國立臺灣大學電機資訊學院資訊網路與多媒體研究所

碩士論文

Graduate Institute of Networking and Multimedia

College of Electrical Engineering and Computer Science

National Taiwan University

Master's Thesis

Pali-VQA：基於 PaliGemma 2 的多級別無參考影片品質評估模型

Pali-VQA: A PaliGemma 2-Based Multi-Level Blind Video Quality Assessment Model

梁家綸

Chia-Lun Liang

指導教授: 廖世偉 博士

Advisor: Shih-Wei Liao, Ph.D.

中華民國 114 年 8 月

August 2025

doi:10.6342/NTU202501694

國立臺灣大學碩士學位論文
口試委員會審定書
MASTER'S THESIS ACCEPTANCE CERTIFICATE
NATIONAL TAIWAN UNIVERSITY

Pali-VQA：基於 PaliGemma 2 的多級別無參考影片品質評估模型

Pali-VQA: A PaliGemma 2-Based Multi-Level Blind Video Quality Assessment Model

本論文係 梁家綸（學號 R11944064）在國立臺灣大學資訊網路與多媒體研究所完成之碩士學位論文，於民國 114 年 7 月 31 日承下列考試委員審查通過及口試及格，特此證明。

The undersigned, appointed by the Graduate Institute of Networking and Multimedia on 31 July 2025 have examined a Master's Thesis entitled above presented by LIANG, CHIA-LUN (student ID: R11944064) candidate and hereby certify that it is worthy of acceptance.

口試委員 Oral examination committee:

廖世偉		
（指導教授 Advisor）		
蘇柏超		貢敖群

系（所）主管 Director: 鄭卜壬

國立臺灣大學碩士學位論文
口試委員會審定書
MASTER'S THESIS ACCEPTANCE CERTIFICATE
NATIONAL TAIWAN UNIVERSITY

Pali-VQA：基於 PaliGemma 2 的多級別無參考影片品質評估模型

Pali-VQA: A PaliGemma 2-Based Multi-Level Blind Video Quality Assessment Model

本論文係 梁家綸 （學號 R11944064）在國立臺灣大學資訊網路與多媒體研究所完成之碩士學位論文，於民國 114 年 7 月 31 日承下列考試委員審查通過及口試及格，特此證明。

The undersigned, appointed by the Graduate Institute of Networking and Multimedia on 31 July 2025 have examined a Master's Thesis entitled above presented by LIANG, CHIA-LUN (student ID: R11944064) candidate and hereby certify that it is worthy of acceptance.

口試委員 Oral examination committee:

（指導教授 Advisor）		

系（所）主管 Director: 鄭卜壬

Acknowledgements

要感謝的人太多了，很擔心有所疏漏。

首先感謝指導教授廖世偉老師，提供諸多實驗室的資源與研究計劃，並在百忙之中給予許多指導和鼓勵，我才能順利找到研究方向並放手一搏。感謝 Google UR Program 的所有前輩，在每次的 Meeting 中給予的回饋都讓我有諸多成長。特別感謝 Fangting 每次針對研究方向給出的具體指引，讓我能在最有前瞻性的研究上一步一步做改善。我還想感謝網媒所、資工所的老師和系辦同仁，打造資源這麼豐腴的學術環境，讓我學習許多知識，成長茁壯。

此外也感謝在實驗室遇到所有的同學，包含 VQA 及 CUJ 組的大家。不管是每次去 Google 吃漢堡、開會前趕報告、還是實驗室內的點點滴滴，都充實了我的碩士生活。感謝身旁所有的摯友忍受我的牢騷，陪我談心、吃飯，滋潤我疲憊的心靈。最重要也最感謝我的家人，使我不用顧慮經濟壓力完成學業，並總給予我極大的支持。你們是我最強大的支柱。

最後感謝堅持到現在的自己，在這個暫時的終點，我將休息片刻，然後眺望遠方，繼續向崇山峻嶺走去。

doi:10.6342/NTU202501694

doi:10.6342/NTU202501694

摘要

隨著使用者創作影片的迅速增加，開發穩定且自動化影片品質評估（VQA）的方法已變得至關重要。儘管大型多模態模型（LMMs）已為無參考影片品質評估（BVQA）帶來了進展，但現行方法仍普遍將品質評估轉化爲一個粗略的五級分類問題，因此限制了模型在影片品質上微小差異的辨識能力。在本文中，我們提出 Pali-VQA，一種高效、基於 LMM 的 BVQA 模型，通過引進多級別評分架構來打破此一限制。Pali-VQA 奠基於 PaliGemma 2，將 BVQA 重新定義為一個最多可達 18 個不同評分等級的細緻分類問題。我們採用低秩自適應（LoRA）進行微調，並結合次序迴歸標籤平滑技術，在正則化模型的同時保留評分等級之間的內在的順序資訊。儘管我們僅在單一數據集上做 LoRA 微調，但在四個實景 VQA 基準測試的實驗中，Pali-VQA 取得了具競爭力的表現，足以媲美或超越那些參數量更大、進行完整微調或使用集成方法的模型。此外，當 Pali-VQA 與非 LMM 的深度神經網路（DNN）BVQA 模型 FAST-VQA 進行集成時，Pali-VQA 更在基準資料集中的三個上超越了所有先前的模型。我們的研究結果顯示，提升評分等級的數量能顯著提高預測表現，為基於 LMM 的影片品質評估方法提供了一條更經濟、更有效的途徑。

關鍵字：影片品質評估、大型多模態模型、PaliGemma、序數迴歸

doi:10.6342/NTU202501694

doi:10.6342/NTU202501694

Abstract

The exponential increase in user-generated video content necessitates robust automated methods for Video Quality Assessment (VQA). While Large Multimodal Models (LMMs) have propelled advances in Blind VQA (BVQA), current approaches typically frame quality prediction as a coarse, five-level classification task, limiting their ability to discern fine-grained video quality differences. In this paper, we introduce Pali-VQA, an efficient LMM-based BVQA model that addresses this limitation by incorporating a multi-level rating framework. Built on the PaliGemma 2 backbone, Pali-VQA reframes BVQA as a fine-grained classification problem parameterized with a maximum of 18 distinct rating levels. We employ Low-Rank Adaptation (LoRA) for fine-tuning and incorporate an ordinal regression label smoothing technique to preserve the inherent ordinal information among rating levels while regularizing the model. Despite being fine-tuned using LoRA on only a single dataset, our experiments on four in-the-wild VQA benchmarks show that Pali-VQA achieves competitive performance, matching or outperforming larger,

doi:10.6342/NTU202501694

fully fine-tuned, or ensemble models. Moreover, when ensembled with FAST-VQA, a non-LMM Deep Neural Network (DNN) BVQA model, Pali-VQA outperforms all previous top models on three of the four datasets. Our findings demonstrate that increasing the granularity of the rating levels significantly enhances predictive performance, offering a more efficient and effective path to LMM-based video quality assessment.

Keywords: Video Quality Assessment, Large Multimodal Model, PaliGemma, Ordinal Regression

doi:10.6342/NTU202501694

Contents

Page

口試委員審定書 i

Acknowledgements iii

摘要 v

Abstract vii

Contents ix

List of Figures xiii

List of Tables xv

Chapter 1 Introduction 1

Chapter 2 Related Work 5

2.1 Knowledge-based BVQA Methods 5

2.2 Non-LMM Deep BVQA Methods 6

2.3 LMM-based BVQA Methods 7

Chapter 3 VQA Datasets 9

3.1 LSVQ Dataset 9

3.2 KoNViD-1k Dataset 10

3.3 LIVE-VQC Dataset 11

3.4 Distribution of MOS Values 12

doi:10.6342/NTU202501694

3.5 Statistical Significance of MOS Differences 12
3.5.1 Pairwise Significance Threshold 13
3.5.2 Summary of Thresholds 14

Chapter 4 Methodology **17**

4.1 Video Preprocessing 18
4.2 Quality Prompt 18
4.3 MOS-to-Level Mapping 18
4.4 LMM Backbone 19
4.5 Prediction Score Calculation 20
4.6 Multi-Level Rating Framework 21
4.7 Ordinal Regression Label Smoothing 21

Chapter 5 Experiments **25**

5.1 Fine-tuning Settings 25
5.2 Evaluation Benchmarks 28
5.3 Evaluation Metrics 28
5.3.1 Pearson Linear Correlation Coefficient (PLCC) 28
5.3.2 Spearman's Rank Order Correlation Coefficient (SRCC) 29
5.4 Inference Setting 30
5.5 Results 32
5.6 Discussion 35
5.7 Ablation Study 35
5.7.1 Multi-level Rating Scheme 36
5.7.2 Ordinal Regression Label Smoothing 38

doi:10.6342/NTU202501694

Chapter 6 Conclusion 41

References 43

doi:10.6342/NTU202501694

doi:10.6342/NTU202501694

List of Figures

3.1 Distribution of MOS for in-the-wild VQA datasets. The orange bars denote the MOS histogram and the red curve the Gaussian KDE. From left to right (top row): the LSVQ training split (n = 28,056), its held-out test split (n = 7,182), and the high-definition 1080p subset (n = 2,575). Bottom row: LIVE-VQC (n = 88), and KoNViD-1k (n = 1,200). All datasets exhibit a strong concentration of scores around the midpoint, underscoring the challenge of discriminating moderately rated videos. 13

4.1 Overview of the Pali-VQA pipeline. Video frames sampled at 0.5 fps are center-cropped and resized to 448×448 pixels. These preprocessed frames are fed to the PaliGemma 2 (3B) backbone together with the rating prompt. A local closed-set softmax is applied to the logits corresponding to the predefined rating tokens, producing a probability distribution that is finally used to calculate the final predicted score at inference time. . . . 17

4.2 Probability distributions for ordinal regression label smoothing on a 14-level classification scheme (true index = 3, 7 and 11). The absolute-distance metric (blue) yields a gently decaying probability mass around the true level, while the squared-distance metric (orange) concentrates probability more sharply—producing a faster decay for more distant levels. . . 23

doi:10.6342/NTU202501694

5.1 Dual inference strategy for PaliVQA. At test time, video frames are sampled at 1 fps and partitioned into two disjoint sets based on frame indices (even vs. odd). Each subset is independently processed by PaliVQA, producing quality scores. The final prediction is computed by averaging both scores. 30

doi:10.6342/NTU202501694

List of Tables

3.1 Minimum, median, and maximum values of the pairwise significance threshold Diff^{min} (MOS points on a 1–5 scale) across all KoNViD-1k video pairs. 15

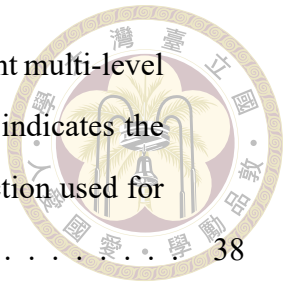
4.1 Mapping of rating levels to descriptors for different classification schemes 22

5.1 VQA performance on the LSVQ-1080p (2,575 videos) and LSVQ-test (7,182 videos) benchmarks. Each benchmark is evaluated using SRCC, PLCC, and their average (Avg). Rankings are highlighted by formatting: **bold + underline** marks the top performer, **bold** alone denotes second place, and underline marks the third. The "Ensemble" column flags methods that use model ensembling, and "Train sets" specifies the datasets used during training/fine-tuning. 33

5.2 VQA performance on out-of-distribution benchmarks: LIVE-VQC (88 videos) and KoNViD-1k (1200 videos). Each benchmark is evaluated using SRCC, PLCC, and their average (Avg). Rankings are highlighted by formatting: **bold + underline** marks the top performer, **bold** alone denotes second place, and underline marks the third. The "Ensemble" column flags methods that use model ensembling, and "Train sets" specifies the datasets used during training/fine-tuning. 34

5.3 Ablation study on the number of rating levels. For each level, we show the performance of individual training configurations, averaged across four benchmarks (LSVQ-1080p, LSVQ-test, LIVE-VQC, KoNViD-1k). The **Overall Avg.** for each level block represents the mean performance across all its configurations. 37

doi:10.6342/NTU202501694



5.4 Classification accuracy across five quality classes for different multi-level schemes and training configurations. The “Level” column indicates the multi-level scheme, “Smoothing” refers to the distance function used for label smoothing, and “LR” is the learning rate. 38

5.5 Ablation study on the label smoothing scheme. We compare ordinal regression label smoothing using absolute distance metrics (Absolute), squared distance metrics (Squared), and a baseline with no explicit smoothing (None). For each scheme, we list the performance of every configuration, averaged across four benchmarks (LSVQ-1080p, LSVQ-test, LIVE-VQC, KoNViD-1k). The **Overall Avg.** represents the mean performance for that scheme across all its tested configurations. 39

doi:10.6342/NTU202501694

Chapter 1 Introduction

Video content has dominated the digital landscape, fundamentally transforming the way users engage with media on the internet. The rapid expansion of video content makes it infeasible for service providers to ensure satisfactory Quality of Experience (QoE) for end users through manual video-quality inspection [50]. Furthermore, most online videos are user-generated content (UGC), which is captured by a variety of devices under diverse conditions and thus subject to complex distortions, such as overexposure, uneven lighting, jitter, motion blur, compression artifacts, noise, color shifts, and lens blur [34, 49].

