VGGSound*er*: Audio-Visual Evaluations for Foundation Models

Daniil Zverev[1,*] Thaddäus Wiedemer[2,3,4,*] Ameya Prabhu[2,3]
Matthias Bethge[2,3] Wieland Brendel[3,4] A. Sophia Koepke[1,2,3]

[1]Technical University of Munich, MCML [2]University of Tübingen [3]Tübingen AI Center
[4]MPI for Intelligent Systems, ELLIS Institute Tübingen

{daniil.zverev,a-sophia.koepke}@tum.de, {thaddaeus.wiedemer,ameya.prabhu}@uni-tuebingen.de

Abstract

The emergence of audio-visual foundation models underscores the importance of reliably assessing their multi-modal understanding. The VGGSound dataset is commonly used as a benchmark for evaluation audio-visual classification. However, our analysis identifies several limitations of VGGSound, including incomplete labelling, partially overlapping classes, and misaligned modalities. These lead to distorted evaluations of auditory and visual capabilities. To address these limitations, we introduce VGGSounder, a comprehensively re-annotated, multi-label test set that extends VGGSound and is specifically designed to evaluate audio-visual foundation models. VGGSounder features detailed modality annotations, enabling precise analyses of modality-specific performance. Furthermore, we reveal model limitations by analysing performance degradation when adding another input modality with our new modality confusion metric. Our dataset and project page are available at `https://vggsounder.github.io/`.

1. Introduction

Rigorous evaluation benchmarks have been instrumental in assessing the effectiveness of audio-visual models [33, 43, 49, 57]. Specifically, multi-modal foundation models integrating visual and auditory data aim to achieve a holistic understanding of audio-visual content. However, the field lacks large-scale modality-aware classification benchmarks with ground-truth annotations indicating whether each label is visible, audible, or both. Such annotations would allow detailed evaluations of multi-modal model capabilities. To address this gap, we introduce VGGSound*er*, an enhanced version of the widely-used audio-visual classification dataset VGGSound [13], which facilitates modality-aware evaluation of audio-visual foundation models.

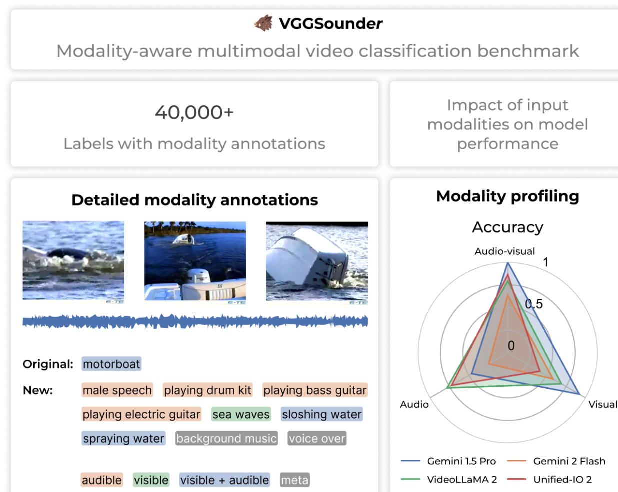


Figure 1. **We introduce VGGSound*er*, a multi-label audio-visual classification benchmark with modality annotations**. We extend the original VGGSound test set with human-annotated `audible`, `visible`, and `visible+audible` labels. We add `meta` labels for common confounders, such as background music. We benchmark eleven recent audio-visual models on VGGSound*er*. It enables selective analysis of a model's auditory and visual capabilities on classes relevant for the queried modality.

VGGSound has several notable limitations. First, its data is inherently multi-label; for instance, a single sample may simultaneously include labels such as `playing drum kit` and `playing acoustic guitar` when multiple instruments are present. Additionally, evaluating how different modalities contribute to model performance becomes difficult without explicit modality annotations, as some labels are either not visually present or not audible (e.g., certain instruments might only be audible but not visible in advertisements). Moreover, overlapping label classes present another challenge; for example,

[*]equal contribution

the `orchestra` label often coincides with labels for individual instruments. These issues result in systematic under-evaluation of multi-modal audio-visual foundation models.

To overcome these limitations, we present VGGSound*er*, an improved benchmark inspired by similar advancements in other domains [12, 28]. We re-annotate the dataset to create a comprehensive multi-label classification setting by collecting detailed annotations for each sample, including (1) additional classes present, (2) explicit modality annotations to label modality misalignment, (3) metadata indicating the presence of background music, voice-over, or static images, and (4) merging of classes to address overlapping classes. Consequently, VGGSound*er* provides a robust, foundation-model-ready benchmark enabling structured analysis of whether models rely on audio or visual cues. Furthermore, we include meta-labels (e.g., background music, voice-over, or static images) to easily filter out unreliable labels during evaluation. Utilising VGGSound*er*, we evaluate audio-visual foundation models, demonstrating their poor performance on our benchmark. We find that the state-of-the-art, closed-source Gemini models consistently rely exclusively on the visual modality. We measure that effect with the modality confusion, i.e. when models get distracted by an additional input modality, which exposes the unsuccessful merging of modalities. These findings highlight the importance of the audio-focused VGGSound*er* benchmark as a critical tool for accurately assessing audio-visual foundation models.

We make the following contributions:
1. We illustrate limitations of VGGSound in Sec. 3.
2. We curate VGGSound*er* with multi-modal human annotations for multi-label classification in Sec. 4.
3. We evaluate state-of-the-art audio-visual models, observing differences between embedding models and autoregressive foundation models in Sec. 5.
4. We propose new metrics to quantify the negative impact of using multiple input modalities in Sec. 5.

2. Related work

Audio-visual learning Many prior works consider audio-visual tasks that include sound source localisation and separation [3, 5, 9, 15, 27, 56, 62, 67, 75, 80, 85, 86, 90], event localisation [50, 51, 74, 78], audio-visual question answering [48, 54, 83, 84], audio-visual synchronisation [14, 23, 25, 38, 39, 42], audio synthesis using visual information [19, 26, 31, 44, 45, 61, 69–71, 87], or audio-driven face image synthesis [7, 40, 77]. Audio-visual data has also been leveraged for speech-related tasks, including speech and speaker recognition [2, 4, 59], or the spotting of spoken keywords [58, 66].

Furthermore, the natural alignment between audio and video has been exploited to learn improved audio-visual embeddings for downstream tasks [6, 10, 11, 18, 20, 21, 46, 60, 63–65, 79]. Using both modalities jointly generally leads to performance boosts over using one modality in isolation. We examine this observation closely and aim to evaluate the effective use of multiple input modalities for the video classification task. To enable this, we propose — to the best of our knowledge, the first multi-label video classification benchmark that includes per-modality annotations for every sample (see Tab. 1).

Dataset	# Clips	# Classes	Multi-label	Modality labels	Annotation pipeline
Flickr-SoundNet [10]	2M	-	✗	✗	-
Kinetics-Sound [7]	18.8k	34	✗	✗	MTurk
AudioSet [26]	2.1M	537	✓	✗	Manual
├ AVE [62]	4k	28	✓	✗	Manual
├ VEGAS [76]	132k	10	✗	✗	MTurk
└─┴ Visually Aligned Sounds [15]	13k	8	✗	✗	MTurk
VGGSound [12]	200k	309	✗	✗	Classifiers+Manual
├ VGGSound-Sparse [32]	7.1k	12	✗	✗	Manual
├ Visual Sound [66]	91k	309	✗	✗	ImageBind [30]
└ VGGSound*er*	15.4k	309	✓	✓	MTurk

Table 1. Comparison of audio-visual classification benchmarks.

Audio-visual foundation models Recently, multi-modal general-purpose models have emerged that can handle diverse downstream tasks without task-specific finetuning – also referred to as multi-modal foundation models. For instance, images or language were used as the bridge between modalities including audio, image, and text [30, 89]. Building on this, PandaGPT [72] leverages Vicuna [22] and ImageBind's embedding space to train a general multi-modal model exclusively on image-text pairs. Unified-IO 2 [53] employs universal tokenisation to process audio, video, and text. VideoLLaMA2 [21] uses a Spatial-Temporal Convolution connector in the visual branch before projecting audio and visual information into the LLM input space. The recently introduced Ola model [52] advances omni-modal processing through progressive modality alignment, using video to bridge audio and visual information. The Gemini models [73] are closed-source multi-modal models that achieve impressive performance on diverse downstream tasks. We use VGGSound*er* to benchmark the audio-visual capabilities of the aforementioned models.

Audio-visual classification benchmarks Audio-visual classification is distinct from general video classification (e.g. on YouTube-8M [1]), as classes typically cover both audible and visible actions or events. Commonly used datasets for audio-visual classification include Kinetics-Sound [8] sourced from the Kinetics dataset [41], Flickr-SoundNet [11] scraped from Flickr, and AudioSet [29] and VGGSound [13], both sourced from YouTube.

Kinetics-Sound features manual labels of human actions, but covers only 34 classes. Flickr-SoundNet is much larger, but only a small subset is labelled. Similarly, only a small fraction of the roughly 2M AudioSet samples are annotated and have aligned audio and video.

Figure 2. **Limitations of VGGSound.** We show video frames from videos in the VGGSound test set along with their annotated label (grey) to demonstrate various limitations. **A**. VGGSound samples are labelled with a single class, yet many videos contain multiple distinct classes. **B**. Additionally, many classes partially overlap or are ambiguous. **C**. Some samples are labelled with classes that are not present in one of the modalities (i.e., the labelled class is not visible or audible).

In contrast, VGGSound ensures audio-visual correspondence for around 200 000 samples and was curated with an automatic pipeline involving class-list generation, and auditory and visual content verification. The visual verification step ensures that a class is represented in the centre frame.
The VEGAS dataset [88] provides better quality assurances for a small subset of AudioSet with only 10 classes. Visually Aligned Sounds [16] subsamples VEGAS and AudioSet after human verification, and Visual Sound [76] subsamples VGGSound using a multi-modal embedding model, both aiming for high audio-visual correspondence. Similarly, VGGSound-Sparse [37] is a subset of VGGSound with a focus on temporally and spatially sparse synchronisation signals (e.g., short loud noises).
Overall, VGGSound strikes the best balance between size, generality, and annotations, making it a common benchmark for audio-visual classification. We update VGGSound to sustain its usability for the development of the next generation of multi-modal foundation models.

3. Limitations of VGGSound

Since we are interested in the VGGSound dataset for benchmarking audio-visual multi-modal models, our analysis focuses on the VGGSound test set,[1] which consists of 15 446 video clips, each 10s long and labelled with one of 309 classes. We qualitatively identify several limitations of the VGGSound annotations outlined below and in Fig. 2.

Co-occurring classes While VGGSound's visual verification aimed to minimise multiple classes co-occurring in a clip, we find that most samples nevertheless clearly contain multiple classes, see Fig. 2A. In some cases, classes are temporally separated, e.g., showing `male speech, man speaking` and then cutting to footage of `firing cannon`. Most often, classes co-occur at the same time, sometimes for the entire duration of the video clip. Overlapping classes are often related, such as different instruments in a band or orchestra, but can also be entirely unrelated, e.g., `donkey, ass braying` co-occurring with `playing violin`. As additional empirical evidence, we provide a co-occurrence matrix computed using CAV-MAE [33], a state-of-the-art audio-visual model, in Appendix D.

Overlapping classes The issue of co-occurring classes is exacerbated by many of the 309 automatically generated classes partially overlapping in their definition, as illustrated in Fig. 2B. We found two pairs of synonymous classes: `timpani` and `tympani` and `dog barking` and `dog bow-wow`. Additionally, some classes are strict subclasses of others, such as the gender-specific versions of `cattle mooing`: `cow lowing` and `bull bellowing`; or the more specific variants of `people eating`: `people eating noodle` and `people eating apple`. Finally, several classes commonly appear together, such as `playing snare drum` which is often played as part of a `drum kit`, or semantically similar concepts: `running electric fan` and `air conditioning noise`, and `sloshing water` and `splashing water`.

Modality misalignment Despite VGGSound's auditory and visual content verification, we find that many of the annotated classes are not visible or not audible, as shown in Fig. 2C. A large fraction of videos contains background music, voice-over and narration, or other background sounds like `bird chirping, tweeting` or `cricket chirping` without a visible source. Similarly, some videos contain visible but inaudible cues for classes like `sea waves`. Static images and slide shows accompa-

[1] Although these issues most likely also apply to the training set.

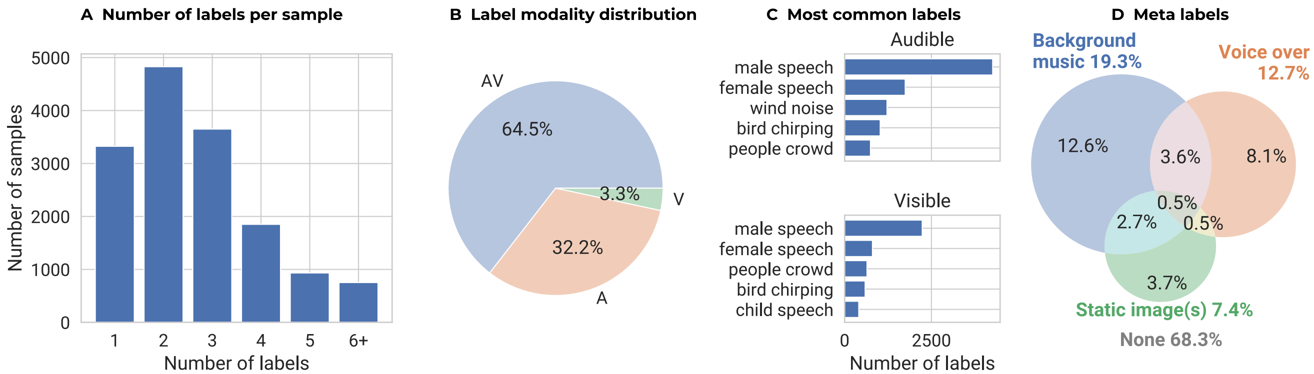


Figure 3. **Overview of VGGSound*er***. **A**. Most samples contain more than one label. **B**. More than a quarter of labels are audible but not visible. In contrast, only a tiny fraction is visible but not audible. **C**. Speech and bird sounds are the most common classes; more details can be found in Appendix B. **D**. Forty percent of the samples contain some combination of `background music`, `voice over`, and `static image(s)`, making the classification task significantly harder.

nied by music or other sounds are other frequent sources of misaligned modalities. Finally, some classes are misaligned by definition: `wind noise` is only audible and not visible. We find that, 48.43 % of the original VGGSound test samples have misaligned modalities. This finding challenges the widely held assumption that VGGSound has strong modality alignment [32, 47].

