TVLT: Textless Vision-Language Transformer

Zineng Tang* **Jaemin Cho*** **Yixin Nie*** **Mohit Bansal**
UNC Chapel Hill
{terran, jmincho, yixin1, mbansal}@cs.unc.edu

Abstract

In this work, we present the Textless Vision-Language Transformer (TVLT), where homogeneous transformer blocks take raw visual and audio inputs for vision-and-language representation learning with minimal modality-specific design, and do not use text-specific modules such as tokenization or automatic speech recognition (ASR). TVLT is trained by reconstructing masked patches of continuous video frames and audio spectrograms (masked autoencoding) and contrastive modeling to align video and audio. TVLT attains performance comparable to its text-based counterpart on various multimodal tasks, such as visual question answering, image retrieval, video retrieval, and multimodal sentiment analysis, with 28x faster inference speed and only 1/3 of the parameters. Our findings suggest the possibility of learning compact and efficient visual-linguistic representations from low-level visual and audio signals without assuming the prior existence of text.[1]

1 Introduction

Humans perceive and learn the external world through signals from multiple modalities. To embody such human learning in machines, substantial research efforts are dedicated to developing vision-and-language (VL) models that can understand the joint semantics between visual and linguistic modalities and solve tasks such as visual question answering [4]. Although most such VL models use written language rather than spoken language as the main verbal communication channel, the default communication modality among humans has been speech, since circa 100,000 BCE [78]. Written language is relatively recent; cuneiform script, the earliest writing system, was developed circa 3,200 BCE [65]. Moreover, we have witnessed an increasing usage of AI models in real-world products such as virtual assistants and smart speakers [40], where perception-level signals such as video and audio are the natural form of input. Intuitively, direct modeling of such signals will potentially yield more compact and efficient representations.

Transformers [81] have recently achieved great success in vision-language representation learning [76; 10; 48; 74; 87; 86] by using text-based modules [15] on text-annotated images or videos. However, it is non-trivial to learn VL representations using transformers that take only low-level visual and acoustic inputs without the prior existence of written language. The challenge lies in the difference between text and acoustic signals; text is discrete and dense in information, while acoustic signals are continuous and sparse in information [26; 7]. Therefore, modality-specific architectures have been used to model data from different modalities. It is only recently that researchers started using modality-agnostic transformer architecture to learn representations of different unimodal [17; 19; 8], vision-text [32; 54], or vision-audio-text [2] data. However, to the best of our knowledge, no previous work has explored a single homogeneous (modality-agnostic) minimalist transformer that learns visual-linguistic representations directly from visual and acoustic input at the perception level (without relying on text), and also makes the textless VL model more compact and efficient than the existing text-based VL models (see Sec. 2 for details).

*equal contribution

[1] Our code and checkpoints are available at: https://github.com/zinengtang/TVLT

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

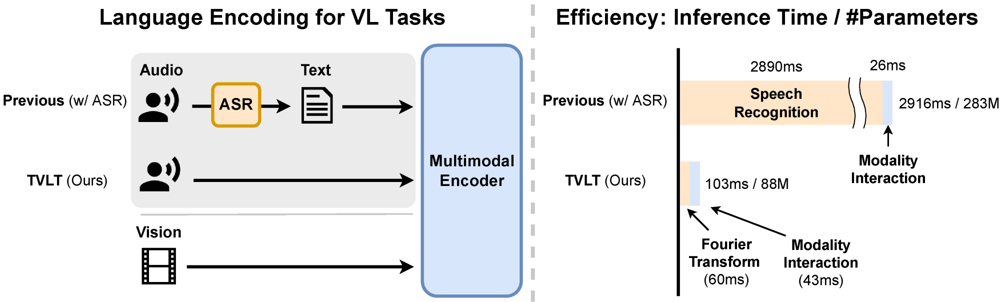


Figure 1: Comparison of previous VL architectures and our proposed textless framework TVLT. The removal of automatic speech recognition (ASR) from the VL pipeline brings efficiency improvement while maintains competitive performance. For inference time calculation, we use 8 video frames and 20s audio (see Sec. 6.2 for detail). As shown in Table 1, TVLT achieves competitive performance to text-counterpart on video retrieval and multimodal sentiment analysis tasks.