To address these challenges, numerous Video Quality Assessment (VQA) algorithms have been developed. The primary objective of VQA is to develop methods that are capable of predicting perceptual video quality scores in close agreement with human mean opinion scores (MOS) obtained from subjective studies [65]. Because undistorted reference videos are rarely available in real-world scenarios, researchers have increasingly focused on the Blind Video Quality Assessment (BVQA) task [24], where models estimate perceptual video quality without access to pristine reference videos. BVQA approaches are commonly classified as either knowledge-driven or data-driven [30]. Knowledge-driven methods extract handcrafted features informed by the human visual system [22, 31, 32, 59]. In contrast, data-driven approaches either leverage pretrained backbones (e.g., ResNet [12] or CLIP [36]) with fixed parameters to extract high-level representations or

doi:10.6342/NTU202501694

learn features directly from raw videos via end-to-end training [23, 26, 40, 41, 51, 53–55, 58, 62, 63].

Recently, large Multimodal Models (LMMs) have risen to prominence in computer vision thanks to their remarkable zero-shot generalization across diverse tasks [3, 5, 25, 28, 39, 60], and many BVQA approaches have leveraged LMMs to evaluate video quality. Q-Align [56] fine-tuned mPLUG-Owl2 [60] on VQA datasets by framing quality prediction as a classification task. VQA² [21] adopted a three-stage training pipeline using LLaVA-OneVision-Chat-7B [25] together with a SlowFast [9] backbone. In [52], the authors designed an ensemble framework that combines PaliGemma 2 [39], a Google-designed LMM, with existing BVQA models such as FAST-VQA [53] and DOVER [55], fusing their predictions via a learnable weight α. FVQ-Rater [57] employed InternVL 2.5 [4] as its LMM backbone and applied Low-Rank Adaptation (LoRA) [18] to align features extracted by other vision encoders. At their core, LMMs are designed to minimize cross-entropy loss over sequences of tokens. This makes them naturally adept at classification, yet poorly suited for regression. To overcome this limitation, most LMM-based BVQA approaches forgo predicting a continuous MOS; instead, they quantize MOS values into a set of discrete levels. This approach, coined regression via classification (RvC), has been widely adopted across various machine learning domains to solve regression problems [11, 14, 44]. Nevertheless, current LMM-based BVQA methods are limited to coarse five-level discretization schemes. For instance, studies [56, 57] converted MOS values into five categories (excellent, good, fair, poor, bad). Similarly, [21] translated MOS values into five discrete levels (high, good, fair, poor, low) during the second training stage. Additionally, [52] described video frame quality through five gradations: high, medium high, medium, medium low, and low. Such coarse discretization can limit model capacity

doi:10.6342/NTU202501694

by discarding valuable information and compromise fine-grained ranking accuracy when the number of classes is insufficient [45].

In this paper, we propose **Pali-VQA**, an efficient LMM-based BVQA model built on Google's PaliGemma 2 [39]. Through experiments with different numbers of quality levels, we demonstrate that increasing the number of discrete categories enhances Pali-VQA's performance on BVQA tasks, as evaluated by both Pearson linear correlation coefficient (PLCC) and Spearman's rank-order correlation coefficient (SRCC). Following [6], we adopt an order-aware label smoothing strategy that redistributes probability mass based on inter-level distances. This technique allows us to capture the ordinal information inherent in these rating levels. Furthermore, we fine-tune the mixed checkpoints of PaliGemma 2, which have been pre-trained on a wide array of vision-language tasks, using LoRA. This lightweight update modifies only a small subset of parameters, while achieving accuracy comparable to state-of-the-art LMM-based VQA models.

In summary, our contributions are threefold.

- We introduce **Pali-VQA,** an LMM-based BVQA model built on PaliGemma 2 that uses only LoRA fine-tuning and matches or surpasses state-of-the-art LMM-based BVQA methods.
- We systematically explore categorization schemes beyond the standard five-level configuration used in most LMM-based BVQA approaches and show that increasing the number of rating levels further enhances BVQA performance.
- We apply a label-smoothing method borrowed from ordinal-regression techniques to preserve the inherent ordinal information among quality levels.

doi:10.6342/NTU202501694

doi:10.6342/NTU202501694

Chapter 2 Related Work

2.1 Knowledge-based BVQA Methods

Knowledge-driven methods design handcrafted features to assess video quality [22, 31, 32, 59]. For instance, Video Intrinsic Integrity and Distortion Evaluation Oracle (VIIDEO) [31] is built on the idea that natural, pristine videos follow specific statistical patterns, whereas distorted videos deviate from these regularities. VIIDEO begins by computing normalized frame differences across two spatial-frequency bands and partitioning them into small patches. Within each patch, the method calculates products of neighboring values in four orientations, fitting the resulting distributions using an Asymmetric Generalized Gaussian Distribution (AGGD). The AGGD parameters extracted from these fits comprise the model' s statistical feature vector. Subsequently, VIIDEO tracks temporal differences in these features, and the final score is computed by aggregating inter-scale correlation coefficients averaged across the entire video. However, when dealing with low-quality, multi-distorted UGC content, deep learning-based BVQA models are more capable of capturing the underlying statistical structures in videos and aligning them with human judgments of visual quality, especially in BVQA scenarios [65].

doi:10.6342/NTU202501694

2.2 Non-LMM Deep BVQA Methods

Data-driven approaches based on deep neural networks (DNNs), requiring fewer parameters than LMM, have excelled in BVQA tasks in the past decade and remain widely adopted. FAST-VQA [53], Dover [55], and ModularBVQA [51] serve as the leading baselines in this category.

FAST-VQA [53] adopts a sampling strategy termed fragment. It breaks each high-resolution video frame into small patches, and then reassembles them into full frames to reduce the computational cost for high-resolution videos. DOVER [55] ensembles two complementary branches to assess video quality. The aesthetic branch evaluates high-level perceptual cues, such as semantics and composition, while the technical branch based on FAST-VQA focuses exclusively on low-level distortions, such as compression artifacts and blur. Meanwhile, ModularBVQA [51] combines a base quality predictor along with spatial and temporal rectifiers. Each component is explicitly crafted to assess visual content, spatial resolution, and frame rate, thus allowing the model to accurately account for their impacts on perceived video quality.

Although non-LMM deep BVQA methods possess great potential for video quality assessment, their full capabilities are mainly constrained by the scarcity of large-scale psychometric datasets and our limited understanding of how viewers perceive visual quality [65].

doi:10.6342/NTU202501694

2.3 LMM-based BVQA Methods

Several LLMs have shown impressive few-shot and zero-shot video understanding through in-context learning, achieving state-of-the-art performance across diverse visual benchmarks [1, 29, 39, 60]. Similarly, VQA tasks also benefit when structured as question-answering problems, allowing for enhanced understanding of video artifacts through integrated reasoning across visual and textual modalities.

For example in [56], the authors proposed Q-Align, a VQA model based on the open-source LMM mPLUG-Owl2 [60]. They trained the LMM to rate either videos or images by converting MOS values into five discrete rating levels. The model outperforms several non-LMM DNN models on benchmarks for Image Quality Assessment (IQA), Image Aesthetic Assessment (IAA), and VQA benchmarks. The authors further introduce OneAlign, which is fine-tuned on a mixed IQA–VQA–IAA dataset, thus unifying all three tasks under a single model. The VQA^2 series models [21] employ LLaVA-OneVision-Chat-7B [25] as their LMM backbones and integrate it with SlowFast [9] for motion features. The models were trained in three stages: distortion-recognition, quality-scoring, and quality understanding, using the VQA^2 Instruction Dataset constructed by the authors in their work. This dataset comprises images and videos sourced from several public IQA/VQA datasets [2, 7, 8, 10, 13, 17, 27, 38, 62], along with question–answer pairs created through collaboration between human annotators and GPT. The instruction dataset is organized according to the three training stages. Specifically, in the second stage (quality-scoring), video MOS values were normalized to a 0 – 100 scale and are subsequently categorized into five discrete quality levels: High, Good, Fair, Poor, and Low.

Besides general LMM-based BVQA methods, several LMM-based models also tar-

doi:10.6342/NTU202501694

get niche scenarios such as Face video quality assessment (FVQA) [57] or VQA for short-form videos [52]. To improve predictions on short-form videos, authors of [52] combine PaliGemma 2 [39] with deep BVQA models by fusing their outputs through a learnable weight α. In [57], the authors introduce FVQ-20K, the first large-scale in-the-wild FVQA dataset, and propose a specialized method called FVQ-Rater. They fine-tuned InternVL 2.5 [4] with SlowFast [9] using LoRA [18] to align all vision encoders with the LLM input space.

Despite their strong performance on UGC BVQA benchmarks, current LMM-based BVQA models require substantial computing power. Fine-tuning VQA^2 takes roughly 15 hours on eight top-tier NVIDIA H800 GPUs, while Q-Align still requires four A100 GPUs. Furthermore, these methods rely on five-class rating schemes to quantize continuous MOS annotations into coarse levels, which may sacrifice fine-grained quality information. [45].

doi:10.6342/NTU202501694

Chapter 3 VQA Datasets

Early VQA datasets apply a single, controlled distortion with algorithmic or codec-based synthetic methods e.g., MPEG-2/H.264 compression to pristine videos [37, 46, 47]. However, these synthetic datasets fail to capture the diverse, mixed artifacts common in user-generated content, where blur, defocus, and multiple compression often appear simultaneously [61]. To address this shortfall, video quality datasets with genuine, in-the-wild distortions [16, 38, 48, 62] have recently become popular, as they more faithfully mirror the UGC videos on social media.

3.1 LSVQ Dataset

The Large-Scale Social Video Quality (LSVQ) dataset [62] is one of the largest in-the-wild VQA benchmarks, comprising 38,811 video clips (5-12 s each) at varying resolutions, each annotated with a MOS on a 0-100 scale. Videos in the LSVQ dataset were obtained from two large-scale public UGC video repositories: Internet Archive (IA) [19] and YFCC100M [43]. Subjective video-quality scores were collected on Amazon Mechanical Turk (AMT), which is a widely used crowdsourcing platform for outsourcing annotation tasks to online workers. In total, 6,284 participants contributed approximately 1.4 million judgments, averaging 35 ratings per video. Each annotation task began with

doi:10.6342/NTU202501694

general instructions, followed by a qualification quiz that participants were required to pass before continuing. Upon successful completion, participants proceeded to the testing phase, where they evaluated 90 videos. During the evaluation process, each video was presented exactly once, and participants rated the perceived quality on a continuous scale ranging from 0 to 100 using a sliding bar. To maintain annotation consistency, four repeated videos and four ”golden” videos (selected from KoNViD-1k [16], with known subjective scores) were included for quality control. Participants whose ratings on these control videos veered significantly from established standards were excluded. As a result, a total of 1,046 participants were disqualified from the study.