Other datasets, such as Visually Aligned Sounds, Visual Sound, and VGGSound-Sparse (see Sec. 2) omitted samples with misaligned modalities. In contrast, we contend that inaudible or invisible cues are common in natural videos and should be considered when benchmarking multi-modal models. We, therefore, place particular emphasis on crafting reliable modality annotations for all samples, allowing users to evaluate models on samples that guarantee modality alignment, and on those where classes are only visible or only audible (see Tab. 1).

Takeaway 1 VGGSound suffers from several issues: class co-occurrence not captured by single labels, overlapping class definitions, and modality misalignment, see Fig. 2.

4. Building VGGSound*er*

We propose a series of fixes for VGGSound's issues, ultimately resulting in the updated VGGSound*er* benchmark. We are not the first aiming to future-proof an existing benchmark: [12] analysed shortcomings of ImageNet [24], ultimately proposing switching to a multi-label classification task with additional manual labels. [28] similarly re-annotated samples in MMLU [35, 36] to fix labelling errors. Both works inspire our approach to improve VGGSound.

To deal with co-occurring classes, we switch to a multi-label classification setting. This effectively handles most overlapping class definitions: a strong model can assign a high probability to multiple classes, even if they partially overlap. This also allows us to ensure that synonymous classes, as well as subclasses and their superclass, always appear together in the ground-truth labels.

To deal with modality misalignment, we add a modality annotation to each label. For example, we can label a video as containing `people clapping` in the audio and containing `playing volleyball` in the audio and video data. We also add meta-labels to indicate whether a sample contains `background music`, `voice over`, or `static image(s)` to optionally treat these cases separately during evaluation.

We employ a pipeline similar to [12] to annotate multiple labels per sample, which we outline below.

Collecting proposals We create a *gold standard* reference set by labelling a small, randomly selected subset of VGGSound test samples with four in-house computer vision experts. The interface used for this first labelling step is shown in Appendix A. We extend the subset until each class is covered at least once, leading to a final size of 417 samples. Labels from different annotators are merged via a simple majority vote.

Given the gold standard set, we want to find a solid strategy for automatically generating label proposals which are shown to the humans labelling the test set. This should have a recall greater than 90 % while maximising precision compared to the gold standard labels to produce a small set of proposals with good label coverage. Our final strategy combines predictions from several state-of-the-art models with a manual heuristic to obtain 93% recall for an average of 30 proposals per sample, see Appendix A.

Human labelling We use Amazon Mechanical Turk to re-annotate the entire VGGSound test set. For each sample, we first ask annotators to indicate whether the video contains `background music`, `voice over`, or `static image(s)`. Then, annotators are asked to indicate for each label proposal whether the class is `audible` and/or `visible`. Finally, annotators can add a class if it is missing from the proposals. Annotators were paid the US minimum wage; the interface used is shown in Appendix A. We let annotators label the samples in batches of 20, each containing two gold standard samples as catch trials. We reject and re-annotate all batches with a catch trial F_1-score below 25%. In addition, we obtain modality annotations for the original VGGSound labels and meta-classes.

Final labels We merge all obtained annotations using majority voting. Additionally, we automatically add synonymous classes and superclasses for a given subclass, e.g., we add `cattle mooing` whenever `cow lowing` is in the set of labels. Further details can be found in Appendix A.

Takeaway 2 We develop VGGSound*er*: A multi-label video classification benchmark extending VGGSound with human-annotated multi-labels, modality annotations, and meta-labels as summarised in Fig. 3.

5. Benchmarking audio-visual models

We use VGGSound*er* to benchmark four popular audio-visual embedding models and seven foundation models, and analyse their auditory and visual capabilities.

Models We evaluate the audio-visual *embedding models* CAV-MAE [33], DeepAVFusion [57], AV-Siam [49], and Equi-AV [43]. Those were finetuned on VGGSound.

We benchmark several models from the closed-source Gemini family [73] in a zero-shot evaluation protocol. Furthermore, we use LLM-assisted evaluation to evaluate the following four open-source autoregressive *foundation models*: VideoLLaMA-2 [21], Unified-IO-2 [53], Panda-GPT [72], and Ola [52]. All models are evaluated in three modes: using unimodal-audio, unimodal-visual, or multi-modal (audio and visual) inputs. Further details about models and their evaluation are provided in Appendix C.1.

Metrics To benchmark the models on VGGSound*er*, we use multi-label classification metrics. For embedding models, all metrics are computed for the top-k predictions, with $k \in \{1, 3, 5, 10\}$. In contrast, prompting foundation models yields an unordered set of class predictions of varying size, and we compute only a single metric using the entire set. As a result, metrics are not directly comparable between embedding and foundation models. To get a sense of their relative performance, we report metrics for embedding models for $k = 1$ in the main text, matching the median number of predictions per sample for the foundation models.

For open-source models such as VideoLLaMA-2, Ola, Unified-IO-2, and Panda-GPT, we employ LLM-assisted evaluation [55, 81], in which the Qwen-3 model [82] is tasked to assess the correspondence between model outputs and target classes. Closed-source models from the Gemini family are evaluated by providing the full list of 309 classes as input. Further details on the evaluation procedures and exact prompts are provided in Appendix C.

Subset accuracy compares the predicted label set to the ground-truth label set and reports the fraction of samples for which they match. This is our strictest metric.

F_1-*score* is the harmonic mean of precision and recall. It is strictly larger than the subset accuracy.

Hit reports the fraction of samples for which *any* of the predicted labels are part of the ground-truth label set. This is the most lenient metric which is strictly larger than the F_1-score. We include this metric for ease of comparison to the "ReaL-Accuracy" used in [12].

All metrics are computed, on a subset of VGGSound*er* without `background music` labels, separately for each input modality (audio, video, and audio-visual) and label modality. We use lowercase symbols a, v, and av to indicate the input modality: audio-only, visual-only, or audio-visual inputs, respectively. For label modality, we use uppercase symbols *A*, *V*, and *AV*, referring to the subsets of the benchmark with audible, visible, and audio-visual labels. We further include $A\neg V$ (audible but not visible) and $V\neg A$ (visible but not audible) to analyze unaligned cues. For clarity, we define shorthand notations such as $a = a(A)$ to denote the model's performance on the audible subset A using only audio input. Analogously, $v = v(V)$ and $av = av(AV)$ refer to video-only and audio-visual performance on their respective label subsets. Furthermore, we use micro-aggregation to balance the contribution from each class.

We additionally measure the negative impact of using multi-modal inputs. In particular, μ is a new metric we propose to measure a model's *modality confusion* (μ). We define it as

$$\mu_{\mathrm{M}} = 100 \cdot \frac{\sum_{x \in M} \mathcal{I}[m(x)\text{-correct} \cap av(x)\text{-wrong}]}{N_{total}}, \quad (1)$$

where $M \in [A, V, A \cap V]$ and their associated modality inputs are $m \in [a, v]$, correct/wrong is determined as in the *Hit* score (with $k = 1$ for embedding models). N_{total} refers to the total number of samples. μ measures the fraction of samples a model correctly classified given an input modality but got wrong when using both modalities simultaneously. We additionally report $\mu_{A \cap V}$ as the percentage of samples a model could solve *in either modality* unimodally but could not solve multi-modally. In other words, the modality confusion μ captures how frequently a model is distracted by an additional input modality, which can indicate the unsuccessful merging of modalities.

Takeaway 3 We propose a new metric, *modality confusion* μ, that measures how frequently a model is distracted by an additional input modality; see Eq. (1).

Model	Subset Accuracy ↑			$\mathbf{F_1}$ ↑					Hit ↑			μ ↓		
	a	*v*	*av*	*a*	*v*	*av*	*a*($A \neg V$)	*v*($V \neg A$)	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A \cap V}$
Embedding Models														
CAV-MAE	**13.19**	19.23	**24.49**	**34.46**	34.91	**42.62**	**13.94**	**19.00**	**62.29**	53.44	**64.17**	**3.58**	6.43	0.77
DeepAVFusion	10.19	11.10	21.53	25.31	21.29	37.35	10.37	10.55	45.77	32.61	56.27	3.74	**3.93**	**0.17**
Equi-AV	11.60	10.52	20.00	29.39	20.42	34.69	12.55	10.65	53.12	31.26	52.24	6.97	7.13	1.38
AV-Siam	12.79	**19.75**	22.83	33.30	**35.41**	39.43	12.90	18.21	60.19	**54.20**	59.36	9.36	8.80	3.58
Closed-source Foundation Models														
Gemini 1.5 Flash	1.78	14.44	16.44	14.49	36.98	42.52	15.61	21.61	32.73	47.36	59.10	10.22	**4.25**	0.77
Gemini 1.5 Pro	**3.05**	**20.86**	**22.53**	**19.26**	**49.73**	**53.74**	**17.73**	**22.90**	**35.03**	**69.23**	**75.42**	**2.09**	4.85	**0.57**
Gemini 2.0 Flash	1.85	12.54	12.69	11.80	34.08	36.45	6.19	18.90	18.51	43.83	47.72	2.39	5.43	1.00
Open-source Foundation Models														
VideoLLaMA 2	12.86	19.85	24.47	38.87	**47.82**	**52.35**	20.34	**28.08**	**58.91**	**52.02**	59.80	12.72	5.46	2.95
Unified-IO 2	11.94	**11.56**	**25.61**	35.31	27.92	48.89	21.38	16.53	54.39	31.05	**65.11**	8.70	**5.16**	**1.79**
PandaGPT	3.19	4.19	5.46	18.73	18.56	20.85	16.82	14.40	21.08	17.01	18.82	**7.59**	5.90	2.47
Ola	**14.11**	8.69	18.19	**47.70**	24.85	46.48	**40.44**	13.45	59.05	24.57	51.51	15.47	6.80	2.49

Table 2. **Audio-visual video classification results on VGGSound*er*.** We report multi-label classification metrics (subset accuracy, F_1-score, Hit accuracy, modality confusion μ) on `background music` free subset for audio- $a(A)$, visual - $v(V)$, audio-visual - $av(AV)$, audio-only - $a(A \neg V)$ and video-only - $v(V \neg A)$ inputs. The embedding models CAV-MAE, DeepAVFusion, and Equi-AV were finetuned on the VGGSound training set. We report metrics for $k = 1$ here and for other k in Appendix D. The closed sourced multi-modal foundation models Gemini and open-sourced models use a zero-shot evaluation protocol and LLM-assisted protocol respectively.

5.1. Re-evaluating the state of the art

We present the benchmark performance of state-of-the-art audio-visual models in Tab. 2.

Overall performance Unsurprisingly, all models perform best with access to both input modalities (*AV*). Across all metrics, both open- and closed-source general-purpose models perform comparably to the purpose-built embedding models CAV-MAE and AV-Siam. This indicates that foundation models have reached — and for some modalities exceeded — the performance of specialised models. However, the embedding models finetuned on VGGSound generally have stronger unimodal performance with audio inputs (*A*) compared to visual inputs (*V*). Interestingly, this trend is reversed for most foundation models, which seem to be biased towards visual inputs, with unimodal video performance (*V*) being substantially higher than unimodal audio performance (*A*).

Takeaway 4 *Foundation models* perform comparably to finetuned *embedding models*. Embedding models more heavily rely on audio cues than on *visual* ones, while foundation models exploit *visible* cues rather than *audible* ones, see Tab. 2.

Modality confusion The *modality confusion score* μ shows that, for all models, a notable fraction of test samples (4–11%) were misclassified when an additional modality was included—despite being correctly classified with unimodal input. Furthermore, for all models, a small portion of test samples is not solvable multi-modally even though they were solvable in both modalities alone ($\mu_{A \cap V}$). In addition, for the majority of foundation models, μ_A is higher than μ_V, indicating that these models forget audible labels more often than visible labels when a second modality is introduced. This suggests that, in such cases, they prefer visible information over audible information. Interestingly, this phenomenon is reversed for the majority of embedding models. This insight is made possible by VGGSound*er*'s per-label modality annotations and shows that all models are susceptible to being distracted given an additional modality. This is a concerning issue for multi-modal models since they should preserve unimodal capabilities when adding a second modality. Being able to evaluate this behaviour on the VGGSound*er* benchmark is a first step towards enabling the development of mitigation strategies, eventually resulting in stronger audio-visual models.

Takeaway 5 Our *modality confusion score* reveals that *all models* are negatively impacted by additional modalities for a substantial amount of samples (see Tab. 2) and provides a means to quantify modality ensembles.

Performance across modalities Fig. 4 shows the performance profiles across modalities. At first glance, we can see that VideoLLaMA-2's performance is well balanced for different input modalities, while models from the Gemini family distinctly underperform on audio inputs. In contrast, embedding models exhibit a moderate balance across modalities, with DeepAVFusion and EquiAV showing slight underperformance for visual input.

As Fig. 4 also illustrates, profiling of this kind is enabled through the modality annotations in VGGSound*er*. In contrast, VGGSound assumed that all classes are perceivable in both modalities, and did not account for background sounds. This resulted in consistent under-evaluation of foundation models (that were not finetuned on VGGSound) for audio inputs.

In addition to the radar plot in Fig. 4, we provide results on VGGSound in Appendix D, showing that all models have substantially lower performance than their hit scores in Tab. 2. This confirms that many model predictions were incorrectly flagged as false positives in VGGSound due to the incomplete ground-truth labels, painting a distorted picture of models' limitations.