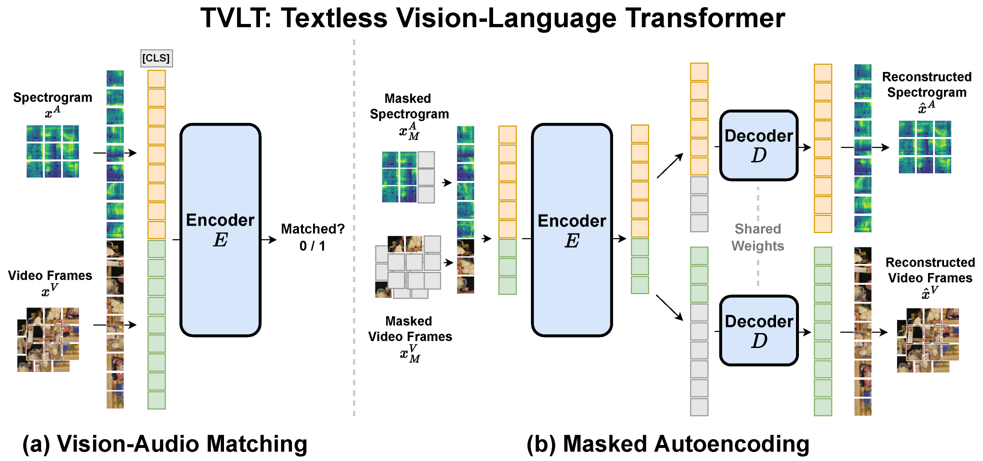
In this work, we propose Textless Vision-Language Transformer (TVLT) for vision-and-language representation learning based on video data as the natural source of raw visual and audio input. As depicted in Fig. 2, TVLT accepts low-level video frames and audio spectrograms as input. We employ a minimalist design for TVLT where homogeneous transformer blocks are used for both the encoder and decoder. TVLT is trained by reconstructing masked patches of continuous video frames and audio spectrograms (masked autoencoding) and contrastive modeling to align video and audio. More importantly, TVLT makes no assumptions about the existence of written language and does not involve explicit modeling of text input, such as automatic speech recognition (ASR) or tokenization, which are crucial submodules in the success of existing VL models in aligning written concepts with visual clues.

Despite the removal of text-based modules and modality-specific designs, TVLT achieves results comparable to its text-based counterparts in multimodal tasks (with either direct audio input, or text converted to audio input via TTS) such as visual question answering, image retrieval, video retrieval, and multimodal sentiment analysis, while being computationally efficient with 1/3 parameters and a 28x faster inference speed, as illustrated in Fig. 1. This indicates that the removal of text-specific modules such as ASR in vision-and-language modeling helps reduce computational redundancy in existing pipelined learning paradigms, where text is first extracted through ASR and then further processed by a text-based VL model. Furthermore, we also show that TVLT can capture acoustic information beyond speech and is more effective in multimodal emotion classification than its text-based counterpart. We hope that our findings spark further research in the realm of textless VL models that take raw signals as input and seek to learn a more compact and efficient vision-and-language representation.

2 Related Work

Text-based Representation Learning. Large-scale unsupervised pretraining of contextualized language models based on written texts has seen great success in recent years. ELMo [58] proposes to pretrain and finetune a large recurrent language model, which improves performance on a diverse set of downstream natural language processing tasks. BERT [15] improves the scalability of the pretrain-then-finetune paradigm by using a transformer [81] model with a masked language modeling objective. Since then, the pre-training of transformers has been extensively explored for transfer learning in language [46; 83; 38; 16; 73; 60; 13]. In these methods, learning is focused on eliciting high-level linguistic semantics and structures from unlabeled written texts or natural sequences of words.

Audio-based Representation Learning. Pretraining methods on audio input involve transferring the continuous 1D audio signal into dense vectors that can be input to a speech or acoustic model. Early work mainly uses recurrent neural networks [12; 11; 70] and convolution networks [66] for audio encoding. To take advantage of the proven expressiveness and genericity of transformers, more recent work proposed using audio spectrograms [19; 20; 7] as image input and then encoding the patches of such images with a transformer, following the same methodology in computer vision [17].


TVLT: Textless Vision-Language Transformer
Spectrogram x^A
[CLS]
Video Frames x^V
Encoder E
Matched? 0 / 1
Masked Spectrogram x^A_M
Masked Video Frames x^V_M
Encoder E
Decoder D
Shared Weights
Decoder D
Reconstructed Spectrogram $\hat{x}^A$
Reconstructed Video Frames $\hat{x}^V$

(a) Vision-Audio Matching

(b) Masked Autoencoding

Figure 2: TVLT is pretrained with two objectives: (a) vision-audio matching (Sec. 4.1) and (b) masked autoencoding (Sec. 4.2). The model takes video frames and audio spectrogram as inputs and does not use text input and completely removes text from the pipeline.

The pretraining objectives for transformers range from classification [19] to masked audio modeling [20; 7]. A line of work uses an audio transformer with discrete audio units for pretraining [27] and speech tasks such as generative spoken language modeling [37; 31] and speech emotion conversion [35]. These works focus on learning the acoustic and linguistic characteristics of a language from raw audio or spectrogram.

Vision-and-Language Representation Learning. Following the success of pretraining of transformer language models, pretraining of image+text [76; 48; 10; 43; 90; 41], video+text [74; 52; 92; 51; 42; 77; 87], and video+text+audio [79; 85; 61; 86; 2] multimodal transformers has recently achieved improvements in downstream VL tasks such as visual question answering [4; 28] and text-to-video retrieval [82; 91]. These methods use text, such as written captions or ASR transcripts, as input into the language channel. There is another line of work on models taking video+audio input, where they can utilize naturally synchronized vision+audio pairs from videos. Audio-visual synchronization is often used for self-supervised learning [56; 5; 55; 34; 6; 53; 49], or for downstream tasks such as automatic speech recognition [1; 72; 71] and video retrieval [75; 63; 64; 45]. Our work is different from these works, in that we focus on the design of a homogeneous and modality-agnostic transformer (Sec. 3) to achieve a novel, unified, and minimalist textless visual-linguistic representation learning method directly from visual and acoustic signals (without relying on text), via masked autoencoding and contrastive modeling objectives (Sec. 4), which also makes the textless VL model more compact and efficient than the existing text-based VL models.

