PADAM: PERCEPTUAL AUDIO DEFECT ASSESSMENT MODEL

Alex Mackin[*]*, Pratha Khandelwal*[*]*, Veneta Haralampieva, Michael Lau,*
Benoit Vallade, David Higham, Josh Anderson

Amazon Prime Video, London, UK

ABSTRACT

We introduce PADAM, a no-reference perceptual model for automated detection of audio defects in professional media content. Our three-stage architecture identifies seven common audio defects through perceptual modeling, combining feature extraction, quality-aware contrastive learning, and robust classification. To address the scarcity of labeled training data, we develop a synthetic defect generation workflow that replicates professional media production pipelines, enabling self-supervised learning guided by audio quality metrics. PADAM achieves 0.75 F1-score on real-world defects during offline evaluation, outperforming existing methods while maintaining robust performance in production.

***Index Terms*—** Audio defect detection, perceptual quality assessment, self-supervised learning, professional media

1. INTRODUCTION

Audio defects in professional media content significantly impact Quality of Experience (QoE), a critical concern as digital platforms become the primary means of content consumption worldwide [1]. While video quality has historically received more research and industry attention, both audio and video quality are essential factors affecting viewer experience [2].

Despite rigorous quality control processes, audio defects like clipping, quantization errors, and packet loss [3] can persist in professional media workflows due to the variety of content sources, and the complexity of processing and distribution chains [4]. Though relatively rare in professional media content, their impact can be particularly severe if high-profile.

The detection of audio defects faces two major challenges: distinguishing between intentional creative effects and unintentional defects [4], and bridging the gap between automated detection and human perception [5]. Modern production frequently employs creative techniques that intentionally incorporate artifacts like quantization and digital distortion, as well as environmental effects such as rain, wind, or background noise. This creative freedom in professional content necessitates context-aware detection approaches [6], as traditional signal-processing methods focusing on specific defect types [7] prove inadequate for diverse content at scale.

[*]Equal Contribution

The key contributions of our work are summarized as:

- A novel three-stage perceptual model for audio defect detection at scale that eliminates the need for costly human annotations through contrastive learning.
- A fusion quality metric used within self-supervised training through soft quality-based clustering.
- A comprehensive synthetic defect generation workflow that simulates seven real-world audio defects by replicating professional media processing scenarios.

2. RELATED WORK

Audio quality assessment metrics fall into Full-Reference (FR) and No-Reference (NR) approaches. Traditional FR metrics like PESQ [8] use psychoacoustic models to compare degraded audio against reference signals, while VISQOL [9] improved accuracy through spectrogram-based comparisons. CDPAM [10] advanced FR assessment through SimCLR-style [11] contrastive learning with audio augmentations.

For NR assessment, most approaches focus on specific domains. Quality-Net [12] introduced an end-to-end BLSTM-based approach for speech quality evaluation, while DNSMOS [13] specifically targets noise suppression assessment. SRMR [14] uses modulation spectral representation for speech intelligibility, and NISQA [15] predicts speech quality dimensions through self-attention mechanisms. More recently, SCOREQ [16] advanced speech quality assessment through contrastive regression, though Huang *et al.* [17] demonstrated that many speech quality models struggle to generalize beyond their training domains. Saishu *et al.* [18] proposed a parallel CNN architecture to identify specific degradation types, while CORN [19] demonstrated that co-training FR and NR models can improve performance.

Audio defect detection in professional media content faces unique challenges, requiring identification of diverse technical problems across multiple content types. Alonso *et al.* [4] developed algorithms for common defects such as loudness and noise using linear prediction and spectral analysis, but these methods require careful parameter tuning. Audio Artifacts (AA) [3] uses transfer learning from PANN [20] to classify four defect types in short audio segments. However, this approach was limited by existing datasets focusing on speech-based problems [21, 22] rather than media defects.