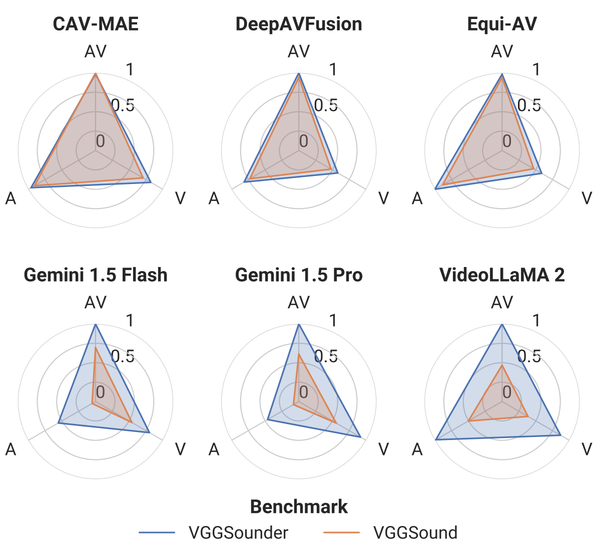


Figure 4. **VGGSound*er* more accurately captures model performance across input modalities**. We show the Hit score on VGGSound*er* and accuracy on VGGSound, normalised by the per-model maximum performance on each benchmark. Specifically for foundation models, we observe a significant difference in performance between VGGSound and VGGSound*er*.

Takeaway 6 VGGSound*er*'s more complete ground-truth labels allow for more accurate, modality-specific profiling of model performance (see Fig. 4).

5.2. Performance analysis using meta-classes

VGGSound*er* includes annotations of three meta-classes for each sample: `background music` indicates whether samples contain music without a visible source, `voice over` similarly marks speech without a visible source, and `static image(s)` flags that the visual stream consists of one or only a few static visual frames.

These new meta-classes allow us to evaluate the model behaviour in challenging scenarios where information from one modality dominates. We consider the performance difference between samples that *do* contain a meta-label and samples that *do not* contain it. Positive numbers indicate that the models perform better on the subset with meta-labels. In Tab. 3, we summarise the main findings, focussing on F_1-score as the most balanced multi-label metric. Additional results are provided in Appendix D.

Background music All models perform worse on video samples containing background music. This indicates that it is challenging to decouple background audio from the rest of the video. The evaluated models are not good at differentiating between sound sources without visual cues, e.g. predicting different instruments in the background music.

Voice over In contrast to background music, we observe a clear difference between embedding models and foundation models for samples with voice-over. While the audio classification performance of embedding models drops substantially. This drop is only slight for VideoLLaMA-2 and Unified-IO 2, and the performance of other foundation models even improves. This indicates that the foundation models are less distracted by voice-over.

Static image(s) The impact of static images is more nuanced: First, audio classification performance improves for embedding models while it decreases for the half of the foundation models. This shows that the purpose-built, VGGSound-finetuned embedding models can more accurately predict specific sounds in the absence of other cues. Second, visual classification performance on static images drops for all models, suggesting that models rely on rich temporal cues to make accurate predictions.

Samples without any meta-label When comparing the model performance on samples without background music, static images, or voice-over annotations (column *neither* in Tab. 3) to the performance on samples that contain either of meta classes, we see a performance gain. This finding concludes that these three categories form challenging subsets of the dataset.

Takeaway 7 Samples with *background music*, *static images(s)*, and *voice over* provide distinct challenges for each model (see Tab. 3). This highlights VGGSound*er*'s value for comprehensive model evaluation.

5.3. Impact of VGGSound*er* labels

Our relabelling pipeline adds two types of labels to those in VGGSound: (1) automatically generated labels based on synonymous classes and sub-/superclasses, and (2) human-curated labels. In Tab. 4 , we ablate the impact of each type of added labels in terms of the relative performance gains (Hit score). A complete breakdown of the effects across all models and metrics is provided in Appendix D. While performance is consistently higher with automatically added

	background music			voice over	static image(s)						neither		
	Δ $\mathbf{F_1}$			$\Delta\mathbf{F_1}$	Δ $\mathbf{F_1}$		**Sub. Acc.** w/		**Sub. Acc.** w/o		Δ $\mathbf{F_1}$		
Model	*a*	*v*	*av*	*a*	*a*	*v*	*a*	*v*	*a*	*v*	*a*	*v*	*av*
Embedding Models													
CAV-MAE	−3.43	−3.60	−4.01	−8.19	4.75	−7.39	22.13	19.48	11.98	19.21	−3.14	−4.71	−4.65
DeepAVFusion	−3.65	−4.86	−4.27	−9.05	4.33	−4.88	15.98	10.96	9.30	10.81	−3.97	−3.71	−5.07
Equi-AV	−4.07	−2.53	−2.54	−7.13	4.26	−5.99	19.00	10.39	10.49	10.45	−3.04	−3.58	−3.33
AV-Siam	−3.75	−4.39	−5.10	−7.96	5.23	−6.85	22.04	19.81	11.57	19.52	−3.19	−4.97	−5.20
Closed-source Foundation Models													
Gemini 1.5 Flash	−1.17	−2.39	−4.17	17.25	−5.28	−7.31	1.43	13.47	1.68	14.39	4.57	−4.07	−6.15
Gemini 1.5 Pro	−1.86	−3.67	−5.80	18.16	−4.90	−6.86	2.33	22.08	2.87	20.80	5.19	−3.28	−4.02
Gemini 2.0 Flash	−0.47	−1.92	−3.47	0.20	1.96	−7.11	3.85	10.88	1.53	12.40	−0.09	−3.97	−4.53
Open-source Foundation Models													
VideoLLaMA 2	−2.43	−4.62	−5.52	−3.97	4.22	−9.52	19.00	18.18	12.04	19.70	−1.48	−4.86	−5.18
Unified-IO 2	−6.41	1.15	−4.18	−4.98	1.88	−6.19	17.92	9.58	10.88	12.00	−3.42	−0.57	−4.53
PandaGPT	−5.98	−0.93	−2.75	3.92	−3.68	−4.86	3.32	4.87	2.91	4.24	−1.92	−0.58	−2.53
OLA	−11.84	0.63	−2.87	10.09	−8.24	−5.40	14.87	8.60	12.88	8.89	−2.54	−0.49	−0.24

Table 3. **Summary of performance differences in the presence/absence of meta-classes.** Difference in $\mathbf{F_1}$ scores (Δ $\mathbf{F_1}$) for audio-visual video classification on VGGSound*er* between videos *with* a meta-class and those *without* it. Positive numbers (Δ) indicate better performance when the meta-class is present. Additional results are provided in Appendix D.

	Human labels ↑			Auto labels ↑		
Model	*A*	*V*	*AV*	*A*	*V*	*AV*
Gemini 1.5 Flash	29.28	14.59	16.36	0.48	0.93	1.51
Gemini 1.5 Pro	28.61	25.52	27.63	0.31	1.99	2.10
Gemini 2.0 Flash	8.80	12.16	11.13	0.22	1.12	1.28

Table 4. **Impact of added labels using different strategies in VGGSound*er*.** We show the change in multi-label classification performance (Δ Hit) when adding automatically added (Auto) or human-annotated (Human) labels to VGGSound*er*, and compare to the original VGGSound data.

labels (Auto), the increase is noticeably smaller than that for human-curated labels. Paired with the observation that models do frequently predict correct classes that were not part of the original VGGSound label set, this indicates that human-curated labels better cover the ground truth.

Takeaway 8 Automatically added labels are an important step, but human-curated labels have a bigger effect on eliminating incorrectly flagged false positives, underscoring the value of accurate human annotation.

6. Discussion

Choice of VGGSound as base dataset VGGSound is commonly used to evaluate audio-visual models on the multimodal video classification task. As it is currently the most suitable testbed for audio-visual classification tasks (due to its size, diversity of categories, non-constrained setting, and relatively strong audio-visual correspondence), it serves as an optimal starting point for our substantially improved VGGSound*er* benchmark with a multi-label evaluation protocol for foundation models that makes the benchmark suitable for meaningful evaluation.

Multi-label vs single-label classification Video content is inherently complex, often containing multiple co-occurring objects and actions both within and across modalities. This makes it unlikely that a given clip belongs to just one class as is the case in the single-label classification task. Therefore, our VGGSound*er* extends the VGGSound test set to multi-label classification. Unlike models trained on a narrow single-label dataset, foundation models develop versatile representations.

7. Conclusion

We introduced an modality-aware evaluation set for audio-visual foundation models. VGGSound*er* builds on the widely used VGGSound dataset by adding: (a) comprehensive human annotations for missing classes, (b) specifying modality information per label, (c) introducing specialised meta-labels for frequently occurring real-world challenges, and (d) using heuristic methods to improve label quality.
Through our newly introduced metric, modality confusion, we observe that incorporating additional modalities does not necessarily yield better results. Models often become more confused on a substantial subset of test samples. Furthermore, finetuned embedding models tend to rely heavily on audio cues, while foundation models depend more on visual information. Additionally, our meta-label analysis highlights distinct challenges across various specialised yet commonly occurring scenarios such as background music, static images, and voice-overs. Overall, we hope that the VGGSound*er* benchmark will advance the evaluation and development of foundational audio-visual models.

Acknowledgements

The authors would like to thank Felix Förster, Sayak Mallick, and Prasanna Mayilvahananan for their help with data annotation, Felix Förster and Monica Riedler for proofreading the paper, and Thomas Klein and Shyamgopal Karthik for their help in setting up MTurk. They also thank numerous MTurk workers for labelling.

This work was in part supported by the BMBF (FKZ: 01IS24060, 01IS24085B, 01IS18039A), the DFG (SFB 1233, project number: 276693517), and the Open Philanthropy Foundation funded by the Good Ventures Foundation. WB acknowledges financial support via an Emmy Noether Grant funded by the German Research Foundation (DFG) under grant no. BR 6382/1-1. WB is a member of the Machine Learning Cluster of Excellence, EXC number 2064/1 – Project number 390727645. This research utilised compute resources at the Tübingen Machine Learning Cloud, DFG FKZ INST 37/1057-1 FUGG. The authors thank the IMPRS-IS for supporting TW.

Author contributions This project was co-led and coordinated by ASK, DZ, TW, and AP. TW, DZ, and ASK analysed and identified issues with VGGSound. TW and DZ implemented the annotation pipeline with input from ASK, AP, and WB. TW ran annotations on MTurk and produced the final label set with support from DZ and input from ASK, AP, and WB. DZ implemented the in-house annotation pipelines with input from ASK, TW, and AP. DZ trained in-house versions of multiple models with help from ASK. DZ evaluated all models with support from ASK and input from TW and AP. TW, ASK, DZ, and AP wrote the manuscript with input from WB and MB. TW and DZ created the figures with feedback from ASK, AP, and WB. MB provided helpful feedback throughout the project.

References

[1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*, 2016. 2

[2] Triantafyllos Afouras, Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. Deep audio-visual speech recognition. *IEEE TPAMI*, 2018. 2

[3] Triantafyllos Afouras, Yuki M Asano, Francois Fagan, Andrea Vedaldi, and Florian Metze. Self-supervised object detection from audio-visual correspondence. In *ECCV*, 2020. 2

[4] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. Asr is all you need: Cross-modal distillation for lip reading. In *ICASSP*, 2020. 2

[5] Triantafyllos Afouras, Andrew Owens, Joon Son Chung, and Andrew Zisserman. Self-supervised learning of audio-visual objects from video. In *ECCV*, 2020. 2

[6] Humam Alwassel, Dhruv Mahajan, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. In *NeurIPS*, 2020. 2

[7] Shivangi Aneja, Justus Thies, Angela Dai, and Matthias Nießner. Facetalk: Audio-driven motion diffusion for neural parametric head models. In *CVPR*, 2024. 2

[8] Relja Arandjelović and Andrew Zisserman. Look, listen and learn. In *ICCV*, 2017. 2

[9] Relja Arandjelovic and Andrew Zisserman. Objects that sound. In *ECCV*, 2018. 2

[10] Yuki M. Asano, Mandela Patrick, Christian Rupprecht, and Andrea Vedaldi. Labelling unlabelled videos from scratch with multi-modal self-supervision. In *NeurIPS*, 2020. 2

[11] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. In *NeurIPS*, 2016. 2

[12] Lucas Beyer, Olivier J Hénaff, Alexander Kolesnikov, Xiaohua Zhai, and Aäron van den Oord. Are we done with imagenet? *arXiv preprint arXiv:2006.07159*, 2020. 2, 4, 5

[13] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. Vggsound: A large-scale audio-visual dataset. In *ICASSP*, 2020. 1, 2

[14] Honglie Chen, Weidi Xie, Triantafyllos Afouras, Arsha Nagrani, Andrea Vedaldi, and Andrew Zisserman. Audio-visual synchronisation in the wild. In *BMVC*, 2021. 2

[15] Honglie Chen, Weidi Xie, Triantafyllos Afouras, Arsha Nagrani, Andrea Vedaldi, and Andrew Zisserman. Localizing visual sounds the hard way. In *CVPR*, 2021. 2

[16] Peihao Chen, Yang Zhang, Mingkui Tan, Hongdong Xiao, Deng Huang, and Chuang Gan. Generating visually aligned sound from videos. *IEEE Transactions on Image Processing*, 2020. 3

[17] Sanyuan Chen, Yu Wu, Chengyi Wang, Shujie Liu, Daniel Tompkins, Zhuo Chen, and Furu Wei. Beats: Audio pre-training with acoustic tokenizers. *arXiv preprint arXiv:2212.09058*, 2022. 17

[18] Yanbei Chen, Yongqin Xian, A. Sophia Koepke, Ying Shan, and Zeynep Akata. Distilling audio-visual knowledge by compositional contrastive learning. In *CVPR*, 2021. 2

[19] Ziyang Chen, Prem Seetharaman, Bryan Russell, Oriol Nieto, David Bourgin, Andrew Owens, and Justin Salamon. Video-guided foley sound generation with multimodal controls. In *CVPR*, 2025. 2

[20] Ying Cheng, Ruize Wang, Zhihao Pan, Rui Feng, and Yuejie Zhang. Look, listen, and attend: Co-attention network for self-supervised audio-visual representation learning. In *ACM MM*, 2020. 2

