JPEG AI Compressed Domain Face Detection

Ayman Alkhateeb
Dep. of Information Engineering
CNIT – University of Brescia, Italy
Email: a.alkhateeb@studenti.unibs.it

Alessandro Gnutti
Dep. of Information Engineering
CNIT – University of Brescia, Italy
Email: alessandro.gnutti@unibs.it

Fabrizio Guerrini
Dep. of Information Engineering
CNIT – University of Brescia, Italy
Email: fabrizio.guerrini@unibs.it

Riccardo Leonardi
Dep. of Information Engineering
CNIT – University of Brescia, Italy
and Dep. of Electronics Engineering
University of Rome Tor Vergata, Italy
Email: riccardo.leonardi@unibs.it

João Ascenso
Instituto Superior Técnico,
Universidade de Lisboa -
Instituto de Telecomunicações
Lisbon, Portugal
Email: joao.ascenso@lx.it.pt

Fernando Pereira
Instituto Superior Técnico,
Universidade de Lisboa -
Instituto de Telecomunicações
Lisbon, Portugal
Email: fp@lx.it.pt

***Abstract*—Learning-based image coding has achieved competitive performance in terms of compression efficiency, while also gaining a key advantage in the ability to carry out computer vision tasks directly in the compressed domain. In fact, the latent representation which is generated using deep learning techniques may natively encapsulate all visual features needed for processing tasks, thereby eliminating the need to perform the expensive synthesis transform process at the decoder side. In this paper, it is proposed to perform face detection using the latent code present in the JPEG AI architecture. First, some experiments show how decoded images can be efficiently processed for face detection without retraining, albeit with some performance degradation. Then, for the first time a compressed domain RetinaFace-based detector applied to JPEG AI latent representations is competitively proposed. The performance achieved is comparable to the performance of the original RetinaFace applied to the reconstructed JPEG AI images, while reducing computational complexity since it bypasses the image decoding process. It is expected that this approach might be extended to other vision tasks since the JPEG AI representation format is not tailored specifically for any computer vision task.**

***Index Terms*—JPEG AI, learning-based image coding, latent representation, face detection, compressed domain processing.**

I. Introduction

The creation and consumption of multimedia content have been growing at an exponential rate due to the emergence of new trends and sophisticated technologies, including the Gigabit wireless backed Internet and digital gadgets. For example, it is estimated that roughly 660 billion photos were taken worldwide in 2013. However, by 2024 this number has increased to an unbelievable 1.94 trillion images, and people post approximately 14 billion images per day on different social networks [1].

This substantial amount of data highlights the importance of addressing issues related to multimedia storage and transmission. Consequently, there is a rapidly growing demand for efficient image compression technology, which plays a crucial role. Traditional image compression techniques, such as JPEG [2], HEVC/H.265 Intra [3], and other end-to-end methods were designed with the primary goal of preserving the fidelity of images while operating within predefined bitrate constraints.

Meanwhile, in recent years there has been substantial progress in the context of image coding using deep learning models, such as convolutional neural networks, recurrent neural networks, vision transformers, and generative adversarial networks. Methods based on these technologies have shown that they can reach a competitive performance in terms of visual fidelity compared with traditional transform-based codecs [4]–[6]. However, nowadays the end consumers of visual content, besides humans, include machines that may perform image processing or computer vision tasks. For example, Video Coding for Machines (VCM) has been proposed in [7], [8], aiming to bridge the gap between video/image compression and feature compression.

These insights motivated the development of JPEG AI, a new learning-based visual compression standard [9]. The key benefit of JPEG AI is the flexibility of its compressed domain representation that, besides image reconstruction, can be utilized to also perform visual processing and computer vision tasks directly in the compressed domain. Therefore, the features that are extracted while encoding the original image can be used instead of the lossy decoded image, performing these tasks with lower complexity. Importantly, the JPEG AI encoder builds the latent independently from any visual task, so there is no need to perform end-to-end training. This new way of performing learning-based image processing and/or computer vision in the compressed domain can have a significant impact in a wide range of applications, such as cloud storage, surveillance systems, autonomous vehicles, ...