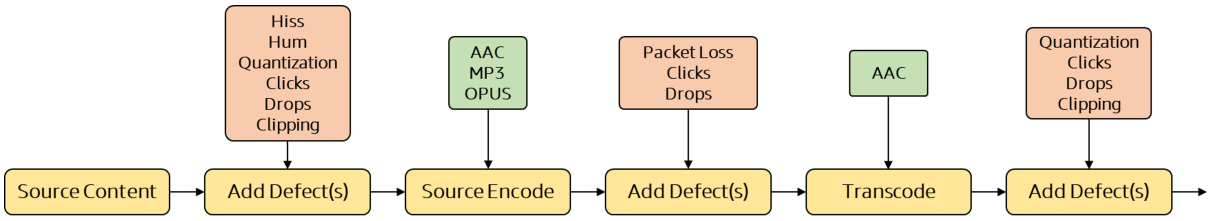

Fig. 1: Synthetic audio defect generation workflow illustrating the three-stage defect injection process (source, source encode, and transcode stages).

In video quality assessment, large-scale perceptual models are being trained without human annotations using self-supervised approaches: CSPT-VQA [23] and CONVIQT [24] use contrastive learning with distortion identification, while RankDVQA [25] clusters using VMAF [26] rankings.

3. DATASET

This section describes our dataset creation process: selecting clean source content, synthesizing representative audio defects, and collecting quality scores for model training.

3.1. Source Content

The absence of large-scale public datasets and rarity of defects in professional media necessitated developing a synthetic generation workflow (Fig.1) that emulates realistic production pipelines. Inspired by BVI-Artefact [27], we generate seven key audio defects across three stages (source, source encode, and transcode). We randomly sample 250K 10-second stereo clips from our internal datasets, creating clean/defective pairs for training (75%) and validation (25%). While most clips contain single defects (65%), we also generate multiple defects (25%) and clean samples (10%).

3.1.1. Synthetic Audio Defect Generation

Based on industry experience, we identified and implemented seven key audio defects commonly found in production content. All defects can affect either random or all channels:

1. **Hum** [28]: Interference from electrical equipment, manifesting as persistent tonal noise. Generated as weighted harmonics: $\sum_{h\in P} w_h A_h \sin(2\pi h f_0 t + \phi)$ with SNR 0-15 dB, where $A_h = \frac{1}{h}$ for square wave or $(-1)^h/h^2$ for triangle wave, w_h are harmonic weights, ϕ is the phase offset in $[-2\pi, 2\pi]$, and P is harmonic pattern (even: $2i$, odd: $2i-1$ or triplen: $3(2i-1)$, where $i \in \mathbb{Z}^+$). Fundamental frequency f_0 is either 50/60 Hz (power-line) or uniform in [40, 500] Hz (other sources).
2. **Hiss** [4]: Broadband background noise, similar to white/pink noise. Generated as colored noise with spectrum $S(f) \propto 1/f^\beta$, $\beta \in [-2, 2]$ and SNR 0-15 dB. We apply a highpass filter with cutoff frequency $f_{\min}$ to attenuate low-frequency content, where $f_{\min}$ is set as [1e-5, 0.5] relative to the Nyquist frequency.
3. **Drops/Ticks/Stutter** [29]: Brief signal interruptions, causing audible gaps or repetitions. Generated as 1-7 interruptions/second i.i.d., each 10-80 ms, and implemented as zeros, sample hold, or sample removal.
4. **Clipping** [30]: Amplitude saturation causing flattened peaks and harmonic distortion. Symmetric thresholds at $(p_c/2)$th and $(100 - p_c/2)$th percentiles, where $p_c \in [7\%, 35\%]$. Minimum 0.4s between intervals. Consistent thresholds maintained across affected channels.
5. **Quantization** [31]: Limited amplitude resolution causing stepwise distortion and noise. Bit depth reduction to 4/6/8 bits with optional dithering (RPDF/TPDF) and noise shaping. Levels calculated as $(2^b/2) + 1$.
6. **Packet Loss** [32]: Missing audio segments from transmission errors in digital streams. We use FFmpeg noise filter with noise amount [1000-9000] and optional drop amount [10000-20000], and resampling compensation.
7. **Clicks** [6]: Brief impulse noises causing pops and crackles, common in damaged media. We generate 2-8 clicks/second, duration 0.06-0.6 ms, amplitude 0.3-1.0. Amplitude signs chosen to minimize clipping.

3.1.2. Encoding and Transcoding

To capture interactions between compression and audio defects, our workflow implements a two-stage compression process: source encoding and transcoding. Source encoding is applied with 50% probability using AAC, MP3, or Opus codecs with common profiles, formats, cutoff frequencies (12-16 kHz), sample rates (32-48 kHz), and bitrates (32-128 kbps). Transcoding is applied with 50% probability using AAC, with optional loudness normalization (-23 LUFS) and low-pass filtering (3.5-7 kHz cutoff) each at 10% probability.