[21] Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, et al. Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms. *arXiv preprint arXiv:2406.07476*, 2024. 2, 5, 17, 18

[22] Wei-Lin Chiang, Zhuohan Li, Ziqing Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. Vicuna: An

open-source chatbot impressing gpt-4 with 90%* chatgpt quality. 2023. 2
[23] Joon Son Chung and Andrew Zisserman. Out of time: automated lip sync in the wild. In *ACCV*, 2016. 2
[24] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 4
[25] Joshua P Ebeneze, Yongjun Wu, Hai Wei, Sriram Sethuraman, and Zongyi Liu. Detection of audio-video synchronization errors via event detection. In *ICASSP*, 2021. 2
[26] Chuang Gan, Deng Huang, Peihao Chen, Joshua B Tenenbaum, and Antonio Torralba. Foley music: Learning to generate music from videos. In *ECCV*, 2020. 2
[27] Ruohan Gao and Kristen Grauman. Co-separating sounds of visual objects. In *ICCV*, 2019. 2
[28] Aryo Pradipta Gema, Joshua Ong Jun Leang, Giwon Hong, Alessio Devoto, Alberto Carlo Maria Mancino, Rohit Saxena, Xuanli He, Yu Zhao, Xiaotang Du, Mohammad Reza Ghasemi Madani, Claire Barale, Robert McHardy, Joshua Harris, Jean Kaddour, Emile van Krieken, and Pasquale Minervini. Are we done with mmlu? *arXiv preprint arXiv:2406.04127*, 2024. 2, 4
[29] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *ICASSP*, 2017. 2
[30] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *CVPR*, 2023. 2
[31] Shir Goldstein and Yael Moses. Guitar music transcription from silent video. In *BMVC*, 2018. 2
[32] Yuan Gong, Alexander H Liu, Andrew Rouditchenko, and James Glass. Uavm: Towards unifying audio and visual models. *IEEE Signal Processing Letters*, 2022. 4
[33] Yuan Gong, Andrew Rouditchenko, Alexander H Liu, David Harwath, Leonid Karlinsky, Hilde Kuehne, and James Glass. Contrastive audio-visual masked autoencoder. In *ICLR*, 2023. 1, 3, 5, 17, 21
[34] Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, et al. A survey on llm-as-a-judge. *arXiv preprint arXiv:2411.15594*, 2024. 20
[35] Dan Hendrycks, Collin Burns, Steven Basart, Andrew Critch, Jerry Li, Dawn Song, and Jacob Steinhardt. Aligning ai with shared human values. In *ICLR*, 2021. 4
[36] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *ICLR*, 2021. 4
[37] Vladimir Iashin, Weidi Xie, Esa Rahtu, and Andrew Zisserman. Sparse in space and time: Audio-visual synchronisation with trainable selectors. In *BMVC*, 2022. 3
[38] Vladimir Iashin, Weidi Xie, Esa Rahtu, and Andrew Zisserman. Sparse in space and time: Audio-visual synchronisation with trainable selectors. In *BMVC*, 2022. 2
[39] Vladimir Iashin, Weidi Xie, Esa Rahtu, and Andrew Zisserman. Synchformer: Efficient synchronization from sparse cues. In *ICASSP*, 2024. 2
[40] Amir Jamaludin, Joon Son Chung, and Andrew Zisserman. You said that?: Synthesising talking faces from audio. *IJCV*, 2019. 2
[41] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 2
[42] Naji Khosravan, Shervin Ardeshir, and Rohit Puri. On attention modules for audio-visual synchronization. In *CVPR Workshop*, 2019. 2
[43] Jongsuk Kim, Hyeongkeun Lee, Kyeongha Rho, Junmo Kim, and Joon Son Chung. Equiav: Leveraging equivariance for audio-visual contrastive learning. In *ICML*, 2024. 1, 5, 17
[44] A. Sophia Koepke, Olivia Wiles, and Andrew Zisserman. Visual pitch estimation. In *SMC*, 2019. 2
[45] A Sophia Koepke, Olivia Wiles, Yael Moses, and Andrew Zisserman. Sight to sound: An end-to-end approach for visual piano transcription. In *ICASSP*, 2020. 2
[46] Bruno Korbar, Du Tran, and Lorenzo Torresani. Cooperative learning of audio and video models from self-supervised synchronization. In *NeurIPS*, 2018. 2
[47] Sangho Lee, Jiwan Chung, Youngjae Yu, Gunhee Kim, Thomas Breuel, Gal Chechik, and Yale Song. Acav100m: Automatic curation of large-scale datasets for audio-visual video representation learning. In *ICCV*, 2021. 4
[48] Guangyao Li, Yake Wei, Yapeng Tian, Chenliang Xu, Ji-Rong Wen, and Di Hu. Learning to answer questions in dynamic audio-visual scenarios. In *CVPR*, 2022. 2
[49] Yan-Bo Lin and Gedas Bertasius. Siamese vision transformers are scalable audio-visual learners. In *ECCV*, 2024. 1, 5, 17
[50] Yan-Bo Lin and Yu-Chiang Frank Wang. Audiovisual transformer with instance attention for audio-visual event localization. In *ACCV*, 2020. 2
[51] Yan-Bo Lin, Yu-Jhe Li, and Yu-Chiang Frank Wang. Dual-modality seq2seq network for audio-visual event localization. In *ICASSP*, 2019. 2
[52] Zuyan Liu, Yuhao Dong, Jiahui Wang, Ziwei Liu, Winston Hu, Jiwen Lu, and Yongming Rao. Ola: Pushing the frontiers of omni-modal language model. *arXiv preprint arXiv:2502.04328*, 2025. 2, 5, 17
[53] Jiasen Lu, Christopher Clark, Sangho Lee, Zichen Zhang, Savya Khosla, Ryan Marten, Derek Hoiem, and Aniruddha Kembhavi. Unified-io 2: Scaling autoregressive multimodal models with vision language audio and action. In *CVPR*, 2024. 2, 5, 17
[54] Jie Ma, Min Hu, Pinghui Wang, Wangchun Sun, Lingyun Song, Hongbin Pei, Jun Liu, and Youtian Du. Look, listen, and answer: Overcoming biases for audio-visual question answering. *NeurIPS*, 2024. 2
[55] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. Video-chatgpt: Towards detailed video

understanding via large vision and language models. *arXiv preprint arXiv:2306.05424*, 2023. 5

[56] Shentong Mo and Pedro Morgado. Localizing visual sounds the easy way. In *ECCV*, 2022. 2

[57] Shentong Mo and Pedro Morgado. Unveiling the power of audio-visual early fusion transformers with dense interactions through masked modeling. In *CVPR*, 2024. 1, 5, 17

[58] Liliane Momeni, Triantafyllos Afouras, Themos Stafylakis, Samuel Albanie, and Andrew Zisserman. Seeing wake words: Audio-visual keyword spotting. In *BMVC*, 2020. 2

[59] Arsha Nagrani, Joon Son Chung, Samuel Albanie, and Andrew Zisserman. Disentangled speech embeddings using cross-modal self-supervision. In *ICASSP*, 2020. 2

[60] Arsha Nagrani, Shan Yang, Anurag Arnab, Aren Jansen, Cordelia Schmid, and Chen Sun. Attention bottlenecks for multimodal fusion. In *NeurIPS*, 2021. 2

[61] Medhini Narasimhan, Shiry Ginosar, Andrew Owens, Alexei A Efros, and Trevor Darrell. Strumming to the beat: Audio-conditioned contrastive video textures. *arXiv preprint arXiv:2104.02687*, 2021. 2

[62] Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multisensory features. In *ECCV*, 2018. 2

[63] Andrew Owens, Jiajun Wu, Josh H McDermott, William T Freeman, and Antonio Torralba. Ambient sound provides supervision for visual learning. In *ECCV*, 2016. 2

[64] Andrew Owens, Jiajun Wu, Josh H McDermott, William T Freeman, and Antonio Torralba. Learning sight from sound: Ambient sound provides supervision for visual learning. *IJCV*, 2018.

[65] Mandela Patrick, Yuki M Asano, Ruth Fong, João F Henriques, Geoffrey Zweig, and Andrea Vedaldi. Multi-modal self-supervision from generalized data transformations. In *NeurIPS*, 2020. 2

[66] KR Prajwal, Liliane Momeni, Triantafyllos Afouras, and Andrew Zisserman. Visual keyword spotting with attention. In *BMVC*, 2021. 2

[67] Rui Qian, Di Hu, Heinrich Dinkel, Mengyue Wu, Ning Xu, and Weiyao Lin. Multiple sound sources localization from coarse to fine. In *ECCV*, 2020. 2

[68] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *ICLR*, 2023. 17

[69] Kun Su, Xiulong Liu, and Eli Shlizerman. Multi-instrumentalist net: Unsupervised generation of music from body movements. *arXiv preprint arXiv:2012.03478*, 2020. 2

[70] Kun Su, Xiulong Liu, and Eli Shlizerman. How does it sound? In *NeurIPS*, 2021.

[71] Kun Su, Xiulong Liu, and Eli Shlizerman. From vision to audio and beyond: A unified model for audio-visual representation and generation. In *ICML*, 2024. 2

[72] Yixuan Su, Tian Lan, Huayang Li, Jialu Xu, Yan Wang, and Deng Cai. Pandagpt: One model to instruction-follow them all. *arXiv preprint arXiv:2305.16355*, 2023. 2, 5, 17

[73] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 2, 5, 17

[74] Yapeng Tian, Jing Shi, Bochen Li, Zhiyao Duan, and Chenliang Xu. Audio-visual event localization in unconstrained videos. In *ECCV*, 2018. 2

[75] Efthymios Tzinis, Scott Wisdom, Aren Jansen, Shawn Hershey, Tal Remez, Daniel PW Ellis, and John R Hershey. Into the wild with audioscope: Unsupervised audio-visual separation of on-screen sounds. In *ICLR*, 2021. 2

[76] Ilpo Viertola, Vladimir Iashin, and Esa Rahtu. Temporally aligned audio for video with autoregression. In *ICASSP*, 2025. 3

[77] Olivia Wiles, A. Sophia Koepke, and Andrew Zisserman. X2face: A network for controlling face generation using images, audio, and pose codes. In *ECCV*, 2018. 2

[78] Yu Wu, Linchao Zhu, Yan Yan, and Yi Yang. Dual attention matching for audio-visual event localization. In *CVPR*, 2019. 2

[79] Fanyi Xiao, Yong Jae Lee, Kristen Grauman, Jitendra Malik, and Christoph Feichtenhofer. Audiovisual slowfast networks for video recognition. *arXiv preprint arXiv:2001.08740*, 2020. 2

[80] Haoming Xu, Runhao Zeng, Qingyao Wu, Mingkui Tan, and Chuang Gan. Cross-modal relation-aware networks for audio-visual event localization. In *ACM MM*, 2020. 2

[81] Runsen Xu, Xiaolong Wang, Tai Wang, Yilun Chen, Jiangmiao Pang, and Dahua Lin. Pointllm: Empowering large language models to understand point clouds. In *ECCV*, 2024. 5, 18

[82] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. 5, 18

[83] Pinci Yang, Xin Wang, Xuguang Duan, Hong Chen, Runze Hou, Cong Jin, and Wenwu Zhu. Avqa: A dataset for audio-visual question answering on videos. In *ACM MM*, 2022. 2

[84] Heeseung Yun, Youngjae Yu, Wonsuk Yang, Kangil Lee, and Gunhee Kim. Pano-avqa: Grounded audio-visual question answering on 360deg videos. In *ICCV*, 2021. 2

[85] Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In *ECCV*, 2018. 2

[86] Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. The sound of motions. In *ICCV*, 2019. 2

[87] Hang Zhou, Ziwei Liu, Xudong Xu, Ping Luo, and Xiaogang Wang. Vision-infused deep audio inpainting. In *ICCV*, 2019. 2

[88] Yipin Zhou, Zhaowen Wang, Chen Fang, Trung Bui, and Tamara L. Berg. Visual to sound: Generating natural sound for videos in the wild. In *CVPR*, 2018. 3

[89] Bin Zhu, Bin Lin, Munan Ning, Yang Yan, Jiaxi Cui, HongFa Wang, Yatian Pang, Wenhao Jiang, Junwu Zhang, Zongwei Li, et al. Languagebind: Extending video-language pretraining to n-modality by language-based semantic alignment. *arXiv preprint arXiv:2310.01852*, 2023. 2

[90] Lingyu Zhu and Esa Rahtu. V-slowfast network for efficient visual sound separation. In *WACV*, 2022. 2

Supplementary Material:
VGGSound*er*: Audio-Visual Evaluations for Foundation Models

A. VGGSound*er*: Relabelling VGGSound

In the main paper, we highlighted several critical shortcomings of VGGSound, such as co-occurring classes, partially overlapping class definitions, multiple classes per sample, and modality misalignment. This appendix provides additional details about the relabelling process for obtaining the VGGSound*er* benchmark, addressing the specific issues identified in VGGSound.

A.1. Labelling of the gold-standard subset

As described in Sec. 4, we started by creating a high-quality reference subset (gold-standard) for reliable label verification. Four experienced computer vision researchers manually annotated randomly selected 10-second videos from the VGGSound test set. Annotators labelled classes clearly present either audibly, visually, or both. We ensured full class coverage by continuing the annotation process until all classes appeared at least once, resulting in 417 samples. These annotations were merged using majority voting. The annotation interface employed in this phase is illustrated in Fig. 5.