3 TVLT: Textless Vision-Language Transformer

We introduce TVLT: **T**extless **V**ision-**L**anguage **T**ransformer, a minimal end-to-end vision-and-language transformer model that accepts a list of embeddings obtained directly from perception-level video and audio input *without text-specific modules*, as depicted in Fig. 1 and Fig. 2.

3.1 Input Embeddings

The input embeddings of TVLT are the sum of (1) modality embedding, (2) temporal/spatial embedding for video, (3) temporal/frequency embedding for audio, and (4) vision/audio patch embedding. As illustrated by the red and blue boxes in Fig. 2, the modality embeddings are two trainable vectors added to the input embeddings and used to indicate whether the input is from vision or audio input. In what follows, we explain the details of vision and audio embeddings.

Vision Embeddings. We adopt ViT [17]-style vision embedding, where each video frame of 224 × 224 pixels is divided into a list of 16 × 16-sized patches. Then, a liner projection layer is applied to the normalized pixel values of each patch, resulting in a 768-dimensional patch embedding. For a video clip with N frame samples, the input tensor with shape $N \times 224 \times 224 \times 3$ (time × height × width × channel) will result in $N \times 14 \times 14$ embeddings. The temporal and spatial embeddings are different trainable vectors added to the time, height, and width axis of the $N \times 14 \times 14$ embeddings to incorporate the temporal and spatial information for each input patch. We treat image input as a single frame video so that our model can handle both image and video tasks without modification of the architecture [9]. Temporal embedding is only added for video inputs; we do not use temporal embedding for images.

Audio Embeddings. To obtain audio embeddings, we first convert the 1D waveform of the raw audio signal to 128-dimensional log Mel-spectrogram having a dimension of $T \times 128$ (time axis × frequency axis).[2] Then, we treat the audio spectrogram as an image, divide the spectrogram images into patches, and apply a liner projection layer on each patch to obtain a 768-dimensional patch embedding. This follows the audio embedding methods in recent work [19; 20; 7], where a similar modality-agnostic transformer is used to model spectrogram patches. We experiment with two different patch sizes: 16×16 (square patches similar to the vision modality) and 2×128 (the same area as the first one but covers the entire frequency domain with a shorter time range) and use trainable temporal and frequency embeddings to indicate the temporal and frequency information of patches.[3]

3.2 Multimodal Encoder-Decoder

The main architecture of TVLT is a transformer [81] consisting of a 12-layer encoder (hidden size 768), E, and an 8-layer decoder (hidden size 512), D. We follow He et al. [26] and use a shallow decoder that only serves for masked autoencoding objective (Sec. 4.2) and has much fewer computations than the encoder. After pretraining, we only use the encoder representation for finetuning on downstream tasks.

4 Pretraining Objectives

By virtue of our minimal and modality-agnostic design, TVLT is pretrained with two objectives: (1) vision-audio matching (Sec. 4.1) and (2) masked autoencoding (Sec. 4.2). For each training batch, we compute each objective through a separate forward pass and use the weighted sum of them for the final loss, where $\lambda^{\text{VAM}} = 1.0$ and $\lambda^{\text{MAE}} = 0.3$.

$$loss = \lambda^{\text{VAM}} loss^{\text{VAM}} + \lambda^{\text{MAE}} loss^{\text{MAE}} \tag{1}$$

4.1 Vision-Audio Matching

We use the vision-audio matching (VAM) objective to learn the global cross-modal representation, as illustrated in Fig. 2 (a). For each video input, we create a (positive) vision-audio pair (x^{V+}, x^{A}). Then, we construct half of the vision-audio pairs inside a batch as mismatched (negative) pairs (x^{V-}, x^{A}), by replacing video frames x^{V+} with randomly sampled video frames x^{V-} from the training dataset.

Following previous vision-and-language transformers [76; 10; 48; 32], a linear layer with sigmoid activation is used as the classification head applied to the encoder output of the first `[CLS]` token to obtain the matching probability p. Then we compute the binary cross-entropy loss as:

$$loss^{\text{VAM}} = -y \log p \tag{2}$$

where y is 1 when the input vision-audio pair (x^{V}, x^{A}) is matched and 0 otherwise.

4.2 Masked Autoencoding

In addition to the VAM objective to learn cross-modal representation, we also use the masked autoencoding (MAE) objective to improve unimodal representations in the vision-and-language

[2] We use `melspectrogram` method of `librosa` [50] with arguments: `sampling rate=44100, n_fft=2048, hop length=512, window=‘hann’, pad_mode=‘constant’, n_mels=128`.