A quintessential computer vision task is face detection, which serves as the foundation for more specific applications like face alignment and recognition. The primary objective of face detection is to accurately detect human faces from images and to return the spatial locations of faces via bounding

This work was partially supported by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, partnership on “Telecommunications of the Future” (PE00000001 - program “RESTART”), Projects SP7 FUN-Media and FRAME.

boxes [10]. The main objective is to discriminate facial regions from complex backgrounds, regardless of factors such as illumination variations, occlusions, pose changes, and facial expressions. Over the years, significant advancements in deep learning have revolutionized face detection techniques, enabling more robust and efficient solutions [11].

This paper investigates the impact of using JPEG AI decoded images at several bitrates on face detection performance, compared to using uncompressed images. These experiments are run on three state-of-the-art face detectors: RetinaFace [12], TinaFace [13], and YOLO5Face [14]. Then, the first ever compressed domain RetinaFace-based detector adapted to the JPEG AI latent representations is proposed and experimentally evaluated. It receives as input the JPEG AI latent instead of the image itself. To achieve this, the feature maps obtained after entropy decoding (and created by the JPEG AI analysis transform network) are leveraged to prune the face detector of the early stages responsible for low-level feature extraction. Moreover, a so-called bridge that consists of a group of additional learnable layers is introduced to align the JPEG AI latent dimensions with the face detector feature map size at the pruning point. The obtained face detection results demonstrate that performing the detection directly in the compressed domain significantly lowers computational complexity while achieving performance comparable to the original RetinaFace operating on decoded images.

In summary, the main contributions of this paper are:

- To study the face detection performance across a wide range of face detector models under different settings when JPEG AI decoded images are used.
- To propose the first-ever compressed domain face detector, consisting of a bridge that adapts the JPEG AI latent space to a pruned version of the RetinaFace detector.
- To demonstrate that the proposed model achieves performance comparable to an anchor that uses decoded images and the original RetinaFace detector, while significantly lowering the overall complexity.

The remainder of the paper is organized as follows. Sec. II provides a concise survey of prior works on compressed domain computer vision tasks. Then, Sec. III outlines the general framework of the proposed compressed domain RetinaFace-based face detector, while Sec. IV provides more details about its design and training process. Lastly, Sec. V discusses the test conditions and analyzes the experimental results, while Sec. VI concludes the paper.

II. Related Work

In the literature, many research works have proposed solutions where computer vision tasks are performed directly in the compressed domain. This section reviews a selection of previous works that specifically employ learning-based codecs.

Several methods resort to joint training of the image compression and task networks. In [15], image classification and segmentation are conducted inputting compact representations obtained from a convolutional auto-encoder directly into pruned inference networks, achieving accuracy similar to when decoded images are used instead, while reducing computational complexity. In this case, both joint and standalone training of the inference networks are considered. In [16], a learning-based compression scheme employing a variational auto-encoder is proposed, and a bridge network jointly trained with the encoder network was developed for visual object detection. A novel feature adaptation module is introduced in [17], which integrates a lightweight attention model to dynamically emphasize and enhance key features within channel-wise information inside a pruned inference model, in order to improve classification accuracy in the compressed domain.

Another option explored in the literature is to select only some of the latent representation channels to perform a computer vision task. The work in [18] suggests a method to enhance machine vision tasks in the compressed domain, using face alignment as an example. It proposes to selectively choose several channels from compressed representations, to perform up-sampling with a simple bridge, and finally to prune the task network. Similarly, [19] also proposes using a specific subset of compressed features. In particular, this work shows that learning from the compressed domain achieves comparable image classification accuracy compared to the uncompressed domain. In [20], the authors instead propose a method leveraging the compressed domain information to enhance segmentation tasks by employing both dynamic and static channel selection, and knowledge distillation.

Recently, vision transformer based networks have been also applied to compressed domain processing. In [21], a vision transformer is applied to enhance image classification in the compressed domain with an end-to-end joint compression and analysis architecture. Furthermore, in [22], the authors adopt and modify the transformer architecture of [21], training it separately from the compression backbone, to perform image classification specifically in the compressed domain. The work in [23] adopts a transformer-based image compression system offering the flexibility to switch between standard and denoising image reconstruction using the same compressed bitstream and decoding network for both, introducing a latent refinement module and a prompt generator for the last decoding layers.