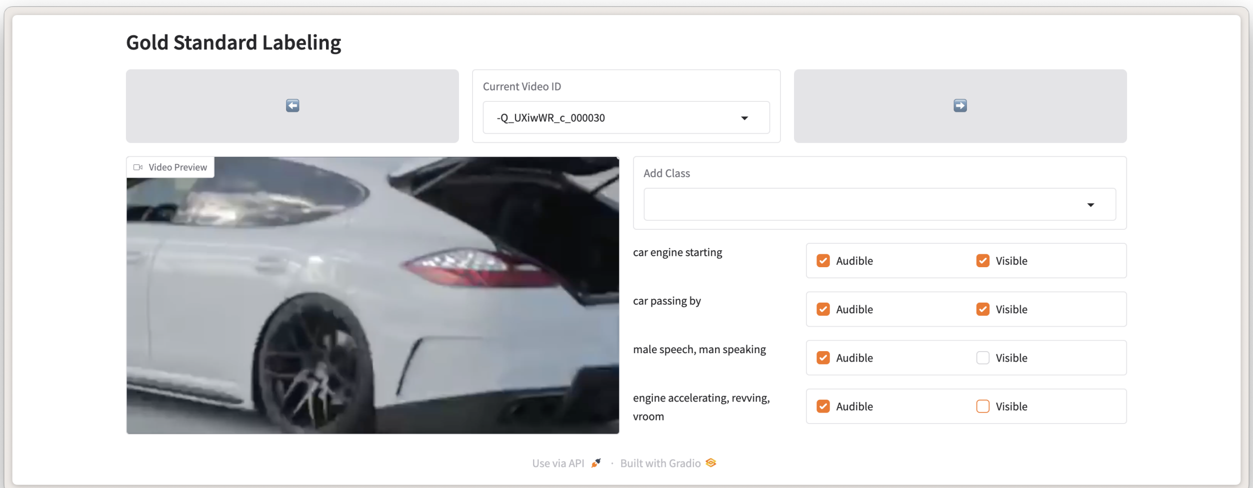


Figure 5. Interface used to annotate the gold standard set in-house.

The annotators were instructed to try to identify all audible and visible classes in the video, including hard cases when background music contains several instruments that compose the melody. For instance, a common instrument for the country music genre would be `playing the drum kit`, `female singing`, `male singing`, `playing the bass guitar`, `playing electric guitar` etc. The annotators are expected to do their best to identify all of the instruments.

> Gold-standard samples serve as high-quality annotations for further labellers' cross-validation and automatic quality assessment. If a labeller shows a high agreement score with the gold-standard labels, we expect them to have high-quality labels outside of the gold-standard subset.

While analysing gold-standard labels, we made several interesting observations (see Tab. 5):

1. There is a significant portion of samples in the gold-standard set for which the original VGGSound labels (24.46%) are absent.
2. The proportion of classes that are only audible across all samples is significantly higher than that of the visible ones.

Metric	Value
Samples	417
Original class correct	283 (67.87%)
Original class audible	39 (9.35%)
Original class visible	22 (5.88%)
Original class absent	102 (24.46%)
Original class is only class	71 (17.03%)
Classes total	309
Classes only visible	6
Classes only audible	25
Average labels added per sample	1.39

Table 5. Relabelling statistics for the gold-standard subset.

While we cannot fix the second issue without substituting the dataset, the first issue quantifies the error introduced by VGGSound and its automatic labelling and verification and can be eliminated with human labelling.

We ran a second round of gold-standard annotations where one computer vision expert checked all 15446 samples and annotations in the VGGSound test set for their validity and enriched the correct labels with modality annotations. The interface for this annotation is illustrated in Fig. 6.

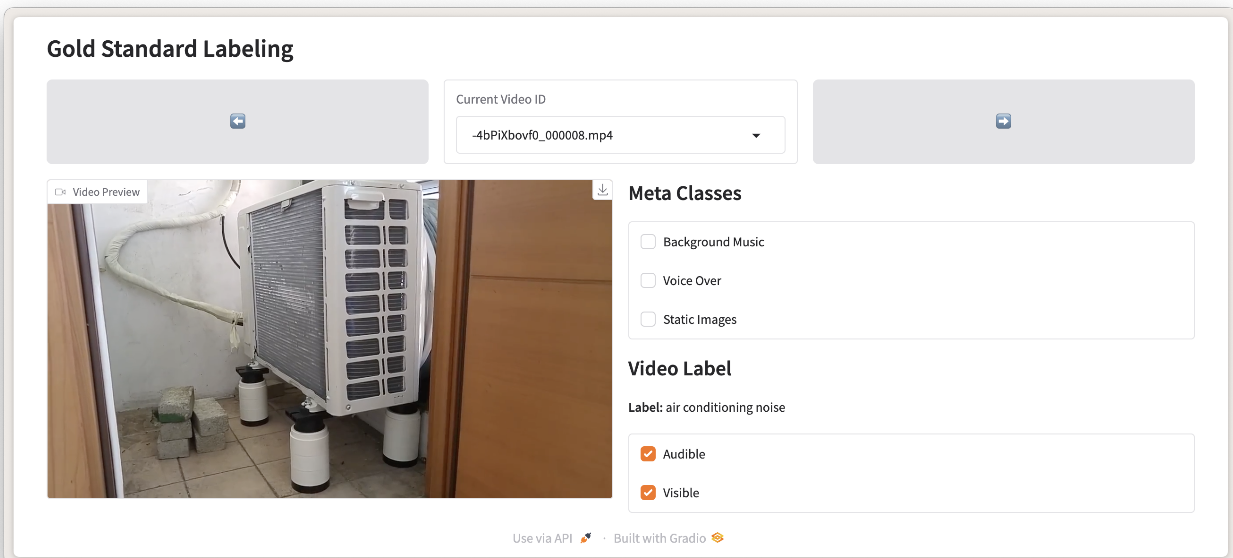


Figure 6. Interface used in-house to annotate the original labels in the VGGSound test set.

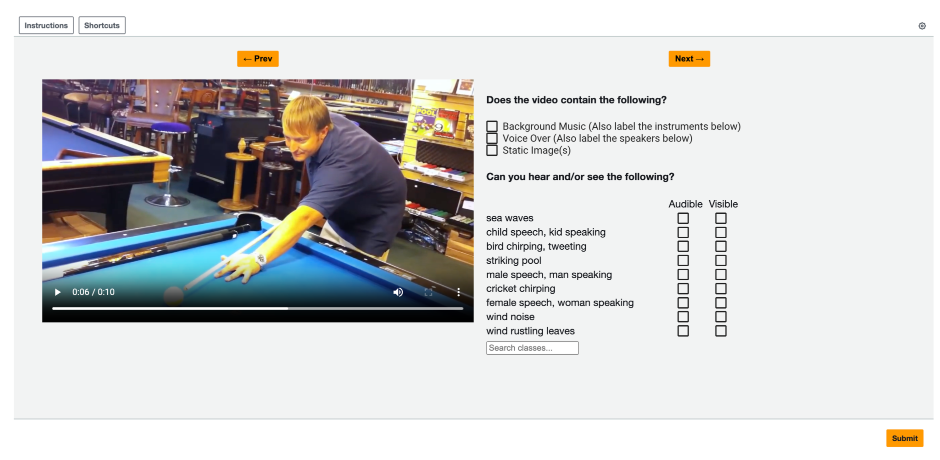
The second set of gold-standard labels firstly enriched the original VGGSound labels with modality annotations, but most importantly confirmed and further improved the estimates in Tab. 5 resulting in the following observation:

Around 48.43% of the original VGGSound test samples have either incorrect target labels or misaligned modalities.

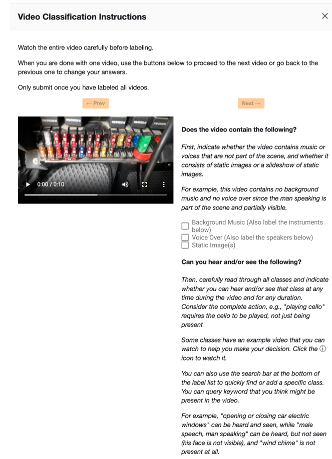
The two sets of gold-standard annotations, while having mixed reliability (cross-validation with four people vs. one person), serve as a strong grounding signal for our subsequent MTurk annotation pipelines.

A.2. Label proposals

To effectively scale human annotations to the entire test set, and to simplify the job for MTurk annotators, we introduced a label proposal generation strategy that combines state-of-the-art audio-visual model predictions with label heuristics. We



(a) **Example labelling interface for one video sample.**



(b) **Labelling instructions**

Figure 7. **Labelling interface and instructions for our full annotation pipeline that we ran on MTurk.** (a) Crowd workers are presented with a 10-second long video clip from the VGGSound test set, along with label proposals. They are tasked to select if those or additional VGGSound classes are audible or visible in the video clip. Furthermore, the workers are asked about meta-classes, such as background music, voice-over, and static images. They also have the option of searching for new classes that are missing in the proposals. (b) Labelling instructions provided to workers on Amazon Mechanical Turk before labelling the first video sample.

considered the following steps in our label proposals:

1. **Model predictions:**
   - We provide the original VGGSound label, extended with modality annotations curated by an in-house labeler, as well as the top-1 predictions of the following models [2] with visual and audio-visual inputs:
     - CAV-MAE
     - AVSiam
     - Equi-AV
     - DeepAVFusion
     - Gemini 1.5 Flash
     - Gemini 1.5 Pro
   - We further included the top-5 predictions when using audio inputs from the same models.
2. **Consensus labels:**
   - We created a secondary pool from the top-10 predictions across all modalities from AVSiam, CAV-MAE, and Equi-AV. Additionally, labels associated with the highest 60,000 logits or probabilities across the dataset were added.
   - Labels were proposed from this pool if at least two models independently agreed on their presence.
3. **Common classes:**
   - Regardless of model predictions, we always proposed frequently occurring classes such as:

`wind noise`, `wind rustling leaves`, `male speech, man speaking`, `female speech, woman speaking`, `child speech, kid speaking`, `bird chirping, tweeting`, `cricket chirping`, `sea waves`.

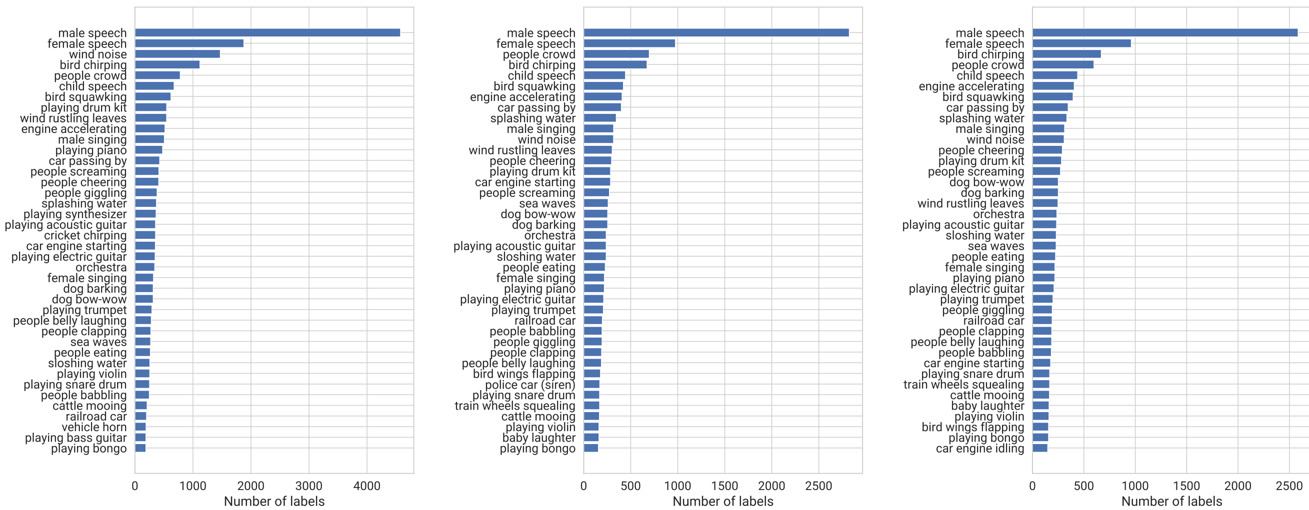
This strategic combination ensured an average of 30 proposals per video, achieving approximately 93% recall relative to the gold-standard set annotations.

A.3. Human labelling

Following our proposal strategy, we conducted extensive human annotation via Amazon Mechanical Turk (MTurk) to verify and expand the automatically generated proposals:

- **Worker qualifications:** We used two types of Amazon Mechanical Turk (AMT) worker qualifications for our annotations. Half of the tasks were completed by AMT Masters with approval rates above 98%, while the other half were assigned to

[2] Gemini 2.0 Flash, VideoLLaMA-2, Unified-IO-2, Panda-GPT, and Ola were not used when the proposals were generated.



(a) *Audible* classes. (b) *Visible* classes. (c) *Audible and visible* classes.

Figure 8. Class label frequency in VGGSound*er* by modality.

non-Master workers who also maintained approval rates above 98%. We adopted this approach as the larger non-Master worker pool resulted in significantly faster task completion.

- **Annotation interface:** Annotators reviewed each video to confirm the presence and modality (audible, visible, or both) of proposed labels. They could also suggest additional missing labels. Workers received detailed instructions (see Fig. 7b), and the annotation interface used is presented in Fig. 7a.
- **Quality control:** Videos were grouped into batches of 20, each containing two gold-standard samples as catch trials. Batches scoring below 25% F1-score on these catch trials were rejected and reassigned.

A.4. Automatically added classes

To resolve overlapping and ambiguous class definitions discussed in Sec. 3, we automatically included synonymous classes and related superclasses whenever subclasses were deemed present. For instance, identifying `cow lowing` led to automatically including the superclass `cattle mooing`. A detailed overview of these automatically added classes and their relationships is provided in Tab. 6.

A.5. Final pipeline

In the last aggregation stage, we merge all crowd-sourced (Mechanical Turk) and in-house annotations so that every video is reviewed by at least three annotators. Annotators are ranked by their mean F1 score on the gold-standard clips, and the top three for each video are retained. Their votes are then combined with majority voting, producing the final label set, which achieves 0.68 macro-averaged F1 on the gold-standard data. As noted in the main paper Sec. 5, our default evaluation subset excludes samples with `background music` label, however, we provide VGGSound*er* with all samples and allow user to choose his preferred regime, if necessary.

B. Class label frequency in VGGSound*er*

Fig. 8 shows the frequency of the 40 most common class labels by modality. We observe that the label distribution appears to be very similar for visible classes and for classes that are audible and visible.