3.2 KoNViD-1k Dataset

The Konstanz Natural Video Database 1k (KoNViD-1k) [16] contains 1,200 eight-second videos rated on a 1-5 MOS scale. The clips were uniformly sampled across six key quality attributes (e.g., blur, contrast) to ensure content diversity. Videos in KoNViD-1k were sourced from YFCC-100M [43]. Subjective quality scores for these videos were obtained through crowdsourcing tasks conducted on the CrowdFlower platform. At the start of the task, participants received instructions adapted from VQEG recommendations [20], which explained different types of video distortions and how to assess overall quality. Sample videos rated “Good”, “Fair”, and “Bad” were also shown to help anchor judgments. At the rating stage, participants were required to choose from five quality categories for each video. To manage workload, participants were limited to rating a maximum of 550 videos. For quality control, since no prior ground truth existed for the dataset, a pilot crowdsourcing test was run on 100 randomly selected videos. From those results, 65 videos with strong agreement were identified and their MOS were used as benchmarks

doi:10.6342/NTU202501694

for test questions in the main study. Any participants scoring below 70% accuracy on these were excluded. After quality-control filtering, the study comprised 642 participants from 64 countries. On average, each video received 114 ratings (including test items), and participants achieved a mean accuracy of 94% on those quality-control questions.

3.3 LIVE-VQC Dataset

The LIVE Video Quality Challenge Database (LIVE-VQC) [38] features 585 ten-second clips captured by 101 different devices, including handheld camcorders and smartphones, each annotated with a 0-100 MOS. The videos in the dataset were captured by 80 mostly naive mobile-camera users, primarily volunteers and acquaintances of LIVE (Laboratory for Image & Video Engineering) members worldwide. As with the LSVQ [62] dataset, video-quality scores were obtained via a large-scale subjective study on Amazon Mechanical Turk (AMT). Participants were required to use non-mobile devices (desktops or laptops) with at least 1280 × 720 resolution and to have an AMT reliability score above 90%. Each study session followed a three-stage workflow. In the first stage, participants were presented with the task requirements, rating instructions, and examples of common distortions. This was followed by a training stage, where participants were required to rate seven example videos. After each video finished playing, participants rated its quality using a continuous sliding bar, with the initial cursor position randomized. Participants were excluded at this stage if any single video took longer than 15 seconds to play, or if any three videos each took more than 12 seconds to play. The subsequent testing stage was similar to the training stage but required rating 43 videos in total: four ”golden” videos from the LIVE-VQA database for validation; 31 videos randomly selected from the new distorted-video database; four repeated videos (randomly chosen from that 31-

doi:10.6342/NTU202501694

video pool); and four specific videos that all participants rated. The study collected over 205,000 opinion scores, yielding an average of 240 scores per video. After removing outliers and stalled ratings, approximately 205 valid scores remained per video. Due to the large number of annotations per video, LIVE-VQC offers more robust quality scores than datasets with fewer annotations, resulting in more reliable ground-truth data for training and evaluation.

3.4 Distribution of MOS Values

Notably, all these datasets show a clustering of MOS values around the midpoint, which makes the task of distinguishing videos of moderate quality challenging. Figure 4.2 illustrates this concentration in the MOS distributions. To better capture the characteristics of the data used for fine-tuning and benchmarking our model in section 5.5, we plot on the metadata provided by the VQA² GitHub repository [21, 35]. The metadata excludes any videos that were used during VQA²'s fine-tuning stage.

3.5 Statistical Significance of MOS Differences

VQA benchmarks often use MOS as the ground truth value. However, each MOS is estimated from a limited number of subjective judgments. Due to sampling variability, small MOS differences may occur by chance rather than indicating true perceptual disparities. To investigate the minimum MOS gap required for a statistically significant distinction, we perform the following analysis.

doi:10.6342/NTU202501694

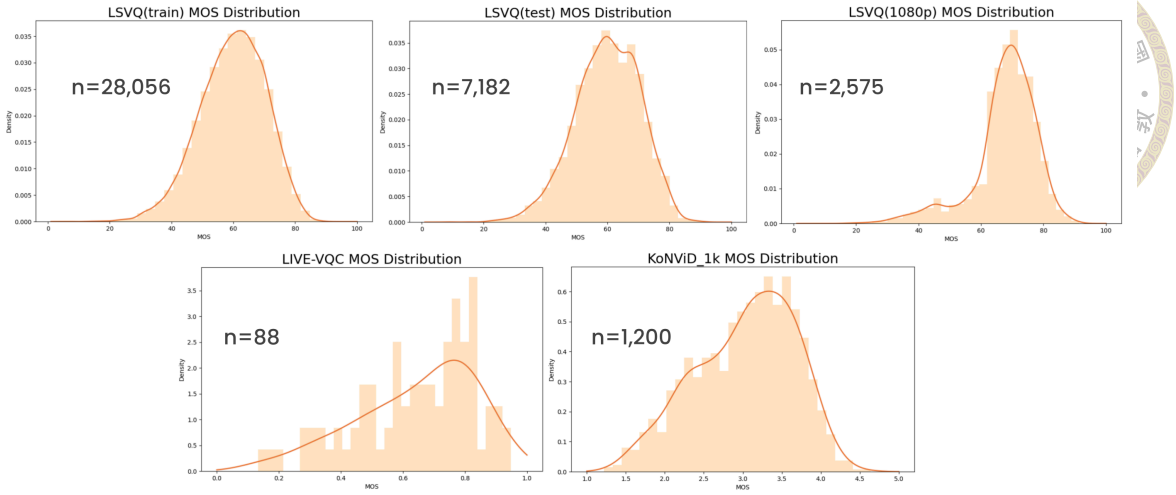


Figure 3.1: Distribution of MOS for in-the-wild VQA datasets. The orange bars denote the MOS histogram and the red curve the Gaussian KDE. From left to right (top row): the LSVQ training split (n = 28,056), its held-out test split (n = 7,182), and the high-definition 1080p subset (n = 2,575). Bottom row: LIVE-VQC (n = 88), and KoNViD-1k (n = 1,200). All datasets exhibit a strong concentration of scores around the midpoint, underscoring the challenge of discriminating moderately rated videos.

3.5.1 Pairwise Significance Threshold

Let $r_{i,j}$ be the rating from observer o_i for video v_j, and let O_j denote the total number of ratings given by the observers for video v_j. We compute the sample mean and standard deviation as

$$s_j = \bar{r}_j = \frac{1}{O_j} \sum_{i=1}^{O_j} r_{i,j}, \tag{3.1}$$

$$\sigma_j = \sqrt{\frac{1}{O_j - 1} \sum_{i=1}^{O_j} \left(r_{i,j} - \bar{r}_j\right)^2}, \tag{3.2}$$

and the corresponding standard error (SE)

$$\mathrm{SE}_j = \frac{\sigma_j}{\sqrt{O_j}}. \tag{3.3}$$

Here, s_j is the sample mean (i.e., the MOS for video v_j), and SE_j reflects the expected variability of s_j if the rating procedure were repeated with O_j observers.

doi:10.6342/NTU202501694

To compare any two videos i and j, we consider their MOS difference

$$\text{Diff}_{i,j} = \left|s_i - s_j\right|.$$

To be deemed statistically significant at the 95% level, we require

$$\text{Diff}_{i,j} > \text{Diff}^{\text{min}}_{i,j} = z_{0.975}\sqrt{\text{SE}_i^2 + \text{SE}_j^2} \approx 1.96\sqrt{\text{SE}_i^2 + \text{SE}_j^2}$$

where

$$z_{0.975} = \Phi^{-1}(0.975), \quad \Phi(z) = \int_{-\infty}^{z} \frac{1}{\sqrt{2\pi}} e^{-t^2/2}\,\mathrm{d}t, \quad Z \sim \mathcal{N}(0,1).$$

Here, $z_{0.975}$ is the 97.5th percentile of the standard normal distribution, satisfying

$$P\big(Z \le z_{0.975}\big) = 0.975.$$

Among the three datasets mentioned earlier, only KoNViD-1k provides the full metadata. Its official repository reports 72,941 crowd-sourced ratings for 1,200 videos [15]. We compute Diff^{min} for all $\binom{1200}{2}$ unordered video pairs in KoNViD-1k.

3.5.2 Summary of Thresholds

Table 3.1 summarizes the minimum, median, and maximum values of the pairwise thresholds Diff^{min}. These results indicate that for the video pair with the lowest variance, a MOS difference of only 0.052 is sufficient to achieve statistical significance (p-value $<$ 0.05). Under typical conditions, represented by the median-variance pair, a MOS difference of approximately 0.142 points is required. For the highest-variance pair, the MOS

doi:10.6342/NTU202501694

difference must reach 0.214 points to be considered statistically significant.

Table 3.1: Minimum, median, and maximum values of the pairwise significance threshold Diff^{min} (MOS points on a 1–5 scale) across all KoNViD-1k video pairs.

Statistic	Minimum	Median	Maximum
Diff^{min}	0.052	0.142	0.214

Although this analysis is based on the KoNViD-1k dataset and its 1-5 MOS scale, the same threshold conditions can be approximately extrapolated to other VQA datasets collected under different scales. By mapping the 1-5 range onto a 0-100 scale (i.e., multiplying by 25), we obtain

$$\text{Diff}^{\text{min}} \approx \{1.3,\ 3.6,\ 5.4\}$$

corresponding to the minimum, median, and maximum cases, respectively. Thus, for a generic VQA dataset reporting MOS on a 0–100 scale, a difference of around 3.6 points can serve as a practical guideline for claiming statistical significance, and up to 5.4 points in high-variance settings.

doi:10.6342/NTU202501694

doi:10.6342/NTU202501694

Chapter 4 Methodology

Our method employs PaliGemma 2 [39] as the LMM backbone. It receives both sampled video frames and quality-assessment prompts as input, and outputs a probability distribution over a fixed set of tokens representing quality levels (e.g. Good, Excellent) after a local softmax layer. We compute the final predicted quality score by weighting each level's quality score by its predicted probability. We term our complete framework Pali-VQA. Figure 4.1 shows an overview of this process.

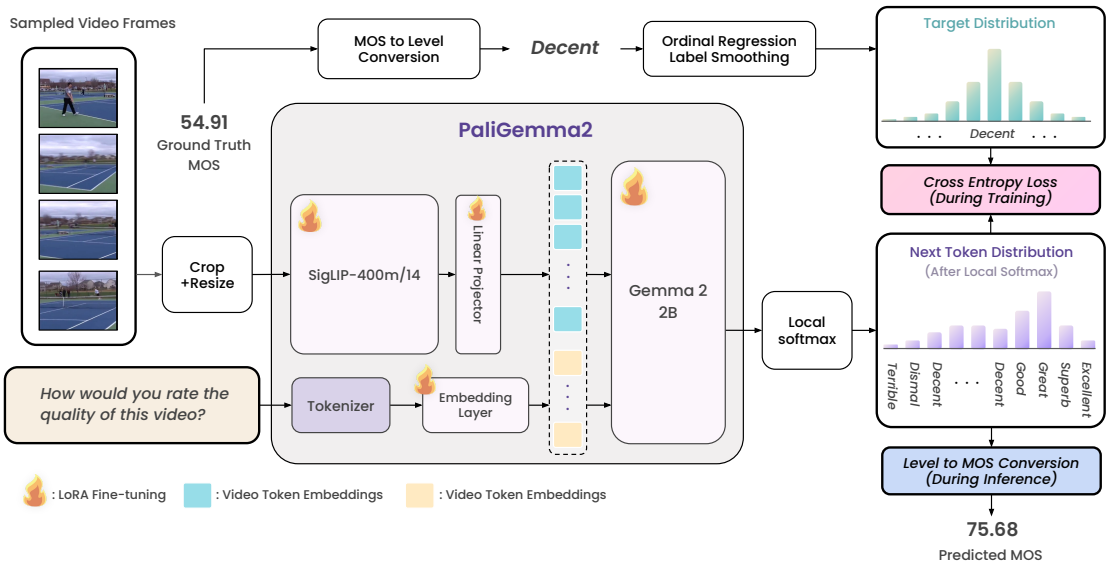


Figure 4.1: Overview of the Pali-VQA pipeline. Video frames sampled at 0.5 fps are center-cropped and resized to 448×448 pixels. These preprocessed frames are fed to the PaliGemma 2 (3B) backbone together with the rating prompt. A local closed-set softmax is applied to the logits corresponding to the predefined rating tokens, producing a probability distribution that is finally used to calculate the final predicted score at inference time.

doi:10.6342/NTU202501694

4.1 Video Preprocessing

Due to computational constraints, we extract one frame every two seconds, up to a maximum of four frames per video. Even for clips shorter than eight seconds, we keep the same interval. For instance a five-second video yields frames at 0, 2, and 4 seconds. Each frame is then center-cropped and rescaled to 448×448 resolution to align with the model's input specifications. We consistently sample from the initial segments of all videos to maintain temporal uniformity across the dataset.