III. Face detection frameworks and background

A. Pixel and compressed domain approaches

Face detection is typically performed in the pixel domain, taking a digital image as input. However, computer vision tasks are increasingly being conducted on compressed multimedia content rather than original data sources [24], and face detection is no exception. Naturally, face detection is expected to have the best accuracy when applied on uncompressed images, as in the blue block in Fig. 1. When the input image has undergone a lossy encoding/decoding process, as in the green block in Fig. 1, it inevitably impacts the detection performance. This issue becomes particularly significant at lower compression rates, where higher distortion in the reconstructed image leads to impactful coding artifacts. Consequently, this degradation in

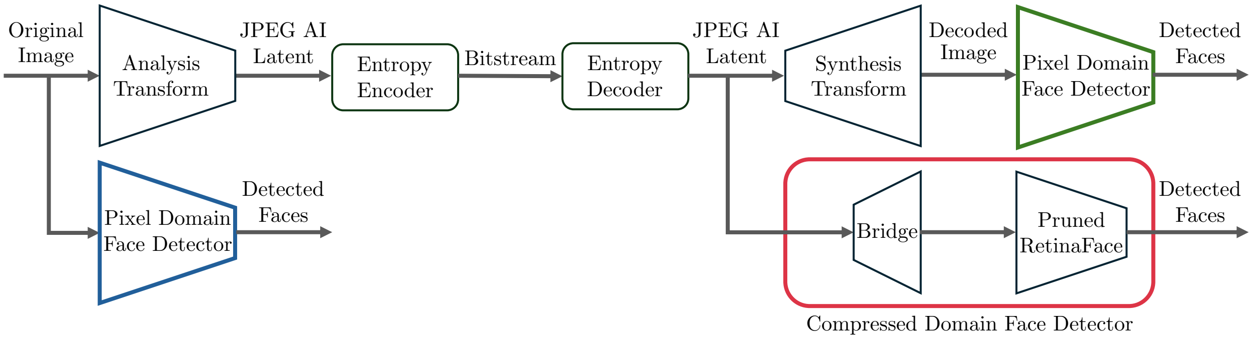


Fig. 1: General face detection framework, illustrating three possible scenarios. A face detector such as RetinaFace can operate in the pixel domain on either an original, uncompressed image (blue block) or a fully decoded image, thus after a JPEG AI encoding/decoding process (green block). On the other hand, the proposed compressed domain face detector architecture operates in the latent space, thus after the encoding process but without fully decoding the image (red block). The depicted high-level view of its architecture includes the bridge replacing the first layers of the face detector (in this case RetinaFace).

image quality adversely affects the performance of computer vision tasks in the pixel domain like face detection [24]–[26].

This paper instead follows the JPEG AI paradigm, where face detection can be directly performed on the compressed domain rather than on the decoded image (see the red block in Fig. 1). In this scenario, image reconstruction can be skipped altogether since the compact latent representation of the image contains enough information to perform both computer vision and image processing tasks. More details on JPEG AI and its philosophy are provided in the next section.

B. JPEG AI

JPEG AI [9] is a new learning-based visual compression project expected to be a standard at the end of 2024. The goal of the JPEG AI standard is to obtain a substantial improvement over the existing JPEG standards regarding both human perception and downstream vision consumption systems.

In the JPEG AI framework, to encode an input image, the latter first goes through an analysis network which is in essence a decorrelating non-linear transformation. This process consists of convolutional layers with learnable filters, incorporating spatial down-scaling in some layers, followed by non-linear activations. Subsequently, the output of these layers is quantized to produce a latent representation, which is a compact form of the input image. The final step involves entropy coding, to eliminate statistical redundancy and generate the final bitstream intended for storage or transmission.

On the decoder side, the bitstream is parsed and entropy decoded, and latent prediction is performed to obtain the latent representation which can then be used by a synthesis transform to reconstruct the original image for human consumption. The alternative is to perform compressed domain image processing tasks, such as super-resolution, inpainting, and color correction, or compressed domain computer vision tasks like classification, semantic segmentation, and face detection.

