.9252	0.7580	0.4215	0.6121	0.4085	0.4117
JRDVCM	0.5510	0.8320	0.2868	0.6434	0.2057	0.4516
OWJRD	0.5341	0.8090	**0.2624**	0.6572	0.2215	0.4351
Proposed	**0.5314**	**0.8640**	0.2663	**0.6657**	**0.1920**	**0.4623**

Table 7: Cross-dataset validation on more methods.

Detector	*YOLO*	*DETR*	*Cascade R-CNN*	*Dynamic R-CNN*
	bpp/mAP@.50	bpp/mAP@.50	bpp/mAP@.50	bpp/mAP@.50
H.266/VVC	5.8720/**0.8460**	5.8720/**0.7290**	5.8720/**0.8570**	5.8720/**0.8040**
Proposed	**0.4532**/0.8401	**0.5435**/0.7100	**0.5023**/0.8470	**0.5123**/0.7832

Table 8: Detection validation on more backbones.

Instance Segmentation	*H.266/ VVC*	*JRDVCM*	*SDVCM*	*MAVCM*	*OWJRD*	*Proposed*
BPP	5.8643	0.4923	0.5275	0.6402	0.4553	**0.4481**
mAP@.50	0.6640	0.6234	0.5911	0.6163	0.6201	**0.6341**

Table 9: Compression validation on more tasks.

drastically lower bitrates, demonstrating strong generalization and coding efficiency.

Detection Backbones. Table 9 evaluates the proposed method across multiple detection backbones: YOLO [Xiao et al. 2025], DETR [Sun et al. 2024], Cascade R-CNN [Cai and Vasconcelos 2018], and Dynamic R-CNN [Zhang et al. 2020]. As seen, the proposed method achieves comparable mAP@0.50 to H.266/VVC across all detectors, while **reducing BPP by over 90%**. This highlights its strong robustness across diverse detection backbones.

Other Tasks. Table 9 presents instance segmentation performance on the *COCO* dataset. The proposed method achieves the highest mAP@0.50 (0.6341) at the lowest BPP (0.4481), outperforming all learned codecs and H.266/VVC. Notably, it improves both accuracy and compression, confirming its effectiveness for popular vision task under constrained BPP.

6 Conclusion

In this paper, we propose an efficient no-reference frame-wise machine redundancy prediction framework for visual coding. To guide machine redundancy estimation, we explore machine perception characteristics and distill them into task-relevant knowledge representations. Building on this, we introduce *MVRNet*, a dual-path network that jointly leverages the original image and perception-guided knowledge for accurate no-reference redundancy prediction. Furthermore, we present *MVRSet*, a large-scale dataset for machine vision redundancy, where we first investigate redundancy annotation strategies and generate frame-wise QP maps during annotation, arriving at a more accurate and practical annotation approach. Extensive experiments demonstrate the superiority of our method.

Acknowledgments

The authors would like to express their gratitude to Mr. Rong Zhang for his early preparatory work on this project. This work was supported in part by the National Natural Science Foundation of China under Grants 62472290 and 62372306, and in part by the Natural Science Foundation of Guangdong Province under Grants 2024A1515011972, and 2023A1515011197.

References

Caesar, H.; Uijlings, J.; and Ferrari, V. 2018. Coco-stuff: Thing and stuff classes in context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1209–1218.

Cai, Z.; and Vasconcelos, N. 2018. Cascade R-CNN: Delving into high quality object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 6154–6162.

Chen, K.; Wang, J.; Pang, J.; Cao, Y.; Xiong, Y.; Li, X.; Sun, S.; Feng, W.; Liu, Z.; Xu, J.; et al. 2019. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*.

Chen, Z.; and Wu, W. 2019. Asymmetric foveated just-noticeable-difference model for images with visual field inhomogeneities. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(11): 4064–4074.

Choi, H.; and Bajić, I. V. 2022. Scalable image coding for humans and machines. *IEEE Transactions on Image Processing*, 31: 2739–2754.

Duan, L.; Liu, J.; Yang, W.; Huang, T.; and Gao, W. 2020. Video coding for machines: A paradigm of collaborative compression and intelligent analytics. *IEEE Transactions on Image Processing*, 29: 8680–8695.

Fischer, K.; Fleckenstein, F.; Herglotz, C.; and Kaup, A. 2021. Saliency-driven versatile video coding for neural object detection. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 1505–1509.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1026–1034.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778.

Hu, J.; Shen, L.; and Sun, G. 2018. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 7132–7141.

Jiang, Q.; Liu, F.; Wang, Z.; Wang, S.; and Lin, W. 2024. Rethinking and Conceptualizing Just Noticeable Difference Estimation by Residual Learning. *IEEE Transactions on Circuits and Systems for Video Technology*.

Jiang, Q.; Liu, Z.; Wang, S.; Shao, F.; and Lin, W. 2022. Toward top-down just noticeable difference estimation of natural images. *IEEE Transactions on Image Processing*, 31: 3697–3712.

Jin, J.; Zhang, X.; Fu, X.; Zhang, H.; Lin, W.; Lou, J.; and Zhao, Y. 2021. Just noticeable difference for deep machine vision. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(6): 3452–3461.

Lee, Y.; Kim, S.; Yoon, K.; Lim, H.; Kwak, S.; and Choo, H.-G. 2023. Machine-Attention-based Video Coding for Machines. In *IEEE International Conference on Image Processing (ICIP)*, 2700–2704.

Lin, H.; Chen, B.; Zhang, Z.; Lin, J.; Wang, X.; and Zhao, T. 2023. DeepSVC: Deep scalable video coding for both machine and human vision. In *Proceedings of the 31st ACM International Conference on Multimedia (ACM MM)*, 9205–9214.