4.2 Quality Prompt

We use the same prompt, "How would you rate the quality of this video?", for every video input, and tokenize it with PaliGemma 2's original tokenizer.

4.3 MOS-to-Level Mapping

Since our LMM outputs discrete tokens rather than continuous values, we quantize each MOS value in the dataset into one of the K rating levels. Formally, we define

$$\mathcal{L} = \{l_1, l_2, \ldots, l_{\mathrm{K}}\}, \quad |\mathcal{L}| = \mathrm{K} \tag{4.1}$$

where $\mathcal{L}$ collects all possible rating levels.

We map any numerical MOS ground-truth value $s \in [\mathrm{m}, \mathrm{M}]$ into one of the K rating levels by uniformly splitting the interval $[\mathrm{m}, \mathrm{M}]$ into K equal-length sub-intervals and then

doi:10.6342/NTU202501694

finding which sub-interval s belongs to. The values m and M denote the minimum and maximum scores on the annotation scale of the dataset (e.g. $\mathrm{m} = 0$ and $\mathrm{M} = 100$ for the LSVQ dataset). Concretely, we define a level mapping function $L(\cdot)$ that maps the original ground-truth score s to some rating level l_i.

$$\mathrm{d} = \frac{\mathrm{M} - \mathrm{m}}{\mathrm{K}}, \quad i = \min\Big(\mathrm{K}, \left\lfloor \frac{s - \mathrm{m}}{\mathrm{d}} \right\rfloor + 1\Big), \quad L(s) = l_i \tag{4.2}$$

4.4 LMM Backbone

We choose PaliGemma 2 [39] as our backbone, and we select its 3B-parameter version because it strikes an effective balance between computational efficiency and accuracy. PaliGemma 2 integrates a pre-trained SigLIP-So400m [64] vision encoder with a Gemma 2 [42] language model using a single linear layer as the multi-modal projector. The model family comes in three parameter sizes (3B, 10B, and 28B), each offered at three different resolution-enhanced versions (224×224, 448×448, and 896×896). This creates a total of nine different model variants, all of which have been pre-trained on extensive collections of vision-language datasets. Google also provides "mix" variants that undergo specialized fine-tuning on 27 additional tasks during the final fine-tuning stage. These tasks encompass diverse capabilities including image captioning, visual question answering, reasoning, and other multimodal applications. For our experiments, we use the *paligemma2-3b-mix-448* checkpoint, which has undergone multi-task fine-tuning and therefore equips PaliGemma 2 with extensive visual understanding capabilities. In PaliGemma 2, video embeddings and text embeddings are concatenated so that the text embeddings immediately follow the video embeddings (for example: <video> How

doi:10.6342/NTU202501694

would you rate the quality of this video?). The combined embeddings are then passed to the LLM component, which is Gemma 2.

4.5 Prediction Score Calculation

After inputing the sampled video frames together with the rating prompt, the LMM outputs a logit vector

$$\mathcal{X} \in \mathbb{R}^{|VOC|},$$

where VOC denotes the full tokenizer vocabulary. To restrict our attention to the K predefined rating levels, we extract the corresponding subset of logits. Let denote the logit for rating level l_i by $\mathcal{X}_{l_i}$. Following Q-Align [56], we compute a local, closed-set softmax over just these L logits. This yields a probability distribution over the rating levels:

$$p_{l_i} = \frac{e^{(\mathcal{X}_{l_i})}}{\sum_{j=1}^{K} e^{\mathcal{X}_{l_j}}}, \qquad i = 1, \ldots, \mathrm{K}, \tag{4.3}$$

with $\sum_{i=1}^{L} p_{l_i} = 1$. Here, p_{l_i} represents the model' s confidence that the input video belongs to rating level l_i. We then assign each rating level l_i back to a continuous quality score via a mapping function $G(\cdot)$:

$$G(l_i) = m + (i-1)(\frac{\mathrm{M}-\mathrm{m}}{\mathrm{K}-1}), \tag{4.4}$$

The model's final predicted quality score $\hat{s}$ is then calculated by the expectation of these scores under the predicted distribution:

$$\hat{s} = \sum_{i=1}^{K} p_{l_i} G(l_i). \tag{4.5}$$

doi:10.6342/NTU202501694

4.6 Multi-Level Rating Framework

Current LMM-based BVQA methods [21, 52, 56, 57] map numerical MOS values to discrete rating levels, yet they are all confined to a five-level scale. However, most VQA benchmarks exhibit tight clustering around middle-range scores [16, 38, 48, 62] with few extremely poor or excellent videos and predominantly fair-to-good quality content. As a result, the five-level setting struggles to capture subtle quality variations, obscuring critical fine-grained information. To address this limitation, we suggest **expanding the number of rating levels** so the model can better distinguish between videos of similar, mediocre quality. We utilize OpenAI's o3-mini model [33] to generate semantically similar low-to-high quality descriptors for 10, 14, and 18-level classification schemes. Each rating level matches a single token in the PaliGemma 2 vocabulary, which allows us to analyze the probability distribution of just the next predicted token. The complete rating schemes are shown in Table 4.1. As discussed in Section 3.5.1, the largest MOS difference needed for statistical significance on a 0-100 scale is roughly 5.4. And since $100/5.4 \approx 18.5$, using at most 18 levels provides a reasonable upper bound.

4.7 Ordinal Regression Label Smoothing

During fine-tuning, we use the standard cross-entropy loss for optimization. However, in our multi-level framework, increasing the number of rating levels reduces the number of samples per level and thus heightens the risk of overfitting. To mitigate this, we apply label smoothing during training. Since the rating levels form an ordered set, the target distribution should assign higher probabilities to levels nearer the ground-truth

doi:10.6342/NTU202501694

Table 4.1: Mapping of rating levels to descriptors for different classification schemes

Level	5-level	10-level	14-level	18-level
1	Bad	Terrible	Abysmal	Abysmal
2	Poor	Dismal	Atrocious	Atrocious
3	Fair	Poor	Terrible	Horrendous
4	Good	Mediocre	Dismal	Appalling
5	Excellent	Fair	Poor	Wretched
6	—	Decent	Mediocre	Dreadful
7	—	Good	Average	Terrible
8	—	Great	Fair	Poor
9	—	Superb	Decent	Substandard
10	—	Excellent	Good	Mediocre
11	—	—	Great	Average
12	—	—	Superb	Fair
13	—	—	Outstanding	Decent
14	—	—	Exceptional	Good
15	—	—	—	Commendable
16	—	—	—	Impressive
17	—	—	—	Outstanding
18	—	—	—	Exceptional

label and lower probabilities as the distance grows. Accordingly, we adopt the ordinal regression label smoothing strategy proposed by [6], computing soft-label probabilities via a distance-based metric that preserves ordinal relationships. Specifically, the smoothed ground-truth probability y_i for level l_i is given by a softmax over the negative distances between the true level l_t and each candidate level in the rating level set $\mathcal{L}$:

$$y_i = \frac{e^{-\phi(t,i)}}{\sum_{k=1}^{K} e^{-\phi(t,k)}} \tag{4.6}$$

To investigate the effect of the distance measure $\phi(\cdot,\cdot)$ on the resulting soft labels, we evaluate two distance functions:

$$\phi_{\text{abs}}(t,i) = |t-i|, \qquad \phi_{\text{sq}}(t,i) = \|t-i\|^2 \tag{4.7}$$

doi:10.6342/NTU202501694

The absolute distance function ϕ_{abs} yields a gently decaying probability mass around the true rating level, whereas the squared distance function ϕ_{sq} produces a sharper decaying distribution. Figure 4.2 illustrates how the ordinal regression label smoothing scheme behaves under the absolute and squared distance metrics. Absolute-distance soft labels create flatter distributions by spreading probability more broadly across neighboring levels, which results in more label smoothing and thus stronger regularization compared to squared-distance labels. For instance, in a 14-level setting with ground-truth label 7, the Shannon entropy is approximately 1.62 for the absolute-distance soft label distribution and 1.07 for the squared-distance distribution. This indicates that absolute-distance label-smoothing introduces greater uncertainty, leading to more aggressive regularization.

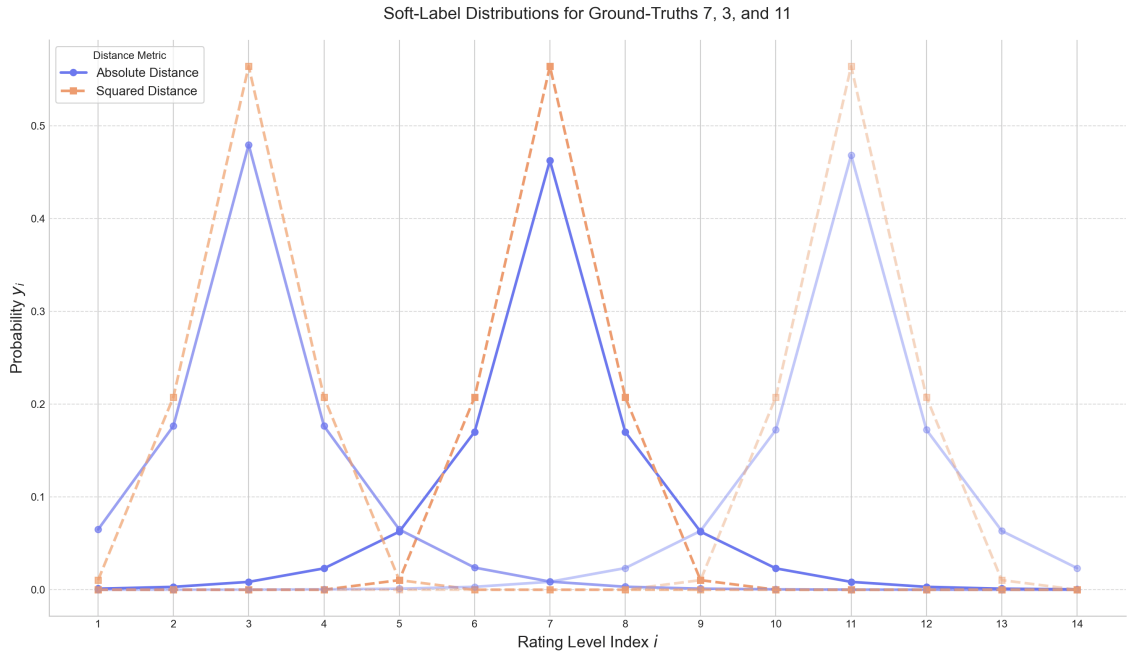


Figure 4.2: Probability distributions for ordinal regression label smoothing on a 14-level classification scheme (true index = 3, 7 and 11). The absolute-distance metric (blue) yields a gently decaying probability mass around the true level, while the squared-distance metric (orange) concentrates probability more sharply—producing a faster decay for more distant levels.

doi:10.6342/NTU202501694

doi:10.6342/NTU202501694

Chapter 5 Experiments

5.1 Fine-tuning Settings

We fine-tune our model using the LSVQ dataset's official training split [62], containing 28,056 videos. Specifically, we set the minimum score in the dataset $m = 1$ rather than $m = 0$, since zero-valued MOS scores do not occur in practice and this adjustment tightens the rating-level interval. Training is conducted on a single NVIDIA H100 SXM5 GPU using LoRA fine-tuning [18] with rank 32. We set the learning rate to 5e-4, apply a weight decay of 1e-2, and use a batch size of 1 with gradient accumulation of 64 steps. For learning rate scheduling, we adopt a cosine annealing with a 3% linear warmup. The model quantizes MOS values into 18 quality tokens using an 18-level multi-level framework. We apply ordinal regression label smoothing [6] based on absolute distance metrics and optimize using cross entropy loss. The model is fine-tuned for 2 epochs to ensure full convergence.