[3] With 16x16 patch, a 20-second audio will have a spectrogram with shape 640×128 (time axis × frequency axis), resulting in $40 \times 8 = 320$ patches.

settings, by masking random patches of visual frames and the audio spectrogram, and reconstruct missing inputs as shown in Fig. 2 (b). Concretely, we randomly drop a portion of visual x^V and audio embeddings x^A, then feed the remaining patch embeddings to the encoder E. We create inputs for the decoder D by adding the dropped embeddings as trainable vectors `[MASK]` to the same location as the original input (gray boxes in Fig. 2 (b)). We also add the corresponding temporal, positional, and frequency embeddings to the decoder input. Note that the temporal, positional, and frequency embeddings of the encoder and decoder are separately parameterized. We calculate the mean squared error between the reconstructed and original video frames and spectrograms:

$$loss^{\text{MAE}} = \frac{1}{N_M^V} \sum_{i \in masked} ||x_i^V - \hat{x}_i^V||_2^2 + \frac{1}{N_M^A} \sum_{j \in masked} ||x_j^A - \hat{x}_j^A||_2^2 \qquad (3)$$

where N_M^V and N_M^A are the number of masked patches for vision and audio, respectively. We compute the loss only on masked patches, similar to BERT [15].

To save computation, we slice the audio and video parts of the encoder output and feed them separately to the decoder, rather than decoding the video frames and the audio spectrogram jointly. In Sec. 6.6, we show that separate decoding achieves better finetuning performance, as well as better efficiency than joint decoding.

4.3 Masking Strategy

Vision Masking. Following MAE [26], we randomly mask 75% of the visual patches, and the masking is applied for each video frame independently.

Audio Masking. Following MAE-AST [7], we randomly mask 75% of the spectrogram patches. To better capture speech-related audio representation, we emphasize audio masking on speech audios. We use Audiotok [3], an audio activity detection tool, to determine speech spans based on the detection of events in the energy of the audio signal. Then, we apply the masking only on those audio spans. We use a probability of 15%. We include the details of speech span detection in appendix.

5 Experimental Setup

To compare the audio-based and text-based language representations for vision-and-language tasks, we pretrain our TVLT and its text-based counterpart on video datasets. Then, we finetune the models on a set of downstream vision-and-language datasets for evaluation.

5.1 Text-based TVLT Counterpart

Our text-based TVLT counterpart has the same architecture as the vanilla TVLT with minor changes to accommodate text-based inputs. Firstly, we use sentence-piece [36] tokenizer and then map each token to trainable vectors to encode the raw text into embeddings, instead of converting the continuous input of frames or spectrograms into patch embeddings as in vanilla TVLT. Secondly, we follow the norm in mask language modeling [15] to use an affine layer as the decoder to recover masked words and set the mask ratio on text to be 15%, instead of using a transformer decoder to reconstruct 75% of the masked video and audio embeddings in vanilla TVLT.

5.2 Pretraining Datasets

HowTo100M. We used HowTo100M [52], a dataset containing 136M video clips of a total of 134,472 hours from 1.22M YouTube videos to pretrain our model. Our vanilla TVLT is pretrained directly using the frame and audio stream of the video clips. Our text-based TVLT is trained using the frame and caption stream of the video. The captions are automatically generated ASR provided in the dataset. We used 0.92M videos for pretraining, as some links to the videos were invalid to download.

YTTemporal180M. YTTemporal180M [87] includes 180M video segments from 6M YouTube videos that spans multiple domains, and topics, including instructional videos from HowTo100M [52], lifestyle vlogs of everyday events from the VLOG dataset [29], and YouTube's auto-suggested videos for popular topics like 'science' or 'home improvement'. Each video segment consists of 1) an image frame extracted from the middle timestep of the segment, and 2) an ASR-based caption of L=32 BPE [18; 68] tokens. For each sample, we randomly sample a 15s video clip from the entire video to form a setting similar to HowTo100M dataset. Concretely, the original dataset provides 100 label files which are random split of the dataset. We sample 20% of YTTemporal180M (0.93M videos) so that the resulting subset consists of a similar number of videos to HowTo100M (0.92M videos), and call it YTT-S. In appendix, we show that pretraining TVLT on YTT-S can improve the downstream task performance of over pretraining on HowTo100M.

5.3 Downstream Tasks

We evaluate models on video-based and image-based vision-and-language tasks to compare the learned representation based on audio and text. For video-based tasks, we experiment with video retrieval [82; 91; 93] and multimodal sentiment analysis [85]. For image-based tasks, we experiment with image retrieval [84] and visual question answering [4; 21]. Although audio comes naturally with video, image-based tasks, such as visual question answering, do not include audio. Thus, we obtain audio queries for visual question answering via the text-to-speech (TTS) synthesis method (Sec. 5.4).

Audio-to-Video Retrieval. Following AVLnet [63], we use MSR-VTT [82], Youcook2 [91], and CrossTask [93] for audio-to-video retrieval. We also follow the same data split in AVLnet [63] to finetune our models on their respective training set.

MSR-VTT is an open domain video dataset, consisting of 10,000 video clips from 20 categories such as music, movies or food. We follow AVLnet for the standard split, i.e., 6,783 training clips and 1000 test clips (where 32 videos do not have sound). We report the test split results.