C. RetinaFace detector

RetinaFace [12] is a single-stage deep learning model tailored for accurate and efficient face detection across various scales and orientations. Notably, it can also predict 3D face shape information. Training is driven by a multi-task loss function, including face classification, facial landmark regression, face box regression, and dense regression loss terms.

The architecture of RetinaFace comprises three key components (see also the bottom half of Fig. 2). First, the backbone acts as the main feature extractor. One of the several possible backbones employed by RetinaFace is ResNet-50, which consists of five stages containing multiple units. Each unit is a residual block consisting of several layers with skip connections. Next, a Feature Pyramid Network (FPN) serves as the component enabling face detection across multiple scales. It generates feature maps at various scales using fully convolutional layers P2 to P6. The FPN is built using both bottom-up and top-down pathways, complemented by lateral connections to ensure comprehensive coverage of different face sizes and orientations. Finally, context modules are employed on feature pyramids to capture more contextual information around each face and effectively detect occluded faces [27]. These modules expand the receptive field, improving performance [28].

IV. Proposed compressed domain face detector

The high-level overview of the proposed compressed domain RetinaFace-based detector is depicted in the red block of Fig. 1, and its detailed architecture is illustrated in Fig. 2. It introduces two key innovative elements: the **bridge** and the **pruned face detector**.

The bridge initially processes the JPEG AI latent input, and its output is then forwarded to the pruned face detector. The primary role of the bridge is to align the latent representation with the feature map at the input of the pruned detector. Since the JPEG AI latent representation is already rich with low-level features obtained through the convolutional layers of the JPEG AI encoding network, it is advantageous to prune the face detector by removing some early stages typically responsible for low-level feature extraction.

Sec. IV-A provides additional details on the architectural design, while Sec. IV-B describes the training process.

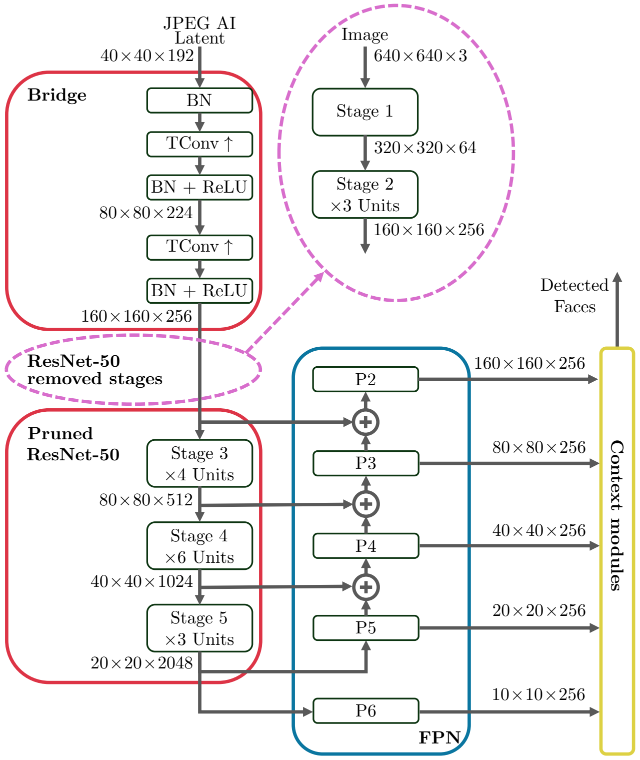


Fig. 2: Detailed architecture of the proposed compressed domain RetinaFace-based detector. Stages 1 and 2 of ResNet-50 are replaced by the bridge, while the rest of RetinaFace (FPN and context modules) is unchanged.

A. Detector pruning and bridge design

We considered three possible pruning points in the early stages of RetinaFace. Following the experimental analysis of which we defer the details to Sec. V-E, the optimal RetinaFace pruning point entails removing stages 1 and 2 from ResNet-50, while leaving the FPN structure intact. This results in the removal of the first 11 convolutional layers from the backbone.