Alg. 1 outlines the entire Pali-VQA fine-tuning protocol. Our fine-tuning protocol operates over N_{epochs} epochs. At each step, we sample up to four frames from the input clip at a fixed rate of 0.5 frames per second and preprocess them to 448×448 resolution. We then compute an ordinal soft-label vector $\mathbf{y} \in \mathbb{R}^{\mathrm{K}}$ by measuring the distance between the normalized ground-truth level and each rating index using a smoothing function ϕ.

doi:10.6342/NTU202501694

The model produces a corresponding probability vector **p** via a softmax over the logits of rating levels. We fine-tune with a standard cross-entropy loss, back-propagating gradients through $\mathcal{M}$ and updating via an optimizer whose learning rate follows a cosine annealing schedule.

doi:10.6342/NTU202501694

Algorithm 1 Pali–VQA Fine-Tuning

Require: Dataset $\mathcal{D} = \{(v_j, s_j)\}_{j=1}^{N}$, LMM model $\mathcal{M}$, prompt $\mathcal{P}$, rating-level set $\mathcal{L}$

Require: Hyperparameters: levels K, score range $[\mathrm{m}, \mathrm{M}]$, smoothing function ϕ, initial learning rate η_0, warmup rate $\omega = 0.03$, total steps T, minimum learning rate η_{min}, number of epochs N_{epochs}

INITIALIZE optimizer with LR η_0
INITIALIZE cosine annealing scheduler with warmup rate ω, total steps T, min LR η_{min}
for epoch = 1 to N_{epochs} **do**
    for each (v_j, s_j) in $\mathcal{D}$ **do**
        FINETUNESTEP(v_j, s_j)
    end for
end for
procedure FINETUNESTEP(v, s)
    $F \leftarrow \text{SAMPLEFRAMES}(v;\ \text{rate} = 0.5,\ \text{max} = 4)$
    $F_{\text{proc}} \leftarrow \{\text{PREPROCESS}(f;\ 448 \times 448) \mid f \in F\}$
    $\mathrm{d} \leftarrow (\mathrm{M} - \mathrm{m})/\mathrm{K}$
    $t \leftarrow \min\big(\mathrm{K}, \lfloor (s - \mathrm{m})/\mathrm{d} \rfloor + 1\big)$
    for $i = 1$ **to** K **do**
        $y_i \leftarrow e^{-\phi(t,i)} \big/ \sum_{k=1}^{\mathrm{K}} e^{-\phi(t,k)}$ ▷ ground-truth soft-label probability for level i
    end for
    $\mathcal{X} \leftarrow \mathcal{M}(F_{\text{proc}}, \mathcal{P})$
    for $i = 1$ **to** K **do**
        $p_i \leftarrow e^{-\mathcal{X}_{l_i}} \big/ \sum_{j=1}^{\mathrm{K}} e^{-\mathcal{X}_{l_j}}$ ▷ predicted probability for level i
    end for
    $\mathcal{L}oss \leftarrow \text{CROSSENTROPY}(\mathbf{p}, \mathbf{y})$
    $\mathcal{L}oss$.backward()
    OPTIMIZERSTEP()
    SCHEDULERSTEP() ▷ Update LR by cosine annealing with warmup ω
end procedure

doi:10.6342/NTU202501694

5.2 Evaluation Benchmarks

For evaluation, we conduct extensive experiments on four in-the-wild video-quality benchmarks, including LSVQtest, LSVQ1080p, KoNViD-1k, and LIVE-VQC. As described in Section 5.1, the minimum score m used at inference for both LSVQtest and LSVQ1080p is also set to one instead of zero. Importantly, this change affects only the inference parameters and doesn't change the original ground-truth MOS. Benchmark metadata were obtained from the VQA² GitHub repository [21, 35] to ensure a fair comparison. The LSVQ testing split is divided into two disjoint subsets: LSVQtest and LSVQ1080p. LSVQtest contains 7,182 videos with resolutions ranging from 240p to 720p, while LSVQ1080p contains 2,575 videos at 1080p. The KoNViD-1k and LIVE-VQC benchmarks the model's cross-dataset generalization capability, as their videos were collected under acquisition conditions different from those of the training data.

5.3 Evaluation Metrics

To quantify model's performance, we report two widely used measures in VQA benchmarks.

5.3.1 Pearson Linear Correlation Coefficient (PLCC)

PLCC evaluates the linear relationship between the model's predicted scores and the ground-truth MOS. Its ranges from -1 to 1, where values closer to 1 indicate stronger

doi:10.6342/NTU202501694

positive linear agreement. The following equation defines the PLCC:

$$\text{PLCC} = \frac{\sum_{i=1}^{N}\left(\hat{s}_i - \bar{\hat{s}}\right)\left(s_i - \bar{s}\right)}{\sqrt{\sum_{i=1}^{N}\left(\hat{s}_i - \bar{\hat{s}}\right)^2}\sqrt{\sum_{i=1}^{N}\left(s_i - \bar{s}\right)^2}}, \tag{5.1}$$

where N is the number of samples; $\hat{s}_i$ and s_i are the predicted and ground-truth quality scores for the i-th sample; $\bar{\hat{s}}$ and $\bar{s}$ denote the mean predicted and mean ground-truth scores, respectively.

5.3.2 Spearman's Rank Order Correlation Coefficient (SRCC)

SRCC measures the monotonic relationship between the predicted scores and the ground-truth MOS by comparing their rank orders. The SRCC also ranges from -1 to 1, with scores approaching 1 indicating stronger agreement in the relative ordering of predicted and actual video qualities. The SRCC is defined by the following equation:

$$\text{SRCC} = 1 - \frac{6\sum_{i=1}^{N}\left(R(\hat{s}_i) - R(s_i)\right)^2}{N\left(N^2 - 1\right)} \tag{5.2}$$

where N represents the number of items in the benchmark dataset, $R(\hat{s}_i)$ denotes the rank of $\hat{s}_i$ among the predicted scores $\hat{s}_1, \hat{s}_2, \ldots, \hat{s}_n$, and $R(s_i)$ represents the corresponding rank of the ground-truth score s_i among $s_1, s_2, \ldots, s_n$.

Together, PLCC and SRCC give complementary views of model performance, quantifying both the correlation between predicted and subjective scores and the reliability of the resulting quality rankings.

doi:10.6342/NTU202501694

5.4 Inference Setting

During inference, we sample frames at 1 fps. Since all benchmark videos are shorter than 20 seconds, we apply no upper limit on sampled frames, and each clip contains at most 20 frames. However, due to computational cost (albeit lower than during fine-tuning), we still cannot process every frame in a single pass. To fully leverage the Pali-VQA model's capabilities while preserving efficiency, we perform dual predictions on each video during inference. First, video frames are sampled at 1 fps and partitioned into two disjoint sets based on frame indices (even and odd). The model generates separate quality scores $\hat{s}_i^{first}$ and $\hat{s}_i^{second}$ for each subset, and the final predicted score $\hat{s}_i$ is computed by averaging these two scores. Figure 5.1 illustrates this method.

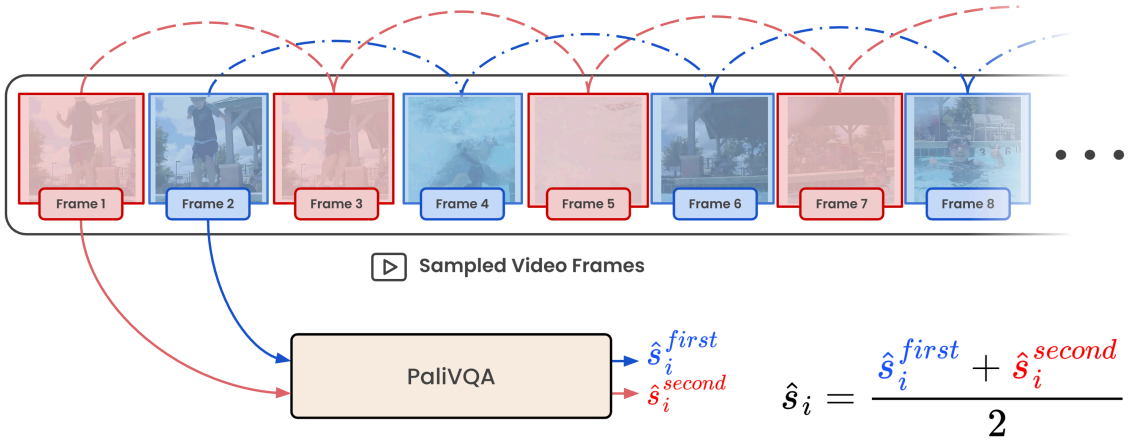
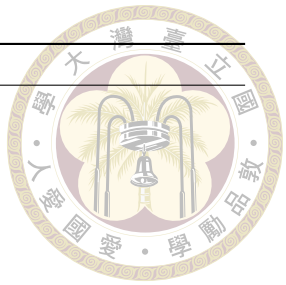


Figure 5.1: Dual inference strategy for PaliVQA. At test time, video frames are sampled at 1 fps and partitioned into two disjoint sets based on frame indices (even vs. odd). Each subset is independently processed by PaliVQA, producing quality scores. The final prediction is computed by averaging both scores.

Alg. 2 presents the steps for estimating a single video's MOS. We begin by dividing the sampled frames into even-indexed and odd-indexed subsets, calculate the expected MOS separately for each subset, and finally take their average.

doi:10.6342/NTU202501694



Algorithm 2 Pali–VQA Single-Video Inference

Require: Video v, LMM model $\mathcal{M}$, prompt $\mathcal{P}$, rating-level set $\mathcal{L}$

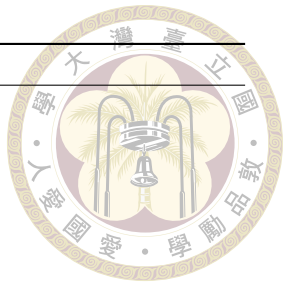
Require: Hyperparameters: levels K, score range $[\mathrm{m}, \mathrm{M}]$

Ensure: Predicted MOS $\hat{s}$

procedure SetScore($F, \mathcal{M}, \mathcal{P}$)
  $F_{\text{proc}} \leftarrow \{\textsc{Preprocess}(f;\ 448 \times 448) \mid f \in F\}$
  $\mathcal{X} \leftarrow \mathcal{M}(F_{\text{proc}}, \mathcal{P})$
  for $i = 1$ **to** K **do**
    $p_i \leftarrow e^{-\mathcal{X}_{l_i}} \big/ \sum_{j=1}^{\mathrm{K}} e^{-\mathcal{X}_{l_j}}$ ▷ predicted probability for level i
  end for
  $G(l_i) \leftarrow \mathrm{m} + (i-1)\big((\mathrm{M}-\mathrm{m})/(\mathrm{K}-1)\big)$
  $\hat{s} \leftarrow \sum_{i=1}^{\mathrm{K}} p_i\, G(l_i)$ ▷ expected quality score
  return $\hat{s}$
end procedure
procedure PredictQuality($v, \mathcal{M}, \mathcal{P}$)
  $F \leftarrow \textsc{SampleFrames}(v;\ \text{rate} = 1,\ \text{max\_frames} = \text{None})$
  $(F^{\text{even}}, F^{\text{odd}}) \leftarrow \textsc{PartitionByIndex}(F)$
  $\hat{s}_{\text{even}} \leftarrow \textsc{SetScore}(F^{\text{even}}, \mathcal{M}, \mathcal{P})$ ▷ score from even-indexed frames
  $\hat{s}_{\text{odd}} \leftarrow \textsc{SetScore}(F^{\text{odd}}, \mathcal{M}, \mathcal{P})$ ▷ score from odd-indexed frames
  $\hat{s} \leftarrow (\hat{s}_{\text{even}} + \hat{s}_{\text{odd}})/2$ ▷ final predicted MOS
  return $\hat{s}$
end procedure