Youcook2 is a video dataset on cooking tutorials that contains 2,000 long videos of 89 cooking recipes. Each recipe has on average 22 videos. It has 9,586 training clips and 3,350 validation clips. We report the validation split results.

CrossTask dataset contains instructional videos for 83 different tasks, divided into 18 primary tasks and 65 related tasks. Primary tasks are manually collected with temporal step human annotations and are the main focus of tasks such as cooking or repairing. Related tasks are automatically collected without any annotations and are tasks related to the primary tasks, such as masking latte (primary) vs. making machiato (related). The goal of related tasks is to assess whether they can improve primary tasks. It has 17,840 training clips and 2,819 validation clips. We report the validation split results. For all three tasks, we extract `mp3` audio from videos with a sample rate of 44.1kHz. We also used the extracted audio or its corresponding ASR as retrieval queries for our experiment.

Multimodal Sentiment / Emotion Analysis. We use CMU-MOSEI [85] for multimodal sentiment analysis. The dataset is made up of 23,454 movie review clips with more than 65.9 hours of YouTube video by 1000 speakers that cover 250 distinct topics. Each video clip also comes with a ground-truth transcription written by the author of the video. Following previous studies, we use the 15,288/4,830 train-test split and report the binary accuracy (A2) for sentiment analysis and weighted accuracy (WA) and F1 score on emotion classification over 6 emotion categories.

Audio-to-Image Retrieval. We use Places-400k (The Places Audio Caption 400K Corpus) [25; 23; 24] for audio-to-image retrieval. The dataset contains approximately 1,000 hours of 400,000 spoken English captions for natural images drawn from the Places-205 [89] image dataset. The queries are conceptual descriptions of the image. The dataset also provides ASR of these audios. Places-205 is a large-scale scene dataset with 205 scene categories such as forest, bedroom, and coast, which contains 2,500,000 images in total.

Visual Question Answering. We use VQAv1 [4] and VQAv2 [21] for visual question answering. VQAv1 contains 204,721 images from COCO [44] and 430,725 questions. VQAv2 is a newer version of VQAv1, with 265,016 images from COCO and 1,105,904 questions. For experiments with audio questions, we generate speech audio from textual questions using TTS (Sec. 5.4) and report test-dev results for both tasks.

Finetuning on Downstream tasks. For each of the downstream tasks, we add a task-specific head (two-layer MLP) on top of the encoder representation. For retrieval tasks, we use an MLP to map

Table 1: Comparison of TVLT and its text-based counterpart on audio-to-video retrieval and video-based multimodal sentiment analysis benchmarks; *HT100M*=HowTo100M, *YTT-S*=YTTemporal180M subset.

Method	Input Mod.			Pretrain Datasets	Audio-to-Video Retrieval (R@1) ↑			Sentiment (A2) ↑	Latency ↓
	V	T	A		MSR-VTT	Youcook2	CrossTask	CMU-MOSEI	(ms)
TVLT	✓	✓		-	3.1	5.0	2.2	68.1	2916
TVLT	✓		✓	-	4.3	4.7	2.7	65.7	103
TVLT	✓	✓		HT100M	17.1	24.9	11.1	76.5	2916
TVLT	✓		✓	HT100M	22.6	31.8	14.9	75.3	103
TVLT	✓	✓		YTT-S	19.3	26.3	12.2	76.6	2916
TVLT	✓		✓	YTT-S	**23.8**	**32.8**	**15.3**	**76.8**	103

encoder representation of `[CLS]` to matching scores $\in [0, 1]$, which correspond to match vs. mismatch pairs, and train the model jointly with binary cross-entropy loss. For visual question answering tasks, we use an MLP to map the encoder representation of `[CLS]` to the answer probabilities with 3129 answer candidates, and train the model jointly with binary cross-entropy loss in a multi-label classification setup. For multimodal sentiment analysis tasks, we use an MLP to map the encoder representation of `[CLS]` token to the entiment scores, and train the model jointly with L2 regression loss.

5.4 Other Details

Automatic Speech Recognition (ASR). For the text-based model mentioned above, we obtain text from audio with different automatic speech recognition (ASR) models. We use the `asr-crdnn-rnnlm-librispeech` ASR model from the Speechbrain package [62]. The model is based on RNN language model and CRDNN encoder-CTC/Attention decoder architecture and is trained on LibriSpeech [57]. We also experiment with the Google Cloud Speech-to-Text API which uses Conformer [22] as the backend model.[4]

Text-to-Speech (TTS). We use `WaveNet` [80] Google Cloud Text-to-Speech API[5] to generate audio input for the questions in VQAv2. Since VQAv2 questions are written in English, we use a `en-US-neutral` speaker. We follow the default pitch and speech configuration. We use the `mp3` audio format with a sample rate of 44.1kHz to match the audio configuration used in the pretraining.

Pretraining. We train TVLT and the text-based TVLT counterpart for 200k steps using Adam optimizer [33] with a learning rate of 1e-5, batch size 4096, and a decay rate of 0.001 with a cosine schedule [47]. We initialize the weights of both models with the masked autoencoder transformer in He et al. [26] that is pretrained on ImageNet [14]. For the pretraining objectives in Eq. (1), we use $\lambda^{\text{VAM}} = 1.0$ and $\lambda^{\text{MAE}} = 0.3$. For each video clip, we uniformly sample 8 frames. Pretraining takes 2 weeks with 4 NVIDIA RTX A6000 GPUs (each 49GB memory).