Class	**Added Class**
timpani	tympani
tympani	timpani
dog barking	dog bow-wow
dog bow-wow	dog barking
Barn swallow calling	Bird chirping, tweeting
Eagle screaming	Bird squawking
Canary calling	Bird chirping, tweeting
Mynah bird singing	Bird chirping, tweeting
Magpie calling	Bird squawking
Warbler chirping	Bird chirping, tweeting
Wood thrush calling	Bird chirping, tweeting
Goose honking	Bird squawking
Duck quacking	Bird squawking
Penguins braying	Bird squawking
Baltimore oriole calling	Bird chirping, tweeting
Crow cawing	Bird squawking
Airplane flyby	Airplane
Baby babbling	People babbling
Bull bellowing	Cattle mooing
Cow lowing	Cattle mooing
People eating noodle	People eating
People eating apple	People eating
Eating with cutlery	People eating
Bathroom ventilation fan running	Running electric fan
Striking bowling	Bowling impact

Table 6. Class mapping used to automatically add synonymous classes and superclasses.

This matches the label modality distribution in Figure 3B in the main
paper. Furthermore, we observe that the class label `male speech` is occurs more frequently than `female speech`.

C. Model evaluations and input prompts

This section provides additional details about the evaluated models, input prompts and evaluation methodology used in the zero-shot and LLM-assisted evaluations described in Sec. 5 of the paper. Specifically, we detail the prompts and methods for generating classification predictions for the models in the Gemini family [73] and for the open-source foundational models VideoLLaMA-2 [21], Unified-IO 2 [53], PandaGPT [72], and Ola [52].

C.1. Models

CAV-MAE [33] combines contrastive learning with masked data modelling to obtain strong audio-visual embeddings, used for downstream retrieval and classification tasks. We use the multi-modal CAV-MAE-Scale+ model, pretrained on AudioSet and fine-tuned on VGGSound. Following [33], unimodal and multi-modal variants use original pretrained model but we fine-tune them on VGGSound only using the respective modality.

DeepAVFusion [57] integrates complementary features from the audio and visual modalities using a deep fusion mechanism, enhancing joint processing for classification tasks. We use publicly available checkpoints for unimodal and multi-modal models pre-trained on AudioSet and we then fine-tune them on VGGSound.

AV-Siam [49] uses a two-stream network to learn joint embeddings from audio and visual data. By maximising similarity for corresponding pairs and minimising it for non-corresponding pairs, the model captures meaningful relationships between modalities. We use public checkpoints of AV-Siam pre-trained on AudioSet, to then fine-tune it on VGGSound.

Equi-AV [43] is a transformer-based model that focusses on learning invariant embedding representations through an equivariant learning approach, making it robust to input variations. Again, we fine-tune original model pre-trained on AudioSet using unimodal or multi-modal VGGSound data.

Gemini 1.5 Flash, Gemini 1.5 Pro and Gemini 2.0 Flash [73] are mixture-of-experts transformer models that process both audio and visual information. For classification, the models are prompted to output class labels from the VGGSound class list that match the input video clip, along with a caption. Unlike models trained on VGGSound, the Gemini models are assumed to be free from VGGSound-specific biases. The complete input prompts are provided in Appendix C.

VideoLLaMA-2.1-AV (VideoLLaMA-2) [21] is a multi-modal foundation model that ingests audio and visual information in two branches that independently process vision-language and audio-language data. The two branches are connected via a language model. VideoLLaMA-2 exhibits strong results on audio-visual question-answering and captioning tasks. Details about the model and prompts used are detailed in Appendix C.

Unified-IO-2 [53] is a 7B-parameter autoregressive encoder–decoder model that tokenises text, images, audio, and discrete actions into one shared sequence, enabling “any-to-any” understanding and generation.

Panda-GPT [72] augments a frozen Vicuna-13B language model with ImageBind encoders by using a single linear projection and LoRA adapters. These are trained on only 160k image-text instruction pairs. Despite this lightweight fine-tuning, the model follows instructions across six modalities (image/video, audio, text, depth, thermal, IMU) and can seamlessly compose their semantics in zero-shot settings.

Ola [52] is an omni-modal 7B LLM that progressively aligns modalities—starting with image–text, and then adding speech and finally audio-visual video. It uses local–global attention fusion, dual audio encoders (Whisper [68] + BEATs [17]) and sentence-wise streaming speech decoding. This staged training yields balanced, competitive accuracy for image QA, video QA, and speech recognition.

Motivation for LLM-assisted evaluation
In Sec. 5, we briefly mentioned standard classification strategies for foundation models, such as:

- Directly asking for a class without providing a list of available classes *(direct)*, Some models, such as VideoLLaMA-2, Unified-IO-2, and PandaGPT, were pretrained on VGGSound. For certain prompts, they return valid VGGSound classes, which makes character-level comparison feasible. However, their overall performance on VGGSound*er* is low, as most outputs are synonym classes not included in the original class set.

- Prepending a list of all available classes to the classification prompt *(zero-shot)*,

Here, we try to mitigate character-level comparison issues by prepending all 309 class names before the prompt: *"Annotate the video, explain in detail what is happening in the video. Use classes from the provided list in the captioning and also add yours."* This approach works well for closed-source foundation models but performs extremely poorly on all open-source models, most likely due to their smaller effective context window.

- Asking 309 independent questions, one per class, for every sample *(multi-prompt)*.

This strategy avoids the context length limitation. Instead of including all class names at once, we ask 309 questions per sample, each with the prompt: *"Do you see or hear the following class 'class' in the video? Answer only with yes or no."* While this pipeline yields higher classification scores, it is computationally expensive and still fails to fully capture the video understanding capabilities of most open-source foundation models.

In conclusion, all the above strategies yield low performance (e.g., low F1 scores) and fail to reliably capture a model's video understanding. To address this, we adopt a hybrid approach: we use the *zero-shot* strategy for closed-source models and introduce an *LLM-assisted evaluation* protocol for open-source foundation models.

Gemini models The Gemini models can handle long prompts very well. Thus, to generate classification predictions with models from the Gemini family, we used a zero-shot evaluation protocol. Specifically, we provided the models with an input prompt, a list of all class names in VGGSound separated with commas, and an input video file. We used the following text template:

```
{CLASSES}
{VIDEO}
Annotate the video, explain in detail what is happening in the video.  Use classes from
the provided list in the captioning and also add yours.
```

LLM-assisted evaluation We evaluated all other foundation models using LLM-assisted evaluation.
Building on similar approaches, [3] we employ Qwen3 [82] (32B quantised to 8 bits) as our LLM for evaluating the alignment between model-generated outputs and the ground truth.
Specifically, for each sample, the open-source foundation models are asked the following questions depending on the input modality:

```
A:
What actions are being performed in this audio, explain all sounds and actions in the
audio?  Please provide a short answer.
```

```
V/AV:
What actions are being performed in this video, explain all sounds and actions in the
video?  Please provide a short answer.
```

The generated answer (video/audio captioning text) and the target labels (list of classes separated with comma) are then both supplied to the Qwen3 evaluator that receives the following system prompt.

[3]We found the VideoLLaMA-2 appendix [21], PointLLM appendix [81], and the Unified-IO 2 code base https://github.com/allenai/unified-io-2/blob/502ac4d81239f82c891a9f412b000c3c8d4e2946/t5x/examples/unified_io/data/prompt_dict.py to be very useful.

LLM system prompt

```
You are an intelligent chatbot designed for evaluating the correctness of generative outputs
 for classification pairs. Your task is to compare the predicted answer with the correct
answer and determine if they match meaningfully. Here's how you can accomplish the task:

- Focus on the meaningful match between the predicted answer and the correct answer.
- Consider synonyms or paraphrases as valid matches.
- Evaluate the correctness of the prediction compared to the answer.
- The correct answer, might contain multiple classes. Treat them independently and evaluate
the correctness of all them w.r.t predicted answer.

Provide your evaluation only as a yes/no and score where the score is an integer value
between 0 and 5, with 5 indicating the highest meaningful match.

Please generate the response in the form of a Python dictionary string where names of
classes are keys and values are dictionary strings with keys 'pred' and 'score', where value
 of 'pred' is a string of 'yes' or 'no' and value of 'score' is in INTEGER, not STRING.

DO NOT PROVIDE ANY OTHER OUTPUT TEXT OR EXPLANATION. Only provide the Python dictionary
string. For example, your response should look like this:

{"male speech, man speaking": {"pred": "yes", "score": 4}, "playing banjo": {"pred": "no", "
score": 0}}

Example 1.

<Question>
Identify the main sounds present in the given audio clip with a few words.

<Correct Answers>
["cat caterwauling", "cat meowing"]

<Predicted Answer>
The main sounds present in the given audio clip are:

1. A ticking sound, possibly from a clock or timer.
2. A mechanical sound, which could be from a machine or device.
3. A human voice, which is speaking in the background.

Output: {"cat caterwauling": {"pred": "no", "score": 0}, "cat meowing": {"pred": "no", "
score": 0}}

Example 2.

<Question>
What actions are being performed in this audio, explain all sounds and actions in the audio?
 Please provide a short answer.

<Correct Answers>
["cuckoo bird calling", "mynah bird singing", "bird chirping, tweeting"]

<Predicted Answer>
The audio features a cuckoo bird calling in the distance and some chirping and tweeting from
 smaller birds.

Output: {"cuckoo bird calling": {"pred": "yes", "score": 5}, "mynah bird singing": {"pred":
"no", "score": 0}, "bird chirping, tweeting": {"pred": "no", "score": 5}}
```

```
Example 3.

<Question>
What actions are being performed in this video, explain all sounds and actions in the video?
 Please provide a short answer.

<Correct Answers>
["male speech, man speaking", "playing hammond organ"]

<Predicted Answer>
The video shows a man who is playing regular piano and speaking with someone.

Output: {"male speech, man speaking": {"pred": "yes", "score": 5}, "playing hammond organ":
{"pred": "yes", "score": 3}}
```

User message template

```
<Question>
{QUESTION}

<Correct Answers>
{ANSWERS}

<Predicted Answer>
{CAPTION}
```

Qwen3 then outputs a Python-formatted dictionary mapping for each target class. The dictionary contains binary "pred" decision (yes/no) and a nuanced confidence score (0–5), accommodating synonymy and paraphrasing.

```
{"male speech, man speaking": {"pred": "yes", "score": 5}, "playing hammond organ":
{"pred": "yes", "score": 3}}
```

This flexible scoring relaxes the strict label matching, yielding richer, semantically-aware assessments that better reflect human judgment and are align with recent "LLM-as-judge" [34] paradigms that have demonstrated enhanced correlation with human evaluators across a diverse set of tasks and domains.

D. Additional quantitative analysis

This appendix extends our quantitative analyses presented in Sec. 5.1 of the main paper, providing further insights into model behaviour on both VGGSound and the newly introduced VGGSound*er* benchmark.

D.1. Model performance on VGGSound

We present the classification performance of state-of-the-art models on the original VGGSound test data in Fig. 9. We observe that the multi-label hit accuracy on VGGSound*er* reported in Tab. 2 in the main paper significantly raises the performance across all models. This suggests that the models predict classes that were not present in the original VGGSound labelling, despite those being correct.
Fig. 10 further compares the averaged F1-scores between the "Foundation model" and "Embedding model" families, highlighting that evaluations on the original VGGSound consistently underestimate model performance across all modalities when compared to evaluations on VGGSound*er*.

	Accuracy ↑			$\mu\downarrow$		
Models	a	v	av	μ_A	μ_V	$\mu_{A\cap V}$
CAV-MAE	59.05	45.57	65.08	4.71	4.84	0.67
DeepAVFusion	40.82	27.24	53.10	4.18	3.17	0.07
Equi-AV	46.68	24.84	50.08	6.91	5.51	0.98
AV-Siam	56.91	47.27	55.25	13.17	8.92	3.92
Gemini 1.5 Flash	0.31	22.12	23.60	1.51	4.17	0.09
Gemini 1.5 Pro	1.29	25.77	21.31	1.62	5.41	0.24
Gemini 2.0 Flash	5.70	20.29	19.39	2.50	4.77	0.63
VideoLLaMA 2	27.98	17.01	21.46	11.16	2.85	1.42
Unified-IO 2	32.28	20.24	52.40	4.88	3.42	0.87
PandaGPT	5.20	7.65	8.95	4.51	4.48	0.94
OLA	10.71	8.63	14.29	7.61	4.05	0.71

Figure 9. **Performance of state-of-the-art models on VGGSound.** We report top-1 classification accuracy for different input modalities (audio *A*, visual *V*, and audio and visual information *AV*). μ is modality confusion metric defined in Sec. 5

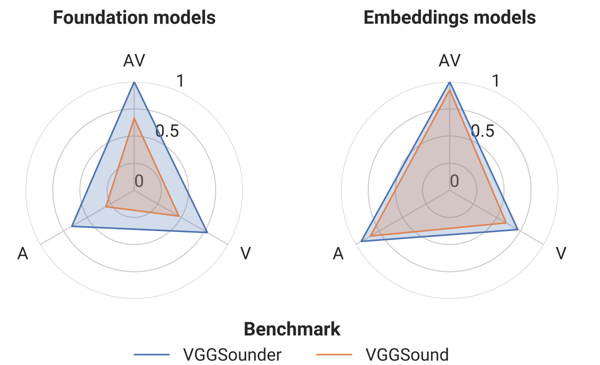


Figure 10. **Performance of state-of-the-art families on VGGSound compared to VGGSound*er*.** Radar plots illustrate the average F1-scores across modalities for two model families: "Foundation models" and "Embedding models" (Tab. 2).

D.2. Co-occurrence matrix on VGGSound

To further illustrate the issue of class overlap described in Sec. 3 of our paper, we include an analysis of class co-occurrences in predictions by the CAV-MAE [33] model in Fig. 11. Specifically, we provide a co-occurrence matrix highlighting frequent simultaneous predictions of certain classes. Notably, labels such as `playing drum kit` and `playing bass drum` are frequently predicted together, as they are not mutually exclusive. This analysis supports our identification of overlapping classes as a key limitation in the original VGGSound annotations and demonstrates the need for explicitly multi-label approaches in video classification tasks.