Algorithm 3 presents the evaluation loop for measuring model performance over the full dataset using PLCC and SRCC. First, we collect predictions via the inference procedure (Alg. 2), and then compute both correlation metrics.

doi:10.6342/NTU202501694



Algorithm 3 Dataset-Level Evaluation (PLCC and SRCC)

Require: Dataset $\mathcal{D} = \{(v_j, s_j)\}_{j=1}^{N}$, LMM model $\mathcal{M}$, prompt $\mathcal{P}$

Ensure: Pearson correlation PLCC and Spearman correlation SRCC

procedure EVALUATEDATASET($\mathcal{D}, \mathcal{M}, \mathcal{P}$)
  Initialize empty vectors $\hat{\mathbf{s}}$ ▷ vector to collect predicted scores
  Initialize empty vectors $\mathbf{s}$ ▷ vector to collect ground-truth scores
  for each (v_j, s_j) in $\mathcal{D}$ **do**
    $\hat{s}_j \leftarrow$ PREDICTQUALITY$(v_j, \mathcal{M}, \mathcal{P})$ ▷ see Alg. 2
    $\hat{\mathbf{s}}$.append$(\hat{s}_j)$ ▷ append predicted score
    $\mathbf{s}$.append(s_j) ▷ append ground-truth score
  end for
  PLCC $\leftarrow$ PEARSONCORR$(\hat{\mathbf{s}}, \mathbf{s})$
  SRCC $\leftarrow$ SPEARMANCORR$(\hat{\mathbf{s}}, \mathbf{s})$
  return (PLCC, SRCC)
end procedure

5.5 Results

As summarized in Tables 5.1 and 5.2, we evaluate the performance of Pali-VQA across four benchmarks, reporting results for SRCC, PLCC and average metrics. Additionally, to explore the potential of ensemble methods, we combine the predictions from Pali-VQA with those from FAST-VQA by averaging their output MOS values and calculating the resulting SRCC and PLCC values. We compare our two proposed models against several existing methods, including the knowledge-based BVQA approach VIIDEO [31], the non-LMM deep BVQA methods FAST-VQA [53], Dover [55], and ModularBVQA [51], as well as the LMM-based methods Q-Align [56] and VQA² [21].

doi:10.6342/NTU202501694

Table 5.1: VQA performance on the LSVQ-1080p (2,575 videos) and LSVQ-test (7,182 videos) benchmarks. Each benchmark is evaluated using SRCC, PLCC, and their average (Avg). Rankings are highlighted by formatting: **bold + underline** marks the top performer, **bold** alone denotes second place, and underline marks the third. The "Ensemble" column flags methods that use model ensembling, and "Train sets" specifies the datasets used during training/fine-tuning.

Model	Ensemble	LSVQ-1080p			LSVQ-test			Train sets
		SRCC	PLCC	Avg	SRCC	PLCC	Avg	
Knowledge-based BVQA methods								
VIIDEO [31]	No	-0.025	-0.065	-0.045	-0.059	-0.041	-0.05	N/A
Deep BVQA methods								
FAST-VQA [53]	No	0.765	0.810	0.788	0.874	0.878	0.876	LSVQ
Minimalist-VQA [41]	Yes	0.769	0.818	0.794	0.880	0.872	0.876	LSVQ
DOVER [55]	Yes	**0.787**	0.828	0.808	0.888	0.886	0.887	LSVQ AVA
Modular-VQA [51]	Yes	**0.791**	0.844	**0.818**	0.894	**0.891**	**0.893**	LSVQ
LMM-based BVQA methods								
Q-Align (7B) [56]	No	0.758	0.833	0.796	0.883	0.882	0.883	LSVQ
One-Align (7B) [56]	No	0.761	0.822	0.792	0.886	0.884	0.885	KonIQ-10k SPAQ KADID-10k LSVQ AVA
VQA^2-scorer (7B) [21]	Yes	0.782	**0.847**	**0.815**	**0.897**	0.885	0.891	KonIQ-10k KADID-10k LSVQ LIVE-Qualcomm Waterloo-I/III LIVE-NFLX-II
Pali-VQA (3B) (Ours)	No	0.761	0.842	0.802	0.888	0.889	0.889	LSVQ
Pali-VQA (3B) + FAST-VQA (Ours)	Yes	0.766	**0.846**	0.806	**0.901**	**0.901**	**0.901**	LSVQ

doi:10.6342/NTU202501694

Table 5.2: VQA performance on out-of-distribution benchmarks: LIVE-VQC (88 videos) and KoNViD-1k (1200 videos). Each benchmark is evaluated using SRCC, PLCC, and their average (Avg). Rankings are highlighted by formatting: **bold + underline** marks the top performer, **bold** alone denotes second place, and underline marks the third. The "Ensemble" column flags methods that use model ensembling, and "Train sets" specifies the datasets used during training/fine-tuning.

Model	Ensemble	LIVE-VQC			KoNViD-1k			Train sets
		SRCC	PLCC	Avg	SRCC	PLCC	Avg	
Knowledge-based BVQA methods								
VIIDEO [31]	No	-0.287	-0.226	-0.257	0.221	0.240	0.231	N/A
Deep BVQA methods								
FAST-VQA [53]	No	0.769	0.815	0.792	0.859	0.857	0.858	LSVQ
Minimalist-VQA [41]	Yes	0.765	0.812	0.789	0.859	0.861	0.860	LSVQ
DOVER [55]	Yes	0.771	0.819	0.795	0.890	0.883	0.887	LSVQ AVA
Modular-VQA [51]	Yes	0.783	0.825	0.804	0.878	**0.884**	0.881	LSVQ
LMM-based BVQA methods								
Q-Align (7B) [56]	No	0.777	0.813	0.795	0.865	0.876	0.871	LSVQ
One-Align (7B) [56]	No	0.766	0.826	0.796	0.876	0.878	0.877	KonIQ-10k SPAQ KADID-10k LSVQ AVA
VQA^2-scorer (7B) [21]	Yes	**0.785**	**0.830**	**0.808**	**0.894**	**0.884**	**0.889**	KonIQ-10k KADID-10k LSVQ LIVE-Qualcomm Waterloo-I/III LIVE-NFLX-II
Pali-VQA (3B) (Ours)	No	0.773	**0.830**	0.802	0.878	0.882	0.880	LSVQ
Pali-VQA (3B) + FAST-VQA (Ours)	Yes	**0.859**	**0.897**	**0.878**	**0.897**	**0.894**	**0.896**	LSVQ

doi:10.6342/NTU202501694

5.6 Discussion

Our Pali-VQA ensemble model, which combines Pali-VQA with FAST-VQA, demonstrates state-of-the-art performance, outperforming existing methods on both SRCC and PLCC metrics across all benchmarks except for LSVQ1080p. The Pali-VQA ensemble model delivers outstanding results especially on LIVE-VQC, achieving an average score of 0.878, computed as (PLCC+SRCC)/2. This result surpasses Q-Align by 10.44% and VQA^2 by 8.7%, positioning our model as the leading LMM-based method. As described in Section 3.3, the LIVE-VQC dataset comprises videos captured by 101 different devices, suggesting our model generalizes well across diverse capture conditions. Furthermore, the standalone Pali-VQA model proves highly effective. It outperforms the larger 7B Q-Align and One-Align models on all benchmarks in terms of the average score, highlighting the efficacy of our proposed multi-level rating scheme. Importantly, our models employ LoRA fine-tuning, adjusting fewer than 3% of their parameters, whereas Q-Align and VQA^2 both require full fine-tuning. Furthermore, our model was fine-tuned exclusively on the LSVQ dataset, without additional IQA/IAA datasets, instruction-tuning sets, complex training strategies, or diverse prompts. In contrast, VQA^2 was fine-tuned on six different datasets.

5.7 Ablation Study

We conduct a detailed ablation study to examine how varying the number of rating levels and applying ordinal regression label smoothing affect the performance of Pali-VQA. To ensure reliability and fairness, we test several configurations. Apart from chang-

doi:10.6342/NTU202501694

ing the learning rate, label-smoothing scheme, and number of rating levels, all models are fine-tuned using LoRA (rank = 32, weight decay = 0.01, batch size = 64) for up to two epochs. If a model converges after the first epoch, we report its epoch-1 results.

5.7.1 Multi-level Rating Scheme

To investigate the influence of our multi-level rating scheme, we perform an ablation study by training models with 18, 14, 10, and 5 rating levels. Table 5.3 presents the experimental results. For each configuration, we report the average SRCC and PLCC across four standard benchmarks (LSVQ1080p, LSVQtest, LIVE-VQC, and KoNViD-1k), as well as the overall mean performance for each level.

Our results highlight three main observations. First, the model reaches its peak average performance with **14 levels** (SRCC = 0.827, PLCC = 0.857) and **18 levels** (SRCC = 0.824, PLCC = 0.859). Second, the average SRCC values for 18, 14, and 10 levels (0.824, 0.827, and 0.821) lie within a narrow margin (Δ SRCC ≈ 0.006), suggesting that adding more than ten levels yields diminishing returns. Third, adopting the standard 5-level setup reduces SRCC to 0.803, a decrease of over 0.024 compared to the 14-level model. This performance drop in SRCC underscores that an adequate number of rating levels is crucial for the model to discriminate between videos of similar quality.

To further explore the effect of our multi-level schemes, we assess the classification performance of the best-performing configuration for each multi-level setting, as determined by the PLCC and SRCC results in Table 5.3. We first convert the continuous ground-truth MOS scores into five quality classes by uniformly partitioning the score range. For example, for the LSVQ benchmark, scores 0-20 represent the first class, 21-

doi:10.6342/NTU202501694

Table 5.3: Ablation study on the number of rating levels. For each level, we show the performance of individual training configurations, averaged across four benchmarks (LSVQ-1080p, LSVQ-test, LIVE-VQC, KoNViD-1k). The **Overall Avg.** for each level block represents the mean performance across all its configurations.

Level	Epoch	Smoothing	LR	Avg SRCC	Avg PLCC
18	2	Absolute	5.00E-04	0.825	0.861
	2	Squared	3.00E-04	0.824	0.859
	2	None (x)	5.00E-04	0.822	0.858
	2	None (x)	3.00E-04	0.824	0.856
	Overall Avg. (18)			**0.824**	**0.859**
14	2	Absolute	5.00E-04	0.828	0.859
	2	Squared	3.00E-04	0.828	0.857
	2	None (x)	5.00E-04	0.826	0.855
	2	None (x)	3.00E-04	0.826	0.855
	Overall Avg. (14)			**0.827**	**0.857**
10	2	Absolute	5.00E-04	0.822	0.855
	2	Squared	3.00E-04	0.822	0.857
	2	None (x)	5.00E-04	0.821	0.855
	2	None (x)	3.00E-04	0.820	0.856
	Overall Avg. (10)			**0.821**	**0.856**
5	2	Absolute	5.00E-04	0.815	0.847
	1	Squared	3.00E-04	0.796	0.837
	1	None (x)	5.00E-04	0.798	0.845
	1	None (x)	3.00E-04	0.803	0.845
	Overall Avg. (5)			**0.803**	**0.844**

40 the second, and so forth. We then measure classification accuracy by determining if a model's predicted score falls into the same class as the ground-truth MOS. The accuracy for each class, shown in Table 5.4, represents an aggregate measure of performance across multiple datasets. For each model configuration, we first calculate its classification accuracy on four distinct benchmarks: LSVQ-1080p, LSVQ-test, LIVE-VQC, and KoNViD-1k. The final value reported in the table is the unweighted average of these four accuracy scores. First of all, all the schemes have 0 accuracy on class 1. This is due to the scarcity of samples in both the fine-tuning and test sets. The results show that with the 5-level scheme, predictions cluster at level 4, yielding high accuracy for class 4 but

doi:10.6342/NTU202501694

inferior performance on classes 1 and 5. The 10-level scheme delivers the most balanced results, performing well across all classes except class 0. As the number of rating levels increases, the model increasingly gravitates towards mid-range scores, as seen in the declining accuracies for classes 2 and 5. Overall, multi-level schemes help distribute MOS predictions more evenly rather than concentrating them in the middle.