Finetuning on Downstream Tasks. We use a learning rate of 1e-5, batch size 256, and a decay rate of 0.001 with a cosine schedule for all tasks. For each video clip, we uniformly sample 8 frames. We use 2 NVIDIA RTX A6000 GPUs.

6 Results and Analysis

6.1 Comparison to Text-based Counterpart

Table 1 shows that TVLT outperforms the text-based counterpart in audio-to-video retrieval tasks when pretrained on either HowTo100M or YTT-S. On CMU-MOSEI sentiment analysis, TVLT also outperforms its text variant when pretrained on YTT-S. In Table 2, although TVLT slightly underperforms the text-based counterpart on audio-to-image retrieval and visual question answering, TVLT can still achieve decently comparable results and remain competitive while being 27x faster during inference due to the removal of ASR from the processing pipeline. More details on

[4] `https://cloud.google.com/speech-to-text`

[5] `https://cloud.google.com/text-to-speech/docs/wavenet`

Table 2: Comparison of TVLT and its text-based counterpart on audio-to-image retrieval and visual question answering benchmarks.

Method	Input Mod.			Pretrain Datasets	Audio-to-Image Retrieval	Visual QA (Acc.) ↑	Latency ↓
	V	T	A		Places-400k (R@1 / R@5 / R@10) ↑	VQAv2	(ms)
TVLT	✓	✓		-	13.0 / 35.9 / 49.7	47.0	2010
TVLT	✓		✓	-	12.7 / 33.3 / 48.0	46.7	52
TVLT	✓	✓		HT100M	50.4 / 78.2 / 87.0	62.1	2010
TVLT	✓		✓	HT100M	48.7 / 77.9 / 86.0	60.8	52
TVLT	✓	✓		YTT-S	**54.3 / 78.9 / 88.8**	**63.2**	2010
TVLT	✓		✓	YTT-S	49.0 / 78.2 / 86.8	61.0	52

efficiency analysis are given in Sec. 6.2. The results provide evidence of the possibility of learning a more compact and efficient vision-and-language representation from raw visual and audio signals compared to the prevailing VL learning paradigms with explicit text-based modules in the pipeline.

6.2 Efficiency Comparison

To test inference latency, we sample 100 videos in CMU-MOSEI. As the average video length in the CMU-MOSEI dataset is 12 seconds, we measure the latency with two sets of input video lengths: 10 and 20 seconds. For 10s and 20s videos, we also use 4 and 8 video frames, respectively. Then we calculate the processing time of Fast Fourier Transform (FFT), SpeechBrain (ASR-SpBr) [62], TVLT, text-based TVLT, and AVLNet on the sampled inputs. SpeechBrain is the default ASR module that we used in our text-based counterpart pipeline (see Sec. 5.4).

Table 3: Latency of FFT, ASR and VL Models.

Model	# Param	Video Input	Latency (ms) ↓			
		Length / # Frames	FFT	ASR	VL	Total
ASR-SpBr	195M	10s / 4	-	2110	-	-
		20s / 8	-	2890	-	-
TVLT	88M	10s / 4	40	-	40	80
		20s / 8	60	-	43	103
TVLT + text	88M + 195M	10s / 4	-	2110	25	2135
	88M + 195M	20s / 8	-	2890	26	2916
AVLnet	158M	10s / 4	40	-	208	248
AVLnet + text	158M + 195M	10s / 4	-	2110	206	2316

As shown in Table 3, we find that ASR dominates the inference time for text-based models. Although ASR helps reduce the input length in transformers (as indicated by the VL module latency decrease), TVLT is more than 27x and 28x faster than text-based TVLT for inference with video input lengths of 10s and 20s, respectively, with only 1/3 of the parameters. The comparison is also shown in Fig. 1. In the bottom rows, we also show the inference latency of AVLnet and its text variant, where TVLT is 3x faster than AVLnet which contains audio-specific convolution modules.

6.3 Text Query vs. Speech Query for Language-based Video Retrieval

For text-to-video retrieval tasks, text captions are commonly used for queries [82]. In Sec. 6.1, we show the experiment of audio-to-video retrieval tasks following AVLnet [63], where the audio queries are the sounds of the original videos. Since video sounds and text captions have different information, the audio-to-video retrieval results are not directly comparable to the results in other text-to-video retrieval papers. For a better comparison, we experiment with video retrieval based on two language queries: 1) text captions and 2) speech audio obtained by TTS (see Sec. 5.4) from text captions. Table 4 shows MSR-VTT video retrieval results of TVLT with text/audio queries and recent text-to-video retrieval models pretrained with a similar scale of data.[6] Although TVLT with audio query slightly underperforms its text query counterpart due to TTS errors, it still outperforms other text-to-video retrieval models (HERO [42] and DeCEMBERT [77]), showing promising possibilities of speech-based video retrieval.

Table 4: Text vs. Speech Query for Video Retrieval.