3.2. Objective Quality Scores

We conducted a targeted perceptual study following ITU-R BS.1534 [33] and ACR methodology [34], where 12 participants rated 616 randomly selected 10s clips on a five-point quality scale. We selected four metrics with highest correlation to human annotations: PESQ [8], CDPAM [10], ViSQOLv3 [9], and DNSMOS [35]. Following CORN [19], we define quality metric $\hat{q} = \sum_m w_m(\frac{x_m - \mu_m}{\sigma_m})$ with weights $w_{\text{CDPAM}} = 0.40$, $w_{\text{DNSMOS}} = 0.23$, $w_{\text{PESQ}} = 0.19$, $w_{\text{ViSQOL}} = 0.18$, where x_m is the raw metric score.

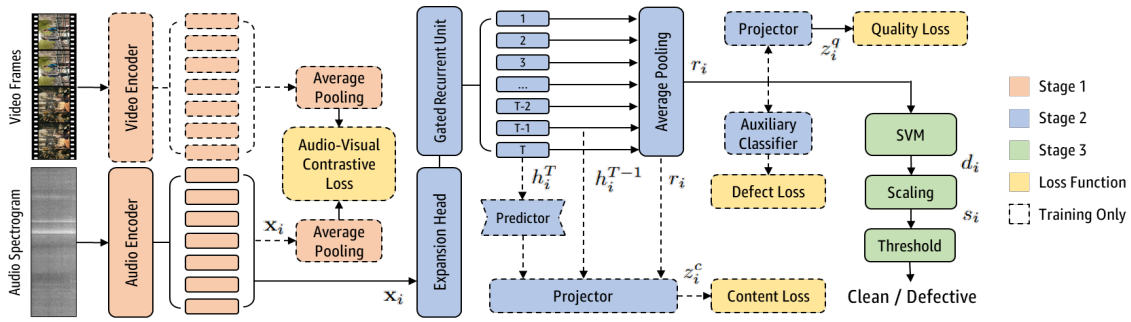

Fig. 2: PADAM architecture with three independently trained stages (preceding stages frozen during subsequent training): (1) Feature Extractor for robust audio representations, (2) Quality Head for perceptual modeling, and (3) SVM Classifier for defect identification. *Training only* components removed at inference.

The fused quality metric $\hat{q}$ achieves superior correlation with human subjective scores compared to individual metrics (see Table 1). Given the moderate correlation of $\hat{q}$ (<0.8), we designed the quality head in Section 4.2 to use soft assignments rather than hard thresholds used in RankDVQA [25] and RMT-BVQA [36] to better handle this uncertainty.

Metric	PESQ	CDPAM	DNSMOS	VISQOL	$\hat{q}$
SROCC (↑)	0.469	0.592	0.325	0.543	**0.596**
LCC (↑)	0.416	0.514	0.320	0.483	**0.558**
RMSE (↓)	0.958	0.903	0.998	0.922	**0.873**

Table 1: Correlation analysis of subjective scores, where SROCC = Spearman Rank Order Correlation Coefficient, LCC = Linear Correlation Coefficient, and RMSE = Root Mean Square Error. Best results in **bold**.

4. METHODOLOGY

PADAM is a no-reference model for perceptually-aligned audio defect assessment. Its three-stage architecture leverages recent advances: (1) generic embeddings from an audio foundation models [37], (2) a quality head combining content-aware [23] and quality-aware perceptual [25, 36] clustering, and (3) a robust SVM classifier for production deployment.

4.1. Stage 1: Feature Extractor

We trained a multi-modal feature extractor to create generic 133ms-resolution audio embeddings using audio-visual contrastive learning inspired by [38]. The architecture combined an AST audio encoder [37] (12 layers, 6 heads, processing 16kHz audio into mel-spectrograms with 128 mels × 41 time steps, producing 3×384 embeddings) with a ViViT video encoder [39] (12 layers, 12 heads, processing 144×256 frames into 5×384 embeddings) that is discarded after training. Training uses bidirectional InfoNCE losses [40] between synchronized audio-video time-averaged embedding pairs as positives and different clean clips as negatives, with temperature $\tau_1 = 0.1$. We trained using 64 V100 GPUs on 1M clean clips from the same content sources as the synthetic dataset, using Adam optimizer with one-cycle learning rate ($1e{-4}$), gradient clipping, and horizontal flipping augmentation.