Table 5.4: Classification accuracy across five quality classes for different multi-level schemes and training configurations. The "Level" column indicates the multi-level scheme, "Smoothing" refers to the distance function used for label smoothing, and "LR" is the learning rate.

Level	Epoch	Smoothing	LR	Class 1	Class 2	Class 3	Class 4	Class 5
18	2	Absolute	5.00E-04	0.00	24.69	71.25	88.75	18.4
14	2	Absolute	5.00E-04	0.00	36.17	69.87	90.03	13.55
10	2	Squared	3.00E-04	0.00	40.05	66.9	91.78	31.03
5	2	Absolute	5.00E-04	0.00	18.5	61.54	93.83	0.00

5.7.2 Ordinal Regression Label Smoothing

To understand how ordinal regression label smoothing schemes affect Pali-VQA's performance in multi-level settings, we focus exclusively on configurations with 14 and 18 levels. We compare models fine-tuned under three different conditions: ordinal regression label smoothing using absolute distance metrics, squared distance metrics, and no label smoothing at all. The differences in performance among these smoothing strategies are minimal, yet absolute-distance smoothing still leads by a small margin, particularly in PLCC. Specifically, absolute-distance smoothing achieves an average PLCC of 0.859, versus 0.858 for squared-distance smoothing and 0.856 without smoothing.

doi:10.6342/NTU202501694

Table 5.5: Ablation study on the label smoothing scheme. We compare ordinal regression label smoothing using absolute distance metrics (Absolute), squared distance metrics (Squared), and a baseline with no explicit smoothing (None). For each scheme, we list the performance of every configuration, averaged across four benchmarks (LSVQ-1080p, LSVQ-test, LIVE-VQC, KoNViD-1k). The **Overall Avg.** represents the mean performance for that scheme across all its tested configurations.

Smoothing	**Level**	**LR**	**Avg SRCC**	**Avg PLCC**
Absolute	18	5.00E-04	0.825	0.861
	18	3.00E-04	0.822	0.858
	14	5.00E-04	0.828	0.859
	14	3.00E-04	0.829	0.860
	Overall Avg. (Absolute)		**0.826**	**0.859**
Squared	18	5.00E-04	0.825	0.858
	18	3.00E-04	0.824	0.859
	14	5.00E-04	0.828	0.856
	14	3.00E-04	0.828	0.857
	Overall Avg. (Squared)		**0.826**	**0.858**
None	18	5.00E-04	0.822	0.858
	18	3.00E-04	0.824	0.856
	14	5.00E-04	0.826	0.855
	14	3.00E-04	0.826	0.855
	Overall Avg. (None)		**0.825**	**0.856**

doi:10.6342/NTU202501694

doi:10.6342/NTU202501694

Chapter 6 Conclusion

We introduce **Pali-VQA**, an LMM-based BVQA model that leverages multi-level classification with 5, 10, 14, and 18 rating levels to enhance video quality assessment accuracy. By fine-tuning the *paligemma2-3b-mix-448* checkpoint of PaliGemma 2 via LoRA and employing a dual-pass inference scheme, we strike an optimal balance between computational efficiency and prediction performance. Our method achieves competitive results compared to other 7B-parameter, fully fine-tuned alternatives while requiring fewer computational resources. We further apply ordinal regression label smoothing to regularize our model while preserving the ordinal relationships among quality levels.

Our experiments demonstrate the efficacy of our method. The standalone Pali-VQA outperforms larger 7B models like Q-Align and One-Align on several in-the-wild VQA benchmarks. Furthermore, our ensemble model surpasses leading methods on multiple benchmarks, all without relying on massive model sizes, complex training recipes, or auxiliary visual instruction dataset.

doi:10.6342/NTU202501694

doi:10.6342/NTU202501694

References

[1] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, R. Ring, E. Rutherford, S. Cabi, T. Han, Z. Gong, S. Samangooei, M. Monteiro, J. Menick, S. Borgeaud, A. Brock, A. Nematzadeh, S. Sharifzadeh, M. Binkowski, R. Barreira, O. Vinyals, A. Zisserman, and K. Simonyan. Flamingo: a visual language model for few-shot learning. arXiv preprint arXiv:2204.14198, 2022.

[2] C. G. Bampis, Z. Li, I. Katsavounidis, T.-Y. Huang, C. Ekanadham, and A. C. Bovik. Towards perceptually optimized adaptive video streaming-a realistic quality of experience database. IEEE Transactions on Image Processing, 30:5182–5197, 2021.

[3] L. Beyer, A. Steiner, A. S. Pinto, A. Kolesnikov, X. Wang, D. Salz, M. Neumann, I. Alabdulmohsin, M. Tschannen, E. Bugliarello, T. Unterthiner, D. Keysers, S. Koppula, F. Liu, A. Grycner, A. Gritsenko, N. Houlsby, M. Kumar, K. Rong, J. Eisenschlos, R. Kabra, M. Bauer, M. Bošnjak, X. Chen, M. Minderer, P. Voigtlaender, I. Bica, I. Balazevic, J. Puigcerver, P. Papalampidi, O. Henaff, X. Xiong, R. Soricut, J. Harmsen, and X. Zhai. Paligemma: A versatile 3b vlm for transfer. arXiv preprint arXiv:2407.07726, 2024.

[4] Z. Chen, W. Wang, Y. Cao, Y. Liu, Z. Gao, E. Cui, J. Zhu, S. Ye, H. Tian, Z. Liu,

doi:10.6342/NTU202501694

L. Gu, X. Wang, Q. Li, Y. Ren, Z. Chen, J. Luo, J. Wang, T. Jiang, B. Wang, C. He, B. Shi, X. Zhang, H. Lv, Y. Wang, W. Shao, P. Chu, Z. Tu, T. He, Z. Wu, H. Deng, J. Ge, K. Chen, K. Zhang, L. Wang, M. Dou, L. Lu, X. Zhu, T. Lu, D. Lin, Y. Qiao, J. Dai, and W. Wang. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. arXiv preprint arXiv:2412.05271, 2025.

[5] W. Dai, J. Li, D. Li, A. M. H. Tiong, J. Zhao, W. Wang, B. Li, P. Fung, and S. Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. arXiv preprint arXiv:2305.06500, 2023.

[6] R. Diaz and A. Marathe. Soft labels for ordinal regression. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 4738–4747, June 2019.

[7] Z. Duanmu, A. Rehman, and Z. Wang. A quality-of-experience database for adaptive video streaming. IEEE Transactions on Broadcasting, 64(2):474–487, 2018.

[8] Z. Duanmu, K. Zeng, K. Ma, A. Rehman, and Z. Wang. A quality-of-experience index for streaming video. IEEE Journal of Selected Topics in Signal Processing, 11(1):154–166, 2017.

[9] C. Feichtenhofer, H. Fan, J. Malik, and K. He. Slowfast networks for video recognition. arXiv preprint arXiv:1812.03982, 2019.

[10] D. Ghadiyaram, J. Pan, A. C. Bovik, A. K. Moorthy, P. Panda, and K.-C. Yang. In-capture mobile video distortions: A study of subjective behavior and objective algorithms. IEEE Transactions on Circuits and Systems for Video Technology, 28(9):2061–2077, 2018.

doi:10.6342/NTU202501694

[11] S. Halawani, I. Albidewi, and A. Ahmad. A novel ensemble method for regression via classification problems. Journal of Computer Science, 7(3):387–393, 2011.

[12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385, 2015.

[13] X. He, D. Jiang, G. Zhang, M. Ku, A. Soni, S. Siu, H. Chen, A. Chandra, Z. Jiang, A. Arulraj, K. Wang, Q. D. Do, Y. Ni, B. Lyu, Y. Narsupalli, R. Fan, Z. Lyu, Y. Lin, and W. Chen. Videoscore: Building automatic metrics to simulate fine-grained human feedback for video generation. arXiv preprint arXiv:2406.15252, 2024.

[14] S. Horiguchi, K. Dohi, and Y. Kawaguchi. Streaming active learning for regression problems using regression via classification. arXiv preprint arXiv:2309.01013, 2023.

[15] V. Hosu, F. Hahn, M. Jenadeleh, H. Lin, H. Men, T. Szirányi, S. Li, and D. Saupe. The konstanz natural video database. https://database.mmsp-kn.de, 2017. Accessed: 2025-07-10.

[16] V. Hosu, F. Hahn, M. Jenadeleh, H. Lin, H. Men, T. Szirányi, S. Li, and D. Saupe. The konstanz natural video database (konvid-1k). In 2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX), pages 1–6, 2017.

[17] V. Hosu, H. Lin, T. Sziranyi, and D. Saupe. Koniq-10k: An ecologically valid database for deep learning of blind image quality assessment. IEEE Transactions on Image Processing, 29:4041–4056, 2020.

[18] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685, 2021.

doi:10.6342/NTU202501694

[19] Internet Archive. Moving image archive. https://archive.org/details/movies, n.d. Accessed: 2025-08-01.

[20] ITU-T. Objective Perceptual Assessment of Video Quality: Full-Reference Television. Tutorial, ITU-T Telecommunication Standardization Bureau, 2004.

[21] Z. Jia, Z. Zhang, J. Qian, H. Wu, W. Sun, C. Li, X. Liu, W. Lin, G. Zhai, and X. Min. VQA^2: Visual question answering for video quality assessment. arXiv preprint arXiv:2411.03795, 2024.

[22] P. Kancharla and S. S. Channappayya. Completely blind quality assessment of user generated video content. IEEE Transactions on Image Processing, 31:263–274, 2022.

[23] T. Kou, X. Liu, W. Sun, J. Jia, X. Min, G. Zhai, and N. Liu. Stablevqa: A deep no-reference quality assessment model for video stability. In Proceedings of the 31st ACM International Conference on Multimedia, MM '23, pages 1066–1076. ACM, Oct. 2023.

[24] B. Li, W. Zhang, M. Tian, G. Zhai, and X. Wang. Blindly assess quality of in-the-wild videos via quality-aware pre-training and motion perception. arXiv preprint arXiv:2108.08505, 2022.

[25] B. Li, Y. Zhang, D. Guo, R. Zhang, F. Li, H. Zhang, K. Zhang, P. Zhang, Y. Li, Z. Liu, and C. Li. Llava-onevision: Easy visual task transfer. arXiv preprint arXiv:2408.03326, 2024.

[26] D. Li, T. Jiang, and M. Jiang. Quality assessment of in-the-wild videos. In Proceedings of the 27th ACM International Conference on Multimedia, MM '19, pages 2351–2359. ACM, Oct. 2019.

doi:10.6342/NTU202501694

[27] H. Lin, V. Hosu, and D. Saupe. Kadid-10k: A large-scale artificially distorted iqa database. In 2019 Eleventh International Conference on Quality of Multimedia Experience (QoMEX), pages 1–3, 2019.

[28] H. Liu, C. Li, Q. Wu, and Y. J. Lee. Visual instruction tuning. arXiv preprint arXiv:2304.08485, 2023.

[29] M. Maaz, H. Rasheed, S. Khan, and F. S. Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. arXiv preprint arXiv:2306.05424, 2024.

[30] X. Min, H. Duan, W. Sun, Y. Zhu, and G. Zhai. Perceptual video quality assessment: A survey. arXiv preprint arXiv:2402.03413, 2024.

[31] A. Mittal, M. A. Saad, and A. C. Bovik. A completely blind video integrity oracle. Trans. Img. Proc., 25(1):289–300, Jan. 2016.