Thus, the proposed bridge architecture, depicted in Fig. 2, begins with a Batch Normalization (BN) layer at the input, followed by two transposed convolution (TConv) layers that each up-sample the feature maps by a factor of 2 and increase the number of channels by 32. The bridge TConv layers are each followed by a BN layer and a ReLU activation. This design ensures that the bridge incorporates an up-sampling factor of 4, thus aligning the JPEG AI latent dimension with the feature map size at the pruned detector input.

B. Training process

A large number of latent representations was first collected by encoding the images from the training set of the WIDER FACE dataset [29] with JPEG AI. The images are encoded using five rate anchors: 0.06 bpp, 0.12 bpp, 0.25 bpp, 0.50 bpp, and 0.75 bpp. For each rate, the proposed compressed domain face detector was then trained using the same settings and hyperparameters, accounting for pruning, as those used for training RetinaFace [12]. In particular, the training process employs the SGD optimizer with a momentum of 0.9, a weight decay of $5 \cdot 10^{-4}$, and a batch size of 24. The learning rate starts at 10^{-3}, increases to 10^{-2} after 5 epochs, and is then reduced by a factor of 10 at 55 and 68 epochs.

Note that the training process in [12] employs Data Augmentation (DA), which is crucial for improving the detection accuracy. While DA is straightforward in the pixel domain, it is more challenging to be performed in the compressed domain. In this work, we resort instead to include the latent representations of the augmented images during training.

When employing RetinaFace in the pixel domain (both on original and decoded images) as a baseline for the experiments in Sec. V, we use the pre-trained models provided by [12], thus no training is necessary in those cases.

V. Performance Assessment

A. Test conditions and metrics

The WIDER FACE dataset [29] is a widely recognized and challenging benchmark for face detection, known for its variability in scale, pose, expression, occlusions, and illumination. The dataset is split into 3 detection difficulty levels: easy, medium, and hard. The dataset randomly allocates 40% of the data for training, 10% for validation, and 50% for testing.

The performance metric employed for evaluation is the Average Precision (AP), one of the most commonly used metric for this purpose, which evaluates the Area Under the Curve (AUC) of the resulting precision-recall curve.

B. Analysis of face detection performance on decoded images

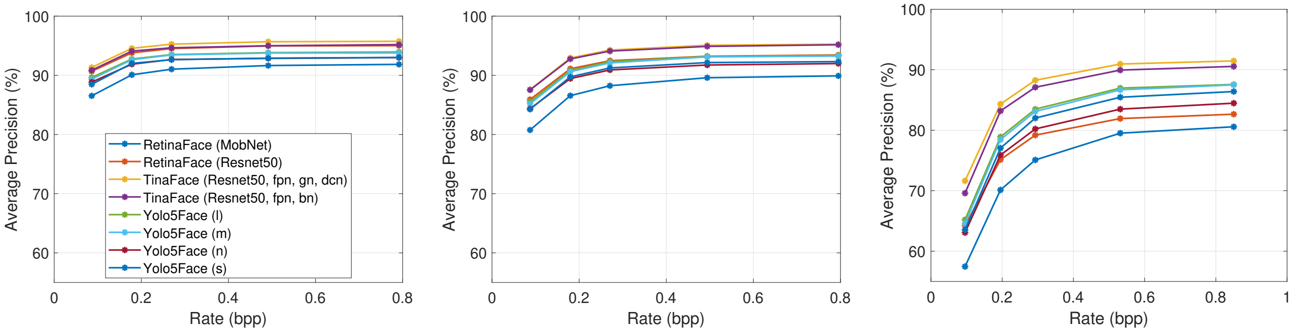
The first objective is to evaluate the impact of JPEG AI coding artifacts on face detection. Three recent models are considered: RetinaFace [12], TinaFace [13], and YOLO5Face [14], each involving different backbones and settings. Initially, face detection is performed in the pixel domain on the original images. Then, the images are compressed using the JPEG AI v4.4 tools-on GPU codec at the aforementioned 5 rate anchors and face detection is performed using the decoded images.

The experimental results are illustrated in Fig. 3. They demonstrate that high performance can be achieved using decoded images as inputs for face detectors. However, as expected, the performance of all considered detectors degrades at lower bitrates, particularly for the hard level dataset.