Method	Pretrain Datasets	Query	Video Retrieval (R@1) ↑
			MSR-VTT
TVLT	HT100M	Caption	22.0
TVLT	HT100M	Speech Audio (TTS)	20.1
HERO [42]	HT100M	Caption	16.8
DeCEMBERT [77]	HT100M, TVQA	Caption	17.5
ClipBERT [39]	COCO, VG	Caption	22.0
AVLnet [63]	HT100M	Caption	22.5

[6] We exclude the models pretrained on large-scale image captions such as Conceptual Captions [69] that has written annotation, or visual encoder pretrained on a large-scale dataset beyond the scale of ImageNet [14], such as CLIP [59], as they are not directly comparable to our models.

Table 5: TVLT on CMU-MOSEI emotion analysis test set; *WA*=weighted accuracy, *F1*=weighted f1.

Method	Input Mod.			Happy		Sad		Angry		Fear		Disgust		Surprise	
	V	T	A	WA	F1	WA	F1	WA	F1	WA	F1	WA	F1	WA	F1
TVLT	✓	✓		64.7	63.9	70.2	66.0	68.9	71.8	66.2	84.4	**70.7**	**82.9**	58.4	86.2
TVLT	✓		✓	**65.1**	**64.1**	**72.2**	**70.0**	**69.9**	**72.1**	**68.1**	**88.0**	68.8	79.6	**62.1**	**87.4**

6.4 Emotion Analysis

Since TVLT takes raw visual and audio input instead of relying solely on text as in text-based TVLT, we further investigate what type of information TVLT can learn beyond speech on CMU-MOSEI emotion classification task. As shown in Table 5, TVLT outperforms the text-based counterpart in most emotion categories, except for 'Disgust'. We conjecture that TVLT is capable of capturing speech-related acoustic information, such as tone and loudness, which is helpful in recognizing these emotions, while this ability is absent from text-based ASR-dependent models.

Table 6: Finetuning performance on audio-to-video retrieval and multimodal sentiment analysis benchmarks. For a fair comparison, we gray out the models that use ground-truth text transcription as additional input for CMU-MOSEI.

Method	Input Mod.			Pretrain Datasets	Audio-to-Video Retrieval (R@1) ↑			Sentiment (A2) ↑
	V	T	A		MSR-VTT	Youcook2	CrossTask	CMU-MOSEI
Multilogue-Net [70]	✓		✓	-	-	-	-	75.2
AVLnet [63]	✓		✓	HT100M	20.1	30.7	13.8	-
TVLT (Ours)	✓		✓	HT100M	22.6	31.8	14.9	75.3
TVLT (Ours)	✓		✓	YTT-S	**23.8**	**32.8**	**15.3**	**76.8**

Table 7: Finetuning performance on audio-to-image retrieval and visual question answering (Visual QA). For Visual QA, we create spoken questions from text via TTS (Sec. 5.4). †CSC (Conceptual Spoken Caption) is 3.3M image-speech pairs, where speech is obtained via TTS API from Conceptual Captions. The CSC dataset is not publicly available.

Method	Input Mod.			Pretrain Datasets	Audio-to-Image Retrieval	Visual QA (Acc.) ↑
	V	T	A		Places-400k (R@1 / R@5 / R@10) ↑	VQAv1 / VQAv2
TextMod [88]	✓	✓		-	-	56.7 / -
SpeechMod [88]	✓		✓	-	-	47.0 / -
AVLnet [63]	✓		✓	HT100M	44.8 / 76.9 / 86.4	-
MILAN [64]	✓		✓	CSC†	**53.4** / **79.1** / 86.3	-
TVLT (Ours)	✓		✓	HT100M	48.7 / 77.9 / 86.0	58.6 / 60.8
TVLT (Ours)	✓		✓	YTT-S	49.0 / 78.2 / **86.8**	**58.9** / **61.0**

6.5 Comparison to State-of-the-art Textless Models

We compare our TVLT with recent models that also take raw visual and audio signals as input but involve audio-specific designs in their networks. As shown in Table 6, TVLT outperforms AVLnet [63] on three audio-to-video retrieval (MSR-VTT, Youcook2, CrossTask) tasks and outperform Multilogue-Net [70] on multimodal sentiment analysis (CMU-MOSEI) task with a simple modality-agnostic design. Similarly, Table 7 shows that TVLT achieves competitive results with AVLnet [63] and MILAN [64] on audio-to-image retrieval (Places-400k). Note that MILAN[7] is pretrained on Conceptual Spoken Caption [30] which contains 3.3M well-aligned image-speech pairs taken from Conceptual Captions [69] with TTS generated speech, whereas our TVLT is able to elicit effective representation from video inputs where vision-and-language clues are only weakly aligned. TVLT also outperforms both variants of the VQA models (TextMod, SpeechMod) in Zhang et al. [88] on VQAv1.

6.6 Ablation Studies

In the following, we show the results of the ablation study on TVLT training details: the audio masking strategy, the encoder/decoder architectures, and the pretraining objectives.

[7] The dataset is also not publicly available.