[32] A. Mittal, R. Soundararajan, and A. C. Bovik. Making a "completely blind" image quality analyzer. IEEE Signal Processing Letters, 20(3):209–212, 2013.

[33] OpenAI. Openai o3mini. https://openai.com/index/openai-o3-mini/, Jan 2025. Accessed: 2025-07-10.

[34] Y. Pei, S. Huang, Y. Lu, X. Li, and Z. Chen. Priorformer: A ugc-vqa method with content and distortion priors. arXiv preprint arXiv:2406.16297, 2024.

[35] Q-Future. Visual-question-answering-for-video-quality-assessment. https://github.com/Q-Future/Visual-Question-Answering-for-Video-Quality-Assessment, 2025. Accessed: 2025-07-09.

doi:10.6342/NTU202501694

[36] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision. arXiv preprint arXiv:2103.00020, 2021.

[37] K. Seshadrinathan, R. Soundararajan, A. C. Bovik, and L. K. Cormack. Study of subjective and objective quality assessment of video. IEEE Transactions on Image Processing, 19(6):1427–1441, 2010.

[38] Z. Sinno and A. C. Bovik. Large-scale study of perceptual video quality. IEEE Transactions on Image Processing, 28(2):612–627, Feb. 2019.

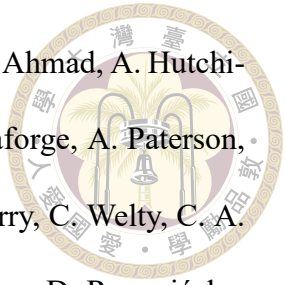
[39] A. Steiner, A. S. Pinto, M. Tschannen, D. Keysers, X. Wang, Y. Bitton, A. Gritsenko, M. Minderer, A. Sherbondy, S. Long, S. Qin, R. Ingle, E. Bugliarello, S. Kazemzadeh, T. Mesnard, I. Alabdulmohsin, L. Beyer, and X. Zhai. Paligemma 2: A family of versatile vlms for transfer. arXiv preprint arXiv:2412.03555, 2024.

[40] W. Sun, X. Min, W. Lu, and G. Zhai. A deep learning based no-reference quality assessment model for ugc videos. In Proceedings of the 30th ACM International Conference on Multimedia, MM ’ 22, pages 856–865. ACM, Oct. 2022.

[41] W. Sun, W. Wen, X. Min, L. Lan, G. Zhai, and K. Ma. Analysis of video quality datasets via design of minimalistic video quality models. arXiv preprint arXiv:2307.13981, 2024.

[42] G. Team, M. Riviere, S. Pathak, P. G. Sessa, C. Hardin, S. Bhupatiraju, L. Hussenot, T. Mesnard, B. Shahriari, A. Ramé, J. Ferret, P. Liu, P. Tafti, A. Friesen, M. Casbon, S. Ramos, R. Kumar, C. L. Lan, S. Jerome, A. Tsitsulin, N. Vieillard, P. Stanczyk, S. Girgin, N. Momchev, M. Hoffman, S. Thakoor, J.-B. Grill,

doi:10.6342/NTU202501694



B. Neyshabur, O. Bachem, A. Walton, A. Severyn, A. Parrish, A. Ahmad, A. Hutchison, A. Abdagic, A. Carl, A. Shen, A. Brock, A. Coenen, A. Laforge, A. Paterson, B. Bastian, B. Piot, B. Wu, B. Royal, C. Chen, C. Kumar, C. Perry, C. Welty, C. A. Choquette-Choo, D. Sinopalnikov, D. Weinberger, D. Vijaykumar, D. Rogozińska, D. Herbison, E. Bandy, E. Wang, E. Noland, E. Moreira, E. Senter, E. Eltyshev, F. Visin, G. Rasskin, G. Wei, G. Cameron, G. Martins, H. Hashemi, H. Klimczak-Plucińska, H. Batra, H. Dhand, I. Nardini, J. Mein, J. Zhou, J. Svensson, J. Stanway, J. Chan, J. P. Zhou, J. Carrasqueira, J. Iljazi, J. Becker, J. Fernandez, J. van Amersfoort, J. Gordon, J. Lipschultz, J. Newlan, J. yeong Ji, K. Mohamed, K. Badola, K. Black, K. Millican, K. McDonell, K. Nguyen, K. Sodhia, K. Greene, L. L. Sjoesund, L. Usui, L. Sifre, L. Heuermann, L. Lago, L. McNealus, L. B. Soares, L. Kilpatrick, L. Dixon, L. Martins, M. Reid, M. Singh, M. Iverson, M. Görner, M. Velloso, M. Wirth, M. Davidow, M. Miller, M. Rahtz, M. Watson, M. Risdal, M. Kazemi, M. Moynihan, M. Zhang, M. Kahng, M. Park, M. Rahman, M. Khatwani, N. Dao, N. Bardoliwalla, N. Devanathan, N. Dumai, N. Chauhan, O. Wahltinez, P. Botarda, P. Barnes, P. Barham, P. Michel, P. Jin, P. Georgiev, P. Culliton, P. Kuppala, R. Comanescu, R. Merhej, R. Jana, R. A. Rokni, R. Agarwal, R. Mullins, S. Saadat, S. M. Carthy, S. Cogan, S. Perrin, S. M. R. Arnold, S. Krause, S. Dai, S. Garg, S. Sheth, S. Ronstrom, S. Chan, T. Jordan, T. Yu, T. Eccles, T. Hennigan, T. Kocisky, T. Doshi, V. Jain, V. Yadav, V. Meshram, V. Dharmadhikari, W. Barkley, W. Wei, W. Ye, W. Han, W. Kwon, X. Xu, Z. Shen, Z. Gong, Z. Wei, V. Cotruta, P. Kirk, A. Rao, M. Giang, L. Peran, T. Warkentin, E. Collins, J. Barral, Z. Ghahramani, R. Hadsell, D. Sculley, J. Banks, A. Dragan, S. Petrov, O. Vinyals, J. Dean, D. Hassabis, K. Kavukcuoglu, C. Farabet, E. Buchatskaya, S. Borgeaud, N. Fiedel,

doi:10.6342/NTU202501694

A. Joulin, K. Kenealy, R. Dadashi, and A. Andreev. Gemma 2: Improving open language models at a practical size. arXiv preprint arXiv:2408.00118, 2024.

[43] B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L.-J. Li. Yfcc100m: The new data in multimedia research. http://projects.dfki.uni-kl.de/yfcc100m/, 2015. Accessed: 2025-08-01.

[44] L. Torgo and J. Gama. Regression by classification. In D. L. Borges and C. A. A. Kaestner, editors, Advances in Artificial Intelligence, 13th Brazilian Symposium on Artificial Intelligence (SBIA '96), Proceedings, volume 1159 of Lecture Notes in Computer Science, pages 51–60, Berlin, Heidelberg, 1996. Springer.

[45] Z. Tu, C.-J. Chen, L.-H. Chen, Y. Wang, N. Birkbeck, B. Adsumilli, and A. C. Bovik. Regression or classification? new methods to evaluate no-reference picture and video quality models. arXiv preprint arXiv:2102.00155, 2021.

[46] H. Wang, W. Gan, S. Hu, J. Y. Lin, L. Jin, L. Song, P. Wang, I. Katsavounidis, A. Aaron, and C.-C. J. Kuo. Mcl-jcv: A jnd-based h.264/avc video quality assessment dataset. In 2016 IEEE International Conference on Image Processing (ICIP), pages 1509–1513, 2016.

[47] H. Wang, I. Katsavounidis, J. Zhou, J. Park, S. Lei, X. Zhou, M.-O. Pun, X. Jin, R. Wang, X. Wang, Y. Zhang, J. Huang, S. Kwong, and C. C. J. Kuo. Videoset: A large-scale compressed video quality dataset based on jnd measurement. arXiv preprint arXiv:1701.01500, 2017.

[48] Y. Wang, S. Inguva, and B. Adsumilli. Youtube ugc dataset for video compression research. In 2019 IEEE 21st International Workshop on Multimedia Signal Processing (MMSP), pages 1–5, 2019.

doi:10.6342/NTU202501694

[49] Y. Wang, J. Ke, H. Talebi, J. G. Yim, N. Birkbeck, B. Adsumilli, P. Milanfar, and F. Yang. Rich features for perceptual quality assessment of ugc videos. In 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 13430–13439, 2021.

[50] Y. Wang and F. Yang. Uvq: Measuring youtube's perceptual video quality. https://research.google/blog/uvq-measuring-youtubes-perceptual-video-quality/, Aug. 2022. Accessed: 2025-07-09.

[51] W. Wen, M. Li, Y. Zhang, Y. Liao, J. Li, L. Zhang, and K. Ma. Modular blind video quality assessment. arXiv preprint arXiv:2402.19276, 2024.

[52] W. Wen, Y. Wang, N. Birkbeck, and B. Adsumilli. An ensemble approach to short-form video quality assessment using multimodal llm. arXiv preprint arXiv:2412.18060, 2024.

[53] H. Wu, C. Chen, J. Hou, L. Liao, A. Wang, W. Sun, Q. Yan, and W. Lin. Fast-vqa: Efficient end-to-end video quality assessment with fragment sampling. arXiv preprint arXiv:2207.02595, 2022.

[54] H. Wu, L. Liao, A. Wang, C. Chen, J. Hou, W. Sun, Q. Yan, and W. Lin. Towards robust text-prompted semantic criterion for in-the-wild video quality assessment. arXiv preprint arXiv:2304.14672, 2023.

[55] H. Wu, E. Zhang, L. Liao, C. Chen, J. H. Hou, A. Wang, W. S. Sun, Q. Yan, and W. Lin. Exploring video quality assessment on user generated contents from aesthetic and technical perspectives. In International Conference on Computer Vision (ICCV), 2023.

doi:10.6342/NTU202501694

[56] H. Wu, Z. Zhang, W. Zhang, C. Chen, L. Liao, C. Li, Y. Gao, A. Wang, E. Zhang, W. Sun, Q. Yan, X. Min, G. Zhai, and W. Lin. Q-align: Teaching lmms for visual scoring via discrete text-defined levels. arXiv preprint arXiv:2312.17090, 2023.

[57] S. Wu, Y. Li, Z. Xu, Y. Gao, H. Duan, W. Sun, and G. Zhai. Fvq: A large-scale dataset and a lmm-based method for face video quality assessment. arXiv preprint arXiv:2504.09255, 2025.

[58] F. Xing, Y.-G. Wang, H. Wang, L. Li, and G. Zhu. Starvqa: Space-time attention for video quality assessment. arXiv preprint arXiv:2108.09635, 2021.

[59] J. Xu, P. Ye, Y. Liu, and D. Doermann. No-reference video quality assessment via feature learning. In 2014 IEEE International Conference on Image Processing (ICIP), pages 491–495, 2014.

[60] Q. Ye, H. Xu, J. Ye, M. Yan, A. Hu, H. Liu, Q. Qian, J. Zhang, F. Huang, and J. Zhou. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. arXiv preprint arXiv:2311.04257, 2023.

[61] J. Yim, Y. Wang, N. A. C. Birkbeck, and B. Adsumilli. Subjective quality assessment for youtube ugc dataset. In 2020 IEEE International Conference on Image Processing, pages 131–135, 2020.

[62] Z. Ying, M. Mandal, D. Ghadiyaram, and A. Bovik. Patch-vq: 'patching up' the video quality problem. In 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 14014–14024. IEEE, June 2021.

[63] J. You and Y. Lin. Efficient transformer with locally shared attention for video quality assessment. In 2022 IEEE International Conference on Image Processing (ICIP), pages 356–360, 2022.

doi:10.6342/NTU202501694

[64] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer. Sigmoid loss for language image pre-training. arXiv preprint arXiv:2303.15343, 2023.

[65] Q. Zheng, Y. Fan, L. Huang, T. Zhu, J. Liu, Z. Hao, S. Xing, C.-J. Chen, X. Min, A. C. Bovik, and Z. Tu. Video quality assessment: A comprehensive survey. arXiv preprint arXiv:2412.04508, 2024.

doi:10.6342/NTU202501694