4.2. Stage 2: Quality Head

Let $\mathbf{x}_i \in \mathbb{R}^{T\times 384}$ represent the embeddings from our frozen audio feature extractor, where $T = 75$ (10s). This embedding passes through a two-layer MLP $f_{\text{head}} : \mathbb{R}^{384} \to \mathbb{R}^{384} \to \mathbb{R}^{512}$, followed by a GRU [41] producing hidden states $h_i^t \in \mathbb{R}^{512}$. We then define two further MLPs: a projector $f_{\text{proj}} : \mathbb{R}^{512} \to \mathbb{R}^{512} \to \mathbb{R}^{128}$ for clustering and a temporal predictor $f_{\text{pred}} : \mathbb{R}^{512} \to \mathbb{R}^{128} \to \mathbb{R}^{512}$. All MLPs use GELU activation and 25% dropout. Following [36], both quality and content losses use masked contrastive learning [42, 43] with batch size B:

$$\mathcal{L}_{\text{q/c}} = \frac{1}{B}\sum_{i=1}^{B} \frac{\sum_{j=1}^{B} m_{ij} \log\left(\frac{\exp(z_i \cdot z_j/\tau)}{\sum_{k=1}^{B} m_{ik}\exp(z_i \cdot z_k/\tau)}\right)}{\sum_{j=1}^{B} m_{ij}} \quad (1)$$

where z_i denotes either quality-aware z_i^q or content-aware z_i^c embeddings (superscripts index embedding type). For the quality loss ($\mathcal{L}_{\text{q}}$) [25], $z_i^q = f_{\text{proj}}(r_i)$ and $r_i = \frac{1}{T}\sum_{t=1}^{T} h_i^t$. Pairs are weighted by the fused quality metric prediction differences. We use a soft assignment mask, $m_{ij}^q = 1 - |\hat{q}_i - \hat{q}_j|$, as opposed to hard thresholding (binary mask) used in [25, 36].

For temporal stability [23, 36], we define content loss ($\mathcal{L}_{\text{c}}$) where $m_{ij}^c = 1$ for pairs from the same clip (identity) and:

$$z_i^c = f_{\text{proj}}\left(\text{concat}\left[\frac{1}{T-1}\sum_{t=1}^{T-1} h_i^t, f_{\text{pred}}(h_i^T), h_i^{T-1}\right]\right) \quad (2)$$

The overall loss $\mathcal{L} = \frac{1}{2}\mathcal{L}_{\text{d}} - \mathcal{L}_{\text{q}} - \mathcal{L}_{\text{c}}$ includes 7-label binary cross-entropy loss $\mathcal{L}_{\text{d}}$ after auxiliary classifier: $\mathbb{R}^{512} \to \mathbb{R}^{7}$. The model was trained using SGD with one-cycle learning rate ($2e{-4}$), weight decay ($1e{-5}$), batch size 256, gradient clipping, and $\tau_2 = 0.2$. We only compute r_i during inference.

4.3. Stage 3: SVM Classifier

While Stage 2's multi-label auxiliary classifier guides representation learning, we use an SVM [44] with RBF kernel for robust binary defect detection. The outlier fraction parameter ν and kernel coefficient γ are optimized via random search. Scores are scaled as $s_i = (1 + \exp(-d_i/\sigma))^{-1}$ where σ is the root mean square (RMS) of all SVM scores d_i in the training set. We threshold these scores for binary classification. A regression model could be used instead for quality estimation.

5. RESULTS

We conduct two analyses: (1) a small-scale offline evaluation of known defects, and (2) a large-scale evaluation on production traffic to assess performance under realistic conditions.

5.1. Offline Evaluation

Our offline test set comprises 596 clean and 10 defective videos, covering TV and movies that range from 20 minutes to 2 hours. The defective videos contain stutter, hiss and quantization defects. While small due to rarity of real-world known defects, this dataset enables model comparison.