C. Face detection results in the compressed domain

Fig. 4 illustrates the performance of the proposed compressed domain RetinaFace-based detector (yellow curve) compared with RetinaFace using uncompressed (blue curve) and decoded (red curve) images. Similarly to using decoded images as input, the performance of the proposed method is bitrate-dependent, achieving higher performance at higher bitrates and lower performance at lower bitrates.

However, the results indicate that using the latent representation as input can achieve performance comparable with using decoded images instead. For instance, the performance gap is under 5% AP for the easy level and under 7% AP for the medium level. Instead, this gap increases for the hard level,

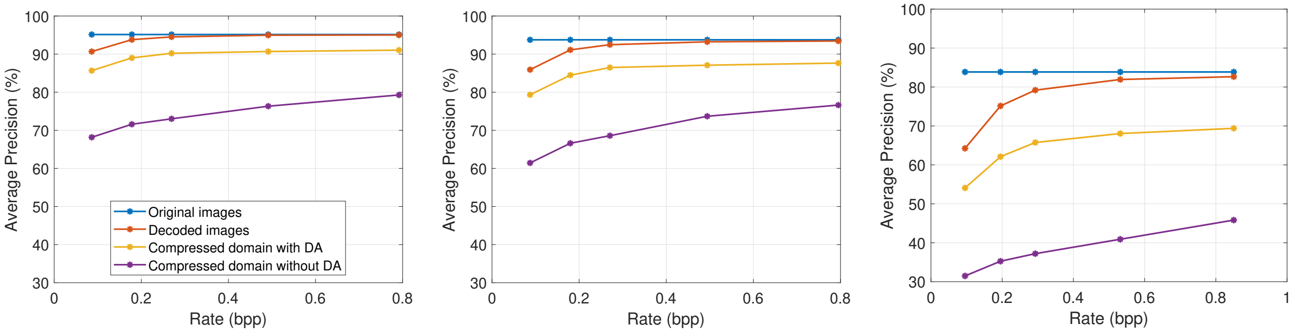


(a) Easy level.

(b) Medium level.

(c) Hard level.

Fig. 3: Face detection performance in the decoded image domain using various detectors under different settings.



(a) Easy level.

(b) Medium level.

(c) Hard level.

Fig. 4: Comparison of face detection performance between: (i) RetinaFace on uncompressed images (blue curve); (ii) RetinaFace on JPEG AI decoded images (red curve); (iii) proposed compressed domain RetinaFace-based detector using JPEG AI latent space as input with data augmentation (yellow curve); and (iv) proposed compressed domain RetinaFace-based detector using JPEG AI latent space as input without data augmentation (purple curve).

suggesting that the JPEG AI latent representation may miss important features in more challenging images.

It is important to remark that the proposed method is able to reduce the overall complexity by skipping the image decoding process. The complexity is analyzed in the next section.

D. Complexity results

In addition to the significant complexity reduction gained from performing computer vision tasks directly in the latent domain, the proposed architecture further simplifies the face detector by removing the first 11 convolutional layers from the ResNet-50 model. On the other hand, it was included a bridge consisting of only two transposed convolutional layers.

Table I shows that the complexity of the proposed compressed domain RetinaFace-based detector is reduced by 81.34 GMAC compared to the original image domain RetinaFace, which also includes the complexity of the JPEG AI decoder.

E. Ablation studies

To determine the optimal detector pruning strategy, three pruning points were identified, the resulting models for each configuration were trained, and their performance on the validation set at 0.75 bpp rate was evaluated.

The first pruning point involved removing stages 1 and 2 as well as the initial two convolutional layers in the first unit of stage 3 within ResNet-50, resulting in the elimination of 13 convolutional layers. For the second pruning point, stages 1 and 2 were removed, leading to the removal of 11 convolutional layers. Finally, the third pruning point involved removing stage 1 and the first unit of stage 2, resulting in the elimination of 5 convolutional layers. The corresponding performance evaluation, obtained without employing DA, is detailed in Table II. The results indicate that the second pruning point achieved the best performance.