D.3. Classification results for other k

Tab. 7 extends the evaluation presented in the main paper by showing multi-label video classification results on VGGSound*er* for varying numbers of top-k predictions, specifically for $k \in \{3, 5, 10\}$. These additional results offer deeper insights into how model performance changes with an increasing number of predictions. Specifically, one can notice the opposite behaviour between the F1-score (goes down with k) and the Hit score (increases with k).

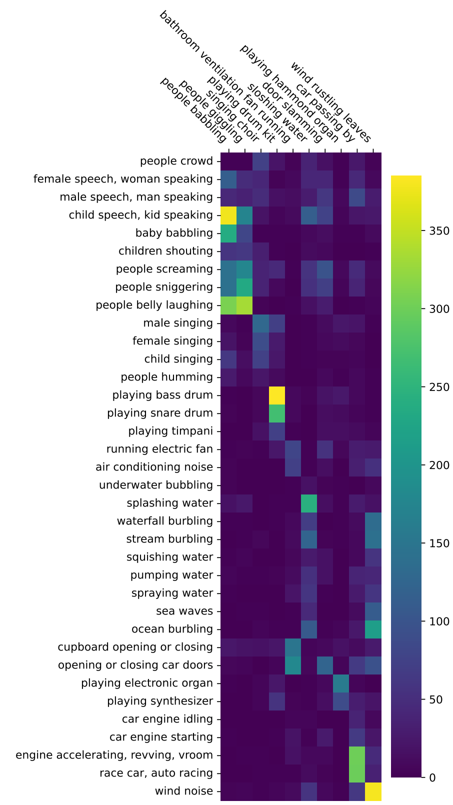


Figure 11. Co-occurrence counts among a subset of VGGSound classes, estimated on the VGGSound test set by the CAV-MAE model. Each cell indicates how frequently two classes appear together, highlighting labels that share overlapping acoustic cues (e.g., `playing drum kit` and `playing bass drum`). Best viewed zoomed in on a screen.

D.4. Performance on subsets of VGGSound*er*

To comprehensively evaluate model robustness in the presence of common confounders highlighted in Sec. 3 (i.e. meta-labels: *background music*, *static images*, and *voice over*), we present additional evaluations on distinct subsets of VGGSound*er*. Specifically, Tabs. 8 to 13 display the performance of state-of-the-art models on subsets only containing or fully excluding the meta-labels. These analyses confirm the importance of accounting for modality-specific and meta-label influences.

k	Model	Subset Accuracy ↑			$\mathbf{F_1}$ ↑			Hit ↑		
		a	v	av	a	v	av	a	v	AV
3	CAV-MAE	0.99	0.56	1.09	39.10	35.58	42.92	81.18	72.14	82.55
	DeepAVFusion	0.22	0.08	0.68	28.07	22.43	37.36	65.15	50.86	74.75
	Equi-AV	0.55	0.24	0.34	33.50	22.78	34.06	73.82	48.13	70.78
	AV-Siam	0.83	0.77	0.74	37.36	37.05	40.91	79.32	73.21	79.83
5	CAV-MAE	0.04	0.04	0.03	35.14	30.79	36.00	87.04	78.73	87.64
	DeepAVFusion	0.00	0.00	0.00	24.91	19.48	31.06	72.24	58.48	80.38
	Equi-AV	0.01	0.01	0.00	30.06	20.44	28.62	80.64	55.67	77.09
	AV-Siam	0.02	0.04	0.02	33.13	31.88	34.67	84.81	79.57	85.53
10	CAV-MAE	0.00	0.00	0.00	25.61	21.66	24.36	91.64	85.27	92.01
	DeepAVFusion	0.00	0.00	0.00	18.36	14.23	21.06	80.35	67.87	85.70
	Equi-AV	0.00	0.00	0.00	22.39	15.24	19.84	87.41	65.49	83.77
	AV-Siam	0.00	0.00	0.00	24.15	22.12	23.94	90.11	85.86	90.99

Table 7. **Audio-visual video classification results on VGGSound*er* for $k \in \{3, 5, 10\}$.** The table is vertically grouped by k. Within each block, the four models are compared across the three metrics and input modalities.

Impact of background music

Model	Subset Accuracy ↑			$\mathbf{F_1}$ ↑					Hit ↑			μ ↓		
	a	v	av	a	v	av	$a(A\neg V)$	$v(V\neg A)$	a	v	av	μ_A	μ_V	$\mu_{A\cap V}$
CAV-MAE	10.80	19.17	23.84	31.03	31.29	38.60	17.83	22.26	55.96	44.57	54.02	4.26	6.92	0.87
DeepAVFusion	8.15	9.48	20.66	21.66	16.40	33.05	12.68	8.27	39.02	23.32	46.18	2.52	3.32	0.09
Equi-AV	9.11	10.11	19.70	25.33	17.86	32.13	14.75	11.52	45.67	25.44	44.98	5.39	6.09	1.09
AV-Siam	10.46	18.50	21.31	29.55	31.00	34.31	16.53	22.65	53.28	44.15	48.02	10.18	8.83	3.70
Gemini 1.5 Flash	1.15	13.91	14.75	13.31	34.57	38.36	11.49	22.10	30.47	44.24	53.50	11.53	3.65	0.78
Gemini 1.5 Pro	1.90	20.84	20.75	17.40	46.04	47.93	13.68	27.50	33.65	62.36	67.51	3.91	3.96	0.61
Gemini 2.0 Flash	1.08	11.32	10.44	11.33	32.14	32.97	9.84	21.84	18.62	39.81	43.11	2.31	4.39	0.83
VideoLLaMA 2	11.24	18.63	22.66	36.43	43.18	46.81	23.97	33.41	53.86	43.65	48.37	15.27	5.35	2.83
Unified-IO 2	9.11	13.45	24.49	28.90	29.07	44.69	20.77	22.97	42.55	28.82	52.24	5.92	5.87	1.57
PandaGPT	1.86	4.64	5.79	12.75	17.64	18.11	8.65	16.09	14.96	15.50	15.18	6.87	5.70	2.22
Ola	8.53	9.77	19.18	35.87	25.44	43.61	29.25	17.17	44.35	23.43	44.85	11.74	7.39	1.70

Table 8. **Audio-visual video classification results on the subset of VGGSound*er* that is labelled as containing *background music*.** Similar to Table 1 in the main paper, we report multi-label classification metrics (subset accuracy, F_1-score, Hit accuracy, modality confusion (μ)) for audio- $a(A)$, visual - $v(V)$, audio-visual - $av(AV)$, audio-only - $a(A\neg V)$ and video-only - $v(V\neg A)$ inputs.

Model	Subset Accuracy ↑			$\mathbf{F_1}$ ↑					Hit ↑			μ ↓		
	a	v	av	a	v	av	$a(A\neg V)$	$v(V\neg A)$	a	v	av	μ_A	μ_V	$\mu_{A\cap V}$
CAV-MAE	13.19	19.23	24.49	34.46	34.91	42.62	13.94	19.00	62.29	53.44	64.17	3.58	6.43	0.77
DeepAVFusion	10.19	11.10	21.53	25.31	21.29	37.35	10.37	10.55	45.77	32.61	56.27	3.74	3.93	0.17
Equi-AV	11.60	10.52	20.00	29.39	20.42	34.69	12.55	10.65	53.12	31.26	52.24	6.97	7.13	1.38
AV-Siam	12.79	19.75	22.83	33.30	35.41	39.43	12.90	18.21	60.19	54.20	59.36	9.36	8.80	3.58
Gemini 1.5 Flash	1.78	14.44	16.44	14.49	36.98	42.52	15.61	21.61	32.73	47.36	59.10	10.22	4.25	0.77
Gemini 1.5 Pro	3.05	20.86	22.53	19.26	49.73	53.74	17.73	22.90	35.03	69.23	75.42	2.09	4.85	0.57
Gemini 2.0 Flash	1.85	12.54	12.69	11.80	34.08	36.45	6.19	18.90	18.51	43.83	47.72	2.39	5.43	1.00
VideoLLaMA 2	12.86	19.85	24.47	38.87	47.82	52.35	20.34	28.08	58.91	52.02	59.80	12.72	5.46	2.95
Unified-IO 2	11.94	11.56	25.61	35.31	27.92	48.89	21.38	16.53	54.39	31.05	65.11	8.70	5.16	1.79
PandaGPT	3.19	4.19	5.46	18.73	18.56	20.85	16.82	14.40	21.08	17.01	18.82	7.59	5.90	2.47
Ola	14.11	8.69	18.19	47.70	24.85	46.48	40.44	13.45	59.05	24.57	51.51	15.47	6.80	2.49

Table 9. **Audio-visual video classification results on the subset of VGGSound*er* that is labelled as *not* containing *background music***

A side-by-side inspection of the two subsets (Tab.8 vs. Tab.9) reveals several interesting points.

(i) Universal but modality-specific gains. Every method improves in terms of F_1 and *Hit* scores when the soundtrack is removed, that is especially clear for the *audio* input modality: for the embedding family we register jumps of up to $+5\%$ in F_1 for both audio and *visual* inputs. Consequently, joint audio–visual inputs rise in performance only slightly ($+3$–5%).

(ii) Same trend for foundation models, but with caveats. Foundation checkpoints with a meaningful audio encoder echo the pattern (Unified-IO2 $+7\%$, Ola $+12\%$); in contrast, the Gemini family remains audio-weak, suggesting that their publicly released models rely heavily on vision.

(iii) Intuition. Background music tends to mask class-specific foreground sounds; once that mask is removed the audio encoder can finally "hear" discriminative cues, whereas vision—being agnostic to the soundtrack—is affected only by the changed clip mix. With noisy audio, every model relies more on the *V* modality as a safety net, which explains why their baseline performance remains respectable despite the severe audio corruption.

Altogether, these observations confirm that background music constitutes a hard confounder, forcing models to rely on vision.

Impact of static images

	Subset Accuracy ↑			$\mathbf{F_1}$ ↑					Hit ↑			μ ↓		
Model	*a*	*v*	*av*	*a*	*v*	*av*	$a(A\neg V)$	$v(V\neg A)$	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A\cap V}$
CAV-MAE	22.13	19.48	27.42	38.24	27.21	37.62	35.18	15.03	61.20	34.74	47.28	4.22	6.85	0.35
DeepAVFusion	15.98	10.96	23.20	28.65	15.80	31.50	26.43	6.33	45.89	20.21	39.59	4.24	3.87	0.00
Equi-AV	19.00	10.39	22.50	32.59	14.24	31.33	30.14	9.25	52.15	18.18	39.37	5.62	4.22	0.35
AV-Siam	22.04	19.81	24.25	37.46	28.10	34.27	33.61	13.87	59.95	35.88	43.06	10.72	7.91	2.64
Gemini 1.5 Flash	1.43	13.47	15.64	9.35	29.51	33.27	8.15	18.63	20.25	30.19	40.60	10.90	4.57	0.88
Gemini 1.5 Pro	2.33	22.08	23.37	14.32	42.50	44.89	12.52	24.44	24.19	51.30	57.47	3.87	5.45	0.53
Gemini 2.0 Flash	3.85	10.88	13.88	13.54	26.91	31.64	12.82	13.75	19.53	29.87	38.31	1.93	3.87	0.88
VideoLLaMA 2	19.00	18.18	23.73	42.36	37.90	43.22	39.41	27.54	56.45	32.31	40.60	15.47	5.10	2.64
Unified-IO 2	17.92	9.58	28.47	35.87	22.12	45.95	33.45	15.93	47.49	18.34	47.80	7.38	3.34	1.05
PandaGPT	3.32	4.87	5.27	14.15	13.71	15.32	12.75	11.01	14.78	11.36	11.78	8.96	5.62	2.11
Ola	14.87	8.60	18.28	37.80	19.72	38.88	33.53	10.53	42.29	15.58	34.97	15.99	5.27	1.41

Table 10. **Audio-visual video classification results on the subset of VGGSound*er* that is labelled as containing *static images***

	Subset Accuracy ↑			$\mathbf{F_1}$ ↑					Hit ↑			μ ↓		
Model	*a*	*v*	*av*	*a*	*v*	*av*	$a(A\neg V)$	$v(V\neg A)$	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A\cap V}$
CAV-MAE	11.98	19.21	24.24	33.49	34.60	42.13	12.31	20.02	61.05	52.70	63.11	3.67	6.50	0.81
DeepAVFusion	9.30	10.81	21.30	24.31	20.67	36.84	8.96	10.33	44.34	31.48	55.19	3.50	3.82	0.16
Equi-AV	10.49	10.45	19.84	28.33	20.23	34.39	10.91	10.94	51.64	30.81	51.51	6.75	7.08	1.37
AV-Siam	11.57	19.52	22.49	32.24	34.95	38.76	11.26	19.51	58.76	53.23	58.06	9.44	8.85	3.64
Gemini 1.5 Flash	1.68	14.39	16.17	14.62	36.82	42.14	15.43	21.92	33.25	47.59	58.92	10.42	4.13	0.77
Gemini 1.5 Pro	2.87	20.80	22.17	19.22	49.36	53.07	17.24	23.81	35.60	68.82	74.80	2.34	4.66	0.58
Gemini 2.0 Flash	1.53	12.40	12.23	11.58	34.02	36.02	6.43	19.91	18.45	43.75	47.31	2.39	5.31	0.97
VideoLLaMA 2	12.04	19.70	24.18	38.13	47.41	51.78	18.90	29.21	58.05	51.42	58.61	13.06	5.46	2.94
Unified-IO 2	10.88	12.00	25.28	33.99	28.31	48.33	19.64	17.98	52.46	31.24	63.57	8.26	5.37	1.78
PandaGPT	2.91	4.24	5.53	17.83	18.58	20.60	15.01	15.00	20.29	17.00	18.49	7.40	5.87	2.44
Ola	12.88	8.89	18.36	46.03	25.12	46.27	38.29	14.45	57.30	24.78	51.05	14.78	6.97	2.40

Table 11. **Audio-visual video classification results on the subset of VGGSound*er* that is labelled as *not* containing *static images***

A side-by-side inspection of the "static image" split (Tab.,10) and its complement (Tab.,11) shows four salient effects.