We evaluate PADAM against five existing no-reference models: AA [3], DNSMOS (P.835) [35], NISQA (v2) [15], SRMR [14] and SCOREQ [16]. All baselines except AA are speech-focused and make predictions in 10-second segment intervals. AA targets four general audio defects with 2-second segments (hiss is the only overlap with our defect set). There are 645 defective segments (0.56%) and 112,412 clean segments (99.4%). A segment is marked as a true positive when it overlaps with a defect. For fair comparison, thresholds are optimized to maximize segment-level F1-score on the test set.

Model	Threshold	Precision (↑)	Recall (↑)	F1-Score (↑)
AA	-	0.02	0.50	0.04
DNSMOS	2.12	0.03	0.27	0.05
NISQA	1.11	0.03	**0.58**	0.05
SRMR	0.33	0.19	0.38	0.25
SCOREQ	1.24	0.03	0.26	0.05
PADAM	0.67	**0.79**	0.56	**0.66**

Table 2: Performance comparison at segment level. AA uses a multi-class classifier, meaning no binary threshold can be computed. Best results in **bold**.

As shown in Table 2, PADAM substantially outperforms existing methods. While baseline models like NISQA can achieve competitive recall, they suffer from low precision due to their speech-focus. AA shows similar limitations, amplified by its shorter 2s segment analysis leading to increased likelihood of false detections. PADAM's high F1-score suggests it can effectively distinguish between actual defects and content-specific variations in professional media.

Model	Filter	Threshold	Precision (↑)	Recall (↑)	F1-Score (↑)
Speech Metrics					
DNSMOS	100s/100s	2.12	0.67	0.20	0.31
NISQA	120s/120s	1.11	0.04	0.20	0.07
SRMR	100s/100s	0.28	0.43	0.30	0.35
SCOREQ	20s/20s	0.99	0.01	0.10	0.02
PADAM					
Feature Extractor	45s/45s	0.63	0.20	0.80	0.32
+ Quality Head	90s/110s	0.62	0.57	0.40	0.47
+ SVM Classifier	20s/80s	0.67	**1.00**	0.60	**0.75**

Table 3: Performance comparison of baseline models and ablation study of PADAM's components. We jointly optimize window parameters and detection thresholds to maximize F1-score. For PADAM's Feature Extractor and Quality Head, we train a linear probe on the training set. Best results in **bold**.

To enhance precision, we implement a temporal filter requiring X seconds of detections within a Y-second rolling window. Since our goal is title-level detection, we consider any overlap between detected segments (after filtering) and ground truth timestamps as a title-level true positive. We exclude Audio Artifacts due to different segment sizes (2s).

As shown in Table 3, baseline models DNSMOS and SRMR achieve reasonable precision, but their specialization in speech defects limits overall coverage. PADAM's Feature Extractor shows strong recall from its generic representation learning, while the Quality Head's perceptual modeling improves discrimination. The SVM classifier handles complex decision boundaries needed for clustered features, enabling accurate detection with considerably shorter windows.

5.2. Production Evaluation

We evaluated PADAM on 17K titles from our video-on-demand catalog. The model detected defects in 135 titles (0.8%). Expert content operators performed manual review of these detections, categorizing them into:

- **Genuine Defects (35)**: Distortion (15), Clipping (9), Noise (7), Other (4)
- **Creative Intent (92)**: Environmental (37), Rain (27), Tone/Buzz (9), Channel Effects (7), Other (12)
- **No Issues / False Positives (8)**

While PADAM's precision for genuine defects is 25.9%, the low positive prediction rate (0.8%) ensures manageable review volume. Considering both defects and creative intent as valid detections yields 94.1% precision. The primary challenge we face is detecting creative intent (68.1%) - intentional audio effects that share similar characteristics with defects. For example, Rain sounds similar to Hiss. These findings suggest content understanding could help differentiate between intentional effects and defects. We cannot calculate recall as manual review of 17K titles is infeasible. Due to production constraints we cannot add these titles to our offline test set.