Regarding DA, the latent representation of the augmented images was also extracted and used for training. The purple line in Fig. 4, instead, shows the performance of the proposed compressed domain face detector without including the latent

TABLE I: Complexity (GMAC) comparison between pixel and compressed domain approaches for 640×640 input images.

Approach	Architecture	Complexity (GMAC)
Pixel domain	JPEG-AI decoder	90.52
	RetinaFace	44.56
	Total	**135.08**
Compressed domain	Bridge	15.69
	Pruned RetinaFace	38,05
	Total	**53.74**

TABLE II: Pruning point selection: detection performance for various detector pruning point locations at 0.75 bpp.

Level	Pruning point 1	Pruning point 2	Pruning point 3
Easy	71.39%	79.28%	77.83%
Medium	67.68%	76.62%	75.20%
Hard	39.89%	45.82%	43.93%

TABLE III: Impact on face detection accuracy when testing images at single scale (0.75 bpp).

Level	Multiresolution	Single scale
Easy	95.01%	90.05%
Medium	93.44%	88.13%
Hard	82.65%	67.36%

representations of the augmented images in training. It is clear that the performance significantly degrades without DA.

Furthermore, it is important to note that the image domain RetinaFace detector performs inference on testing images at multiple target sizes, which significantly impacts detection accuracy. However, this approach has not been adopted here.

To elaborate, Table III reports the RetinaFace performance when using decoded images (at 0.75 bpp) at a fixed single scale. The performance degrades compared to using a multi-resolution approach. For example, the performance for the hard level when tested at a single scale is 67.36% AP, while the proposed model achieves 69.39% AP at the same rate. This suggests that the proposed compressed domain detector performance could also exceed that of the pixel domain detector at the same conditions. Thus, implementing multi-resolution inference in the proposed technique would likely narrow the performance gap with the image domain RetinaFace.

VI. Conclusions

In this paper, a compressed domain RetinaFace-based detector that operates on JPEG AI latent representations is proposed. Our results demonstrate comparable performance to the original RetinaFace when processing decoded images from the WIDER FACE dataset, especially for images classified as easy and medium difficulty levels. Importantly, our approach eliminates the need for image decoding, thereby reducing complexity. Additionally, it was assessed how the JPEG AI codec compression artifacts at different rates affect face detection accuracy, using various state-of-the-art face detection methods. Finally, our ablation studies suggest that implementing multi-resolution inference in the compressed domain could lower the performance gap with the image domain RetinaFace. This investigation will be conducted in future works.

References

[1] M. Broz, How many photos are there? (statistics & trends in 2024), `https://phototutorial.com/photos-statistics`, Last accessed on 2024-06-25.

[2] G. K. Wallace, The JPEG still picture compression standard, IEEE Trans. Consum. Electron. 38 (1) (1992) xviii–xxxiv.

[3] J. Lainema, F. Bossen, W. J. Han, J. Min, K. Ugur, Intra coding of the HEVC standard, IEEE Trans. Circuits Syst. Video Technol. 22 (12) (2012) 1792–1801.

[4] J. Ballé, V. Laparra, E. P. Simoncelli, End-to-end optimized image compression, in: Proc. Int. Conf. Learn. Repr. (ICLR), 2017.

[5] G. Toderici, D. Vincent, N. Johnston, S. Jin Hwang, D. Minnen, J. Shor, M. Covell, Full resolution image compression with recurrent neural networks, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2017, pp. 5306–5314.

[6] Y.-H. Ho, C.-C. Chan, W.-H. Peng, H.-M. Hang, M. Domański, Anfic: Image compression using augmented normalizing flows, IEEE Open J. Circuits Syst. 2 (2021) 613–626.

[7] L. Duan, J. Liu, W. Yang, T. Huang, W. Gao, Video coding for machines: A paradigm of collaborative compression and intelligent analytics, IEEE Trans. Image Proc. 29 (2020) 8680–8695.

[8] W. Gao, S. Liu, X. Xu, M. Rafie, Y. Zhang, I. Curcio, Recent standard development activities on video coding for machines, preprint arXiv:2105.12653 (2021).

[9] J. Ascenso, E. Alshina, T. Ebrahimi, The JPEG AI standard: Providing efficient human and machine visual data consumption, IEEE Multimedia 30 (1) (2023) 100–111.