(i) Vision takes the hit, audio steps up. Across the classic embedding models, the *visual* branch loses on average 6–7% absolute in F_1, while using *audio* inputs results in gains $+3$–6%. The same holds true for the joint audio-visual. Hit scores

mirror the trend: Hit for visual inputs plunges by up to 20%, whereas Hit for audio inputs remains flat or edges upward for most of the models.

(ii) Foundation models react unevenly. VideoLLaMA2 loses 8% on vision yet gains 3% on audio—whereas the vision-centric Gemini family suffers a broad decline, unable to compensate for the poor visual signal.

(iii) Intuition. Static clips provide far less discriminative visual evidence than genuine video, reducing motion and viewpoint cues. The audio track, in contrast, is untouched; consequently, models shift their reliance toward the acoustic channel, explaining the systematic audio gain and the parallel vision loss.

(iv) Modality-confusion drifts upward. With vision degraded, many architectures become more uncertain about which modality to trust; a few (AV-Siam) even over-correct, raising μ_A by $+1.8$ while slightly easing μ_V.

In sum, static imagery acts as the visual analogue to background music: it removes discriminative content in one modality (vision) and forces models to lean on the other (audio), exposing how well a model can rebalance modalities.

Impact of voice-over narration

	Subset Accuracy ↑			$\mathbf{F_1}$ ↑					**Hit** ↑			$\mu\downarrow$		
Model	*a*	*v*	*av*	*a*	*v*	*av*	$a(A\neg V)$	$v(V\neg A)$	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A\cap V}$
CAV-MAE	2.78	14.34	17.38	26.68	28.36	35.37	11.19	25.50	51.65	43.55	53.62	4.21	7.06	0.65
DeepAVFusion	2.02	9.41	15.33	16.79	17.78	29.13	6.51	13.99	32.50	27.34	44.23	2.69	3.79	0.18
Equi-AV	3.51	7.75	14.53	22.45	15.44	28.14	9.62	10.74	43.45	23.71	42.65	7.47	6.35	1.07
AV-Siam	2.73	13.94	15.66	25.72	28.51	31.11	9.56	26.85	49.79	43.78	47.15	10.97	8.96	3.50
Gemini 1.5 Flash	5.26	9.43	10.85	29.43	30.61	34.01	27.55	23.61	63.71	40.02	48.28	22.18	4.57	1.36
Gemini 1.5 Pro	7.73	17.70	16.07	34.58	46.22	50.84	29.26	27.76	68.30	65.70	75.50	4.15	4.27	0.95
Gemini 2.0 Flash	0.72	7.40	8.36	11.87	27.47	29.85	6.10	21.62	19.95	36.38	40.57	2.79	5.81	1.30
VideoLLaMA 2	6.34	16.08	19.40	34.98	42.76	47.72	21.74	36.14	54.43	47.60	54.69	13.58	5.40	2.43
Unified-IO 2	4.69	9.08	18.15	29.81	24.80	40.94	23.17	21.65	48.61	27.13	53.44	9.49	5.16	2.02
PandaGPT	3.66	4.28	4.86	20.96	18.58	18.92	19.07	17.73	26.75	17.47	18.03	9.79	6.52	3.32
Ola	15.46	7.06	16.13	54.14	23.48	48.53	47.31	16.08	76.08	24.64	57.53	16.19	4.98	2.85

Table 12. **Audio-visual video classification results on the subset of VGGSound*er* that is labelled as containing *voice over* narrations**

	Subset Accuracy ↑			$\mathbf{F_1}$ ↑					**Hit** ↑			$\mu\downarrow$		
Model	*a*	*v*	*av*	*a*	*v*	*av*	$a(A\neg V)$	$v(V\neg A)$	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A\cap V}$
CAV-MAE	14.18	19.92	25.38	34.92	35.19	42.94	15.67	18.90	62.44	53.10	63.70	3.62	6.44	0.81
DeepAVFusion	10.94	11.02	22.26	25.84	20.89	37.77	11.84	9.55	46.21	31.52	56.03	3.65	3.83	0.15
Equi-AV	12.23	10.83	20.73	29.58	20.67	35.18	13.80	10.84	52.88	31.18	52.19	6.59	7.04	1.37
AV-Siam	13.74	20.34	23.56	33.66	35.59	39.70	14.66	18.09	60.17	53.70	58.90	9.29	8.79	3.61
Gemini 1.5 Flash	1.14	15.06	16.91	12.18	37.42	42.94	12.38	21.45	27.71	47.79	59.56	8.76	4.09	0.69
Gemini 1.5 Pro	2.11	21.31	23.11	16.42	49.55	53.10	14.29	23.32	29.87	68.38	73.86	2.15	4.75	0.53
Gemini 2.0 Flash	1.84	13.04	12.87	11.68	34.66	36.73	7.38	19.23	18.32	44.10	47.84	2.31	5.17	0.92
VideoLLaMA 2	13.45	20.15	24.84	38.95	47.72	52.06	21.18	28.23	58.44	50.99	58.30	13.10	5.45	3.00
Unified-IO 2	12.37	12.29	26.46	34.79	28.58	49.28	20.85	17.37	52.60	31.17	64.26	8.05	5.30	1.71
PandaGPT	2.83	4.27	5.61	17.05	18.39	20.65	13.90	14.34	18.89	16.65	18.23	7.14	5.77	2.30
Ola	12.67	9.14	18.68	44.05	25.16	45.64	35.76	13.96	53.30	24.33	49.35	14.64	7.18	2.29

Table 13. **Audio-visual video classification results on the subset of VGGSound*er* that is labelled as *not* containing *voice over* narrations**

Considering the "voice-over" split (Tab.,12) with its complement (Tab.,13) exposes a two–way story that depends on how each model treats speech.

(i) Embedding models are confused. For all four embedding models, results when using *audio* inputs jump by roughly $+5$–10% in F_1 when the narration track is removed, and Hit for audio climbs in parallel. This confirms that spoken commen-

tary *masks* class-specific sounds. However, once silenced, the models can finally "hear" the underlying events, again, similar to the background music meta-class.

(ii) Reduced performance for speech-centric foundation models. Gemini 1.5 and PandaGPT fail when narration disappears: F_1 for audio inputs plunges by around -17% and Hit for audio inputs drops by up to 39%. Our intuition is, that these models exploit the speech content as a shortcut.

(iii) Middle ground for broad-coverage LMMs. Unified-IO 2 and VideoLLaMA-2 are between the two extremes: they register a moderate audio lift (+4–5%) and a small visual bump (+1–2%), yielding a , $+1$–, 8% improvement in terms of F_1 score for audio-visual inputs. We hypothesise that their balanced training helps them survive the removal of speech while still profiting from the clearer acoustic scene.

(iv) Modality-confusion μ reacts in both directions. For speech-reliant models, clearer acoustics *reduce* uncertainty, whereas for event-focused encoders (trained on VGGSound) it slightly *rises* because the freshly revealing audio now dominates the fusion gate.

Taken together, voice-over narration acts as the mirror image of background music: it can be a *helpful shortcut* for speech-aware foundation models, yet a *destructive mask* for sound classifiers trained on VGGSound.

Confounder–free subset

	Subset Accuracy ↑			$\mathbf{F_1}$ ↑					Hit ↑			μ ↓		
Model	*a*	*v*	*av*	*a*	*v*	*av*	$a(A\neg V)$	$v(V\neg A)$	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A\cap V}$
CAV-MAE	13.51	19.53	24.90	34.80	35.59	43.21	11.71	18.99	62.87	54.68	65.26	3.52	6.42	0.80
DeepAVFusion	10.56	11.04	21.84	25.86	21.50	38.01	8.98	10.27	46.74	33.04	57.43	3.86	3.90	0.16
Equi-AV	11.75	10.73	20.19	29.57	20.97	35.17	10.67	10.77	53.42	32.22	53.11	6.85	7.31	1.42
AV-Siam	13.02	20.08	23.21	33.58	36.03	39.99	10.82	17.60	60.67	55.35	60.39	9.29	8.86	3.65
Gemini 1.5 Flash	1.27	14.93	16.90	12.86	37.67	43.46	14.05	21.55	29.32	48.58	60.70	8.79	4.13	0.71
Gemini 1.5 Pro	2.35	20.86	22.97	17.26	50.01	53.89	15.69	22.27	31.43	69.78	75.44	1.97	4.80	0.55
Gemini 2.0 Flash	1.85	13.01	12.93	11.73	34.83	37.09	5.89	18.93	18.38	44.86	48.56	2.39	5.36	0.95
VideoLLaMA 2	13.02	20.08	24.94	38.87	48.36	52.84	17.78	27.52	59.11	52.65	60.42	12.61	5.37	2.97
Unified-IO 2	11.94	11.76	25.96	35.18	28.25	49.42	18.84	16.01	54.07	31.62	66.07	8.41	5.24	1.75
PandaGPT	3.00	4.09	5.43	18.21	18.57	21.08	16.00	14.68	20.34	17.06	18.98	7.17	5.76	2.34
Ola	13.38	8.86	18.24	46.34	25.07	46.10	38.24	13.13	56.92	24.78	50.88	15.23	7.08	2.46

Table 14. **Audio-visual video classification results on the subset of VGGSound*er* that is labelled as *not* containing *background music, static images, or voice over* narrations**

The split that *simultaneously* excludes background music, static images, and voice-over narration (Tab.,14) serves as an upper-bound reference and reveals how each system performs when no major nuisance factor is present.

Removing *all* three meta-classes unlocks the highest scores yet observed and sharpens modality agreement.

D.5. Ablation study for additional labels in VGGSound*er*

In this section, we conduct an ablation study to quantify the benefits introduced by different components of our annotation pipeline described in Section 3. Specifically, we compare model performance on three variants of ground-truth labels: (a) Original VGGSound labels extended only with automatically added synonymous and superclass labels, (b) Original VGGSound labels extended exclusively with human annotations, (c) Original VGGSound labels extended comprehensively with both automatically added labels and human annotations (VGGSound*er*).
Detailed performance results in Tab. 16 and Tab. 17 demonstrate a consistent improvement across HIT and F1 metrics when employing the complete set of annotations (scenario c). This clearly illustrates the reduction in false-positive identifications and improved accuracy achieved through our annotation pipeline. This again highlights the importance of combining automated processes with thorough human verification in creating robust benchmarks for evaluating audio-visual models.

Model	Subset Accuracy ↑			$\mathbf{F_1}$ ↑			Hit ↑			$\mu \downarrow$		
	a	*v*	*av*	*a*	*v*	*av*	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A \cap V}$
Gemini 1.5 Flash	0.31	22.12	23.60	1.71	33.15	35.94	2.98	31.83	41.23	1.51	4.17	0.09
Gemini 1.5 Pro	1.29	25.77	21.31	4.43	36.41	35.62	6.11	41.72	45.70	1.62	5.41	0.24
Gemini 2.0 Flash	5.70	20.29	19.39	9.95	32.34	33.91	9.49	30.55	35.31	2.50	4.77	0.63
VideoLLaMA 2	27.98	17.01	21.46	41.32	31.46	36.80	30.05	22.72	27.90	11.16	2.85	1.42
Unified-IO 2	32.28	20.24	52.40	43.71	33.84	64.06	33.71	22.84	54.20	4.88	3.42	0.87
PandaGPT	5.20	7.65	8.95	12.68	16.83	19.55	8.54	11.23	13.30	4.51	4.48	0.94
Ola	10.71	8.63	14.29	23.33	17.81	28.86	18.06	10.95	22.41	7.61	4.05	0.71

Table 15. **Audio-visual video classification results on VGGSound** inputs.

Model	Subset Accuracy ↑			$\mathbf{F_1}$ ↑			Hit ↑			$\mu \downarrow$		
	a	*v*	*av*	*a*	*v*	*av*	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A \cap V}$
Gemini 1.5 Flash	1.67	14.49	16.42	14.42	36.72	41.93	32.25	46.43	57.59	10.48	4.20	0.80
Gemini 1.5 Pro	2.86	21.32	22.59	19.17	49.15	52.85	34.72	67.25	73.33	2.43	4.76	0.58
Gemini 2.0 Flash	1.83	12.83	12.66	11.75	33.98	36.00	18.29	42.72	46.44	2.35	5.34	0.95
VideoLLaMA 2	12.71	19.40	23.98	38.37	47.26	51.59	56.94	50.22	57.36	13.04	5.48	2.87
Unified-IO 2	11.41	11.65	25.99	34.17	28.37	48.54	51.44	30.52	62.39	8.15	5.32	1.74
PandaGPT	2.93	4.27	5.41	17.69	18.55	20.53	19.68	16.53	18.04	7.40	5.81	2.39
OLA	13.32	8.75	18.44	45.89	25.22	46.36	55.76	24.28	50.02	14.59	6.96	2.34

Table 16. **Audio-visual video classification results on VGGSound + human annotations**.

Model	Subset Accuracy ↑			$\mathbf{F_1}$ ↑			Hit ↑			$\mu \downarrow$		
	a	*v*	*av*	*a*	*v*	*av*	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A \cap V}$
Gemini 1.5 Flash	1.66	14.35	16.15	14.26	36.58	41.84	32.29	46.83	58.14	10.44	4.16	0.77
Gemini 1.5 Pro	2.83	20.85	22.22	18.90	49.12	52.80	34.76	68.05	74.07	2.40	4.69	0.58
Gemini 2.0 Flash	1.70	12.33	12.30	11.70	33.76	35.87	18.53	43.14	46.93	2.37	5.25	0.97
VideoLLaMA 2	12.55	19.64	24.17	38.40	47.11	51.51	57.93	50.57	57.85	13.16	5.44	2.93
Unified-IO 2	11.39	11.89	25.42	34.11	28.10	48.25	52.09	30.66	62.91	8.23	5.28	1.75
PandaGPT	2.94	4.27	5.52	17.61	18.42	20.43	19.89	16.75	18.20	7.47	5.86	2.42
OLA	13.03	8.88	18.36	45.55	24.94	46.04	56.20	24.38	50.37	14.83	6.91	2.36

Table 17. **Audio-visual video classification results on VGGSound + human annotations + automatically added labels**