6. CONCLUSIONS

In this paper we present PADAM, a three-stage model for reference-free audio defect detection in professional media content. Our architecture combines feature extraction, quality-aware contrastive learning, and robust classification, leveraging a novel defect generation pipeline that realistically simulates seven distinct audio defects. Offline evaluation shows PADAM outperforms existing methods on real content. During production deployment we find that while effective at detecting technical defects, distinguishing creative intent remains challenging, accounting for 68% of issues. Production content presents complex scenarios with overlapping effects and content-specific variations. Future work should explore multi-modal content understanding and more accurate perceptual quality metrics to improve self-supervised training.

7. REFERENCES

[1] Z. Akhtar and T. Falk, "Audio-visual multimedia quality assessment: A comprehensive survey," *IEEE Access*, vol. 5, pp. 21090–21117, 2017.

[2] J. Beerends and F. De Caluwe, "The influence of video quality on perceived audio quality and vice versa," *J Audio Eng Soc*, vol. 47, no. 5, pp. 355–362, 1999.

[3] D. Higham, A. Bagla, and V. Haralampieva, "A no-reference model for detecting audio artifacts using pretrained audio neural networks," in *Proc IEEE/CVF WACV*, 2022, pp. 9–13.

[4] P. Alonso-Jiménez, L. Joglar-Ongay, X. Serra, and D. Bogdanov, "Automatic detection of audio problems for quality control in digital music distribution," in *Proc Audio Eng Soc*, 2019.

[5] S. Möller and A. Raake, *Quality of Experience*, Springer, 2014.

[6] D. Wolff, R. Mignot, and A. Roebel, "Audio defect detection in music with deep networks," *arXiv preprint arXiv:2202.05718*, 2022.

[7] M. Brandt and J. Bitzer, "Automatic detection of hum in audio signals," *J Audio Eng Soc*, vol. 62, no. 9, pp. 584–595, 2014.

[8] A. Rix, J. Beerends, M. Hollier, and A. Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc IEEE ICASSP*, 2001, vol. 2, pp. 749–752.

[9] M. Chinen, F. Lim, J. Skoglund, and N. Gureev et al., "ViSQOL v3: An open source production ready objective speech and audio metric," in *Proc Int Conf QoMEX*, 2020, pp. 1–6.

[10] P. Manocha, Z. Jin, R. Zhang, and A. Finkelstein, "CDPAM: Contrastive learning for perceptual audio similarity," in *Proc IEEE ICASSP*, 2021, pp. 196–200.

[11] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc Int Conf Mach Learn*, 2020, pp. 1597–1607.

[12] S. Fu, Y. Tsao, H. Hwang, and H. Wang, "Quality-net: An end-to-end non-intrusive speech quality assessment model based on BLSTM," *arXiv preprint arXiv:1808.05344*, 2018.

[13] C. Reddy, V. Gopal, and R. Cutler, "DNSMOS: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors," in *Proc IEEE ICASSP*, 2021, pp. 6493–6497.

[14] T. Falk, C. Zheng, and W. Chan, "A non-intrusive quality and intelligibility measure of reverberant and dereverberated speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 7, pp. 1766–1774, 2010.

[15] G. Mittag, B. Naderi, A. Chehadi, and S. Möller, "NISQA: A deep cnn-self-attention model for multidimensional speech quality prediction with crowdsourced datasets," *arXiv preprint arXiv:2104.09494*, 2021.

[16] A. Ragano, J. Skoglund, and A. Hines, "SCOREQ: Speech quality assessment with contrastive regression," in *Advances in Neural Information Processing Systems*, 2024, vol. 37, pp. 105702–105729.

[17] W. Huang, E. Cooper, and T. Toda, "MOS-Bench: Benchmarking generalization abilities of subjective speech quality assessment models," *arXiv preprint arXiv:2411.03715*, 2024.

[18] Y. Saishu, A. Poorjam, and M. Christensen, "A CNN-based approach to identification of degradations in speech signals," *EURASIP J Audio Speech Music Process*, vol. 2021, no. 1, pp. 9, 2021.

[19] P. Manocha, D. Williamson, and A. Finkelstein, "CORN: Co-trained full-and no-reference speech quality assessment," in *Proc IEEE ICASSP*, 2024, pp. 376–380.

[20] Q. Kong, Y. Cao, T. Iqbal, and Y. Wang et al., "PANNs: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Trans Audio Speech Lang Process*, vol. 28, pp. 2880–2894, 2020.