[10] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, A. Zisserman, The Pascal Visual Object Classes (VOC) challenge, Int. J. Comput. Vis. 88 (2010) 303–338.

[11] S. Minaee, P. Luo, Z. Lin, K. Bowyer, Going deeper into face detection: A survey, preprint arXiv:2103.14983 (2021).

[12] J. Deng, J. Guo, E. Ververas, I. Kotsia, S. Zafeiriou, RetinaFace: Single-shot multi-level face localisation in the wild, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2020, pp. 5203–5212.

[13] Y. Zhu, H. Cai, S. Zhang, C. Wang, Y. Xiong, TinaFace: Strong but simple baseline for face detection, preprint arXiv:2011.13183 (2020).

[14] D. Qi, W. Tan, Q. Yao, J. Liu, YOLO5Face: Why reinventing a face detector, in: Proc. ECVA Eur. Conf. Comput. Vis. (ECCV), 2022, pp. 228–244.

[15] R. Torfason, F. Mentzer, E. Agustsson, M. Tschannen, R. Timofte, L. Van Gool, Towards image understanding from deep compression without decoding, preprint arXiv:1803.06131 (2018).

[16] Y. Mei, F. Li, L. Li, Z. Li, Learn a compression for objection detection – VAE with a bridge, in: Proc. IEEE Int. Conf. Visual Comm. Image Process. (VCIP), 2021, pp. 1–5.

[17] Y. Deng, L. J. Karam, DNN-compressed domain visual recognition with feature adaptation, preprint arXiv:2305.08000 (2023).

[18] J. Liu, H. Sun, J. Katto, Learning in compressed domain for faster machine vision tasks, in: Proc. IEEE Int. Conf. Visual Comm. Image Process. (VCIP), 2021, pp. 01–05.

[19] Z. Wang, M. Qin, Y.-K. Chen, Learning from the CNN-based compressed domain, in: Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV), 2022, pp. 3582–3590.

[20] J. Liu, H. Sun, J. Katto, Semantic segmentation in learned compressed domain, in: Proc. IEEE Picture Coding Symp. (PCS), 2022, pp. 181–185.

[21] Y. Bai, X. Yang, X. Liu, J. Jiang, Y. Wang, X. Ji, W. Gao, Towards end-to-end image compression and analysis with transformers, in: Proc. AAAI Conf. Artif. Intell., Vol. 36(1), 2022, pp. 104–112.

[22] R. Ji, L. J. Karam, Compressed-domain vision transformer for image classification, IEEE J. Emerg. Sel. Topics Circuits Syst. (2024).

[23] Y.-H. Chen, K.-W. Ho, S.-R. Tsai, G.-H. Lin, A. Gnutti, W.-H. Peng, R. Leonardi, Transformer-based learned image compression for joint decoding and denoising (2024). arXiv:2402.12888.

[24] A. Seleem, A. F. Guarda, N. M. Rodrigues, F. Pereira, Deep learning-based compressed domain multimedia for man and machine: A taxonomy and application to point cloud classification, IEEE Access 11 (2023) 128979–128997.

[25] N. Bousnina, J. Ascenso, P. L. Correia, F. Pereira, Impact of conventional and AI-based image coding on AI-based face recognition performance, in: Proc. IEEE Eur. Workshop Vis. Inf. Process., 2022, pp. 1–6.

[26] J. Liu, H. Sun, J. Katto, Improving multiple machine vision tasks in the compressed domain, in: Proc. IEEE Int. Conf. Pattern Recognit. (ICPR), 2022, pp. 331–337.

[27] P. Hu, D. Ramanan, Finding tiny faces, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2017, pp. 951–959.

[28] Y. Feng, S. Yu, H. Peng, Y.-R. Li, J. Zhang, Detect faces efficiently: A survey and evaluations, IEEE Trans. Biometrics Behav. Identity Sci. 4 (1) (2021) 1–18.

[29] S. Yang, P. Luo, C.-C. Loy, X. Tang, WIDER FACE: A face detection benchmark, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2016, pp. 5525–5533.