Lin, W.; and Ghinea, G. 2021. Progress and opportunities in modelling just-noticeable difference (JND) for multimedia. *IEEE Transactions on Multimedia*, 24: 3706–3721.

Liu, A.; Lin, W.; Paul, M.; Deng, C.; and Zhang, F. 2010. Just noticeable difference for images with decomposition model for separating edge and textured regions. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(11): 1648–1652.

Liu, L.; Hu, Z.; Chen, Z.; and Xu, D. 2023. Icmh-net: Neural image compression towards both machine vision and human vision. In *Proceedings of the 31st ACM International Conference on Multimedia (ACM MM)*, 8047–8056.

Liu, Y.; Yin, H.; Wang, H.; Wang, X.; and Yin, L. 2024. A Non-reference Just Recognized Distortion Prediction Framework for Object Detection Task. In *2024 Data Compression Conference (DCC)*, 570–570.

Pan, Z.; Zhang, G.; Peng, B.; Lei, J.; Xie, H.; Wang, F. L.; and Ling, N. 2024. JND-LIC: Learned image Compression via just noticeable difference for human visual perception. *IEEE Transactions on Broadcasting*, 71: 217–228.

Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; and Lerer, A. 2017. Automatic differentiation in pytorch. In *Conference on neural information processing systems (NIPS)*.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster R-CNN: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, 28.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2016. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6): 1137–1149.

Shen, X.; Ni, Z.; Yang, W.; Zhang, X.; Wang, S.; and Kwong, S. 2020. Just noticeable distortion profile inference: A patch-level structural visibility learning approach. *IEEE Transactions on Image Processing*, 30: 26–38.

Sheng, X.; Li, L.; Liu, D.; and Li, H. 2024. Vnvc: A versatile neural video coding framework for efficient human-machine vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(7): 4579–4596.

Sun, H.; Zhou, M.; Chen, W.; and Xie, W. 2024. TR-DETR: Task-reciprocal transformer for joint moment retrieval and highlight detection. In *AAAI Conference on Artificial Intelligence (AAAI)*, volume 38, 4998–5007.

Tan, M.; and Le, Q. 2019. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning (ICML)*, 6105–6114.

Vahab, A.; Naik, M. S.; Raikar, P. G.; and Prasad, S. 2019. Applications of object detection system. *International Research Journal of Engineering and Technology (IRJET)*, 6(4): 4186–4192.

Wang, M.; Ngan, K. N.; and Li, H. 2016. Low-delay rate control for consistent quality using distortion-based Lagrange multiplier. *IEEE Transactions on Image Processing*, 25(7): 2943–2955.

Wang, M.; Ngan, K. N.; and Xu, L. 2014. Efficient H.264/AVC video coding with adaptive transforms. *IEEE Transactions on Multimedia*, 16(4): 933–946.

Wang, M.; Xu, Z.; Liu, X.; Xiong, J.; and Xie, W. 2022. Perceptually quasi-lossless compression of screen content data via visibility modeling and deep forecasting. *IEEE Transactions on Industrial Informatics*, 18(10): 6865–6875.

Wang, M.; Zhang, J.; Huang, L.; and Xiong, J. 2021. Machine learning-based rate distortion modeling for VVC/H.266 intra-frame. In *IEEE International Conference on Multimedia and Expo (ICME)*, 1–6.

Wang, S.; Ma, L.; Fang, Y.; Lin, W.; Ma, S.; and Gao, W. 2016. Just noticeable difference estimation for screen content images. *IEEE Transactions on Image Processing*, 25(8): 3838–3851.

Wu, J.; Li, L.; Dong, W.; Shi, G.; Lin, W.; and Kuo, C.-C. J. 2017. Enhanced just noticeable difference model for images with pattern complexity. *IEEE Transactions on Image Processing*, 26(6): 2682–2693.

Xiao, Y.; Xu, T.; Xin, Y.; and Li, J. 2025. FBRT-YOLO: Faster and Better for Real-Time Aerial Image Detection. In *AAAI Conference on Artificial Intelligence (AAAI)*, volume 39, 8673–8681.

Xie, W.; Wang, S.; Zhang, R.; and Wang, M. 2023. Visual Redundancy Removal of Composite Images via Multimodal Learning. In *ACM International Conference on Multimedia (ACM MM)*, 6765–6773.

Yang, W.; Huang, H.; Hu, Y.; Duan, L.-Y.; and Liu, J. 2024. Video coding for machines: Compact visual representation compression for intelligent collaborative analytics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46: 5174–5191.

Yang, X.; Ling, W.; Lu, Z.; Ong, E. P.; and Yao, S. 2005. Just noticeable distortion model and its applications in video coding. *Elsevier Signal processing: Image communication*, 20(7): 662–680.

Zhang, H.; Chang, H.; Ma, B.; Wang, N.; and Chen, X. 2020. Dynamic R-CNN: Towards high quality object detection via dynamic training. In *Springer European Conference on Computer Vision (ECCV)*, 260–275.

Zhang, Q.; Wang, S.; Zhang, X.; Jia, C.; Wang, Z.; Ma, S.; and Gao, W. 2025. Perceptual Video Coding for Machines via Satisfied Machine Ratio Modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12): 7651–7668.

Zhang, Q.; Wang, S.; Zhang, X.; Ma, S.; and Gao, W. 2021. Just recognizable distortion for machine vision oriented image and video coding. *International Journal of Computer Vision*, 129(10): 2889–2906.

Zhang, Y.; Lin, H.; Sun, J.; Zhu, L.; and Kwong, S. 2024. Learning to predict object-wise just recognizable distortion for image and video compression. *IEEE Transactions on Multimedia*, 26: 5925–5938.