[21] H. Dubey, A. Aazami, V. Gopal, and B. Naderi et al., "ICASSP 2023 deep noise suppression challenge," *IEEE Open J Signal Process*, 2024.

[22] S. Quackenbush and R. Lefevbre, "Performance of MPEG unified speech and audio coding," in *Audio Engineering Society Convention 131*. Audio Engineering Society, 2011.

[23] P. Chen, L. Li, J. Wu, W. Dong, and G. Shi, "Contrastive self-supervised pre-training for video quality assessment," *IEEE Trans Image Process*, vol. 31, pp. 458–471, 2021.

[24] P. Madhusudana, N. Birkbeck, Y. Wang, B. Adsumilli, and A. Bovik, "ConViQT: Contrastive video quality estimator," *IEEE Trans Image Process*, vol. 32, pp. 5138–5152, 2023.

[25] C. Feng, D. Danier, F. Zhang, and D. Bull, "RankDVQA: Deep VQA based on ranking-inspired hybrid training," in *Proc IEEE/CVF WACV*, 2024, pp. 1648–1658.

[26] Z. Li, C. Bampis, J. Novak, and A. Aaron et al., "VMAF: The journey continues," *Netflix Tech Blog*, vol. 25, no. 1, 2018.

[27] C. Feng, D. Danier, F. Zhang, A. Mackin, A. Collins, and D. Bull, "BVI-Artefact: An artefact detection benchmark dataset for streamed videos," in *Proc Picture Coding Symp*, 2024, pp. 1–5.

[28] R. Dobre, V. Niţă, A. Ciobanu, C. Negrescu, and D. Stanomir, "A hum removal algorithm used for audio restoration purposes," in *Proc Int Symp Signals Circuits Syst*, 2015, pp. 1–4.

[29] R. Mühlbauer, "Automatic audio defect detection," *Bachelor Thesis, Johannes Kepler Univ Linz*, 2010.

[30] ITU-R, "Algorithms to measure audio programme loudness and true-peak audio level," Tech. Rep. BS.1770-4, International Telecommunication Union, 2015.

[31] E. Murray, M. Kasher, and P. Spasojevic, "Optimizing audio compression through entropy-controlled dithering," *arXiv preprint arXiv:2501.02293*, 2025.

[32] C. Perkins, O. Hodson, and V. Hardman, "A survey of packet loss recovery techniques for streaming audio," *IEEE Network*, vol. 12, no. 5, pp. 40–48, 1998.

[33] ITU-R, "Method for the subjective assessment of intermediate quality level of audio systems," Tech. Rep. BS.1534-3, International Telecommunication Union, 2015.

[34] ITU-T, "Subjective video quality assessment methods for multimedia applications," Tech. Rep. P.910-5, International Telecommunication Union, 2021.

[35] C. Reddy, V. Gopal, and R. Cutler, "DNSMOS P.835: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors," in *Proc IEEE ICASSP*, 2022, pp. 886–890.

[36] T. Peng, C. Feng, D. Danier, and F. Zhang et al., "RMT-BVQA: Recurrent memory transformer-based blind video quality assessment for enhanced video content," *arXiv preprint arXiv:2405.08621*, 2024.

[37] Y. Gong, Y. Chung, and J. Glass, "AST: Audio spectrogram transformer," *arXiv preprint arXiv:2104.01778*, 2021.

[38] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," in *European conference on computer vision*. Springer, 2020, pp. 776–794.

[39] A. Arnab, M. Dehghani, G. Heigold, and C. Sun et al., "ViViT: A video vision transformer," in *Proc IEEE/CVF ICCV*, 2021, pp. 6836–6846.

[40] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[41] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Gated feedback recurrent neural networks," in *Proc Int Conf Mach Learn*, 2015, pp. 2067–2075.

[42] P. Khosla, P. Teterwak, C. Wang, and A. Sarna et al., "Supervised contrastive learning," *arXiv preprint arXiv:2004.11362*, 2020.

[43] J. Cui, Z. Zhong, S. Liu, B. Yu, and J. Jia, "Parametric contrastive learning," *arXiv preprint arXiv:2107.12028*, 2021.

[44] B. Schölkopf, A. Smola, R. Williamson, and P. Bartlett, "New support vector algorithms," *Neural Comput*, vol. 12, no. 5, pp. 1207–1245, 2000.