Audio Masking Strategy. In Table 8, we show the result of finetuning performance with different audio masking configurations, described in Sec. 4.3. For patch sizes, masking audio patches on detected speech spans improves performance across the board. However, we did not observe strict superiority between the two patch sizes; 2 × 128 achieves higher scores on MSR-VTT, while 16 × 16 achieves higher scores on VQAv2. For our default pretraining configuration, we use the 16 × 16 patch size and use speech span detection, since the 16 × 16 sized patch is also used in visual embedding (thus modality-agnostic) and speech span detection improves performance with minimal additional computation (see appendix).

Table 8: Audio masking configurations.

Patch Size	Masking on speech	MSR-VTT (R@1)	VQAv2 (Acc.)
16 × 16		21.7	57.8
16 × 16	✓	**22.3**	58.6
2 × 128		21.0	58.8
2 × 128	✓	21.2	**59.2**

Encoder Architecture. As described in Section 3.2, we use the joint encoder in TVLT. We compare this to modality-specific encoders for vision and audio. Table 9 below compares the separate encoders with the joint encoder for two tasks: VQAv2 and MSR-VTT. To tackle VQAv2 with separate encoders, we learned a two-layer self-attention fusion layer over the concatenation of hidden states of the vision and audio encoder. Our joint encoder architecture achieves better accuracy on both tasks than a separate encoder architecture. The results show that although vision and audio spectrogram are two different modalities, the single joint encoder could learn useful cross-modal representation for VL tasks without needing modality-specific encoders.

Table 9: Encoder variants.

Encoder	MSR-VTT (R@1)	VQAv2 (Acc.)
Separate	9.6	53.1
Joint	**10.2**	**54.6**

Decoder Architecture. As described in Sec. 4.2, we use separate decoders (with shared weights) for the vision and audio MAE pretraining objectives. We compare this separate decoding with joint decoder, where we feed the concatenated encoder outputs to the decoder and jointly reconstruct the video frames and spectrogram. Table 10 shows that pretraining with separate decoder outperforms joint decoder on finetuning performance, while being more efficient as well.

Table 10: Decoder variants.

Decoder	MSR-VTT (R@1)	VQAv2 (Acc.)
Separate	**22.3**	**58.6**
Joint	22.0	58.1

Pretraining Objectives. We measure the impact of each pretraining objective described in Sec. 4. Table 11 shows that each of the pretraining objectives (MAE and VAM) improves finetuning performance over random weight initialization. The combination of VAM and MAE further improves the finetuning performance, and we use this configuration as default for TVLT pretraining.

Table 11: Pretraining objectives.

Objectives	MSR-VTT (R@1)	VQAv2 (Acc.)
Random init	4.3	46.7
VAM	21.0	56.2
MAE	18.6	54.1
VAM + MAE	**22.3**	**58.6**

7 Conclusion

In this work, we present TVLT, a simple end-to-end vision-and-language transformer that can accept low-level visual and audio signals for vision-and-language representation learning. Our TVLT achieves competitive performance with other state-of-the-art audio-based vision-and-language models on visual question answering, image retrieval, video retrieval, and multimodal sentiment analysis. We also show that by eliminating the need for expensive ASR in the model pipeline, TVLT can be 28x faster than its text-based counterpart while achieving comparable performance. We comprehensively analyze the efficiency of our model and show ablation studies over different training variants. We hope that our research will inspire further exploration of simple and efficient vision-and-language frameworks with low-level signals.

Acknowledgments

We thank the reviewers for their helpful comments. This work was supported by ARO Award W911NF2110220, DARPA KAIROS Grant FA8750-19-2-1004, ONR Grant N000141812871, and NSF-AI Engage Institute DRL-211263. The views, opinions, and/or findings contained in this article are those of the authors and not of the funding agency.

References

[1] Triantafyllos Afouras, Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. 2018. Deep Audio-visual Speech Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–13.

[2] Hassan Akbari, Liangzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. 2021. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. *Advances in Neural Information Processing Systems*, 34.

[3] Sehili Amine. 2021. auditok: an audio/acoustic activity detection and audio segmentation tool.

[4] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *ICCV*.

[5] Relja Arandjelovic and Andrew Zisserman. 2017. Look, listen and learn. In *ICCV*.

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

[7] Alan Baade, Puyuan Peng, and David Harwath. 2022. Mae-ast: Masked autoencoding audio spectrogram transformer. *arXiv preprint arXiv:2203.16691*.

[8] Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. 2022. data2vec: A general framework for self-supervised learning in speech, vision and language. *CoRR*, abs/2202.03555.

[9] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. 2021. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, pages 1728–1738.

[10] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Uniter: Learning universal image-text representations. In *ECCV*.

[11] Yu-An Chung and James Glass. 2018. Speech2vec: A sequence-to-sequence framework for learning word embeddings from speech. *arXiv preprint arXiv:1803.08976*.

[12] Yu-An Chung, Chao-Chung Wu, Chia-Hao Shen, Hung-Yi Lee, and Lin-Shan Lee. 2016. Audio word2vec: Unsupervised learning of audio segment representations using sequence-to-sequence autoencoder. *arXiv preprint arXiv:1603.00982*.

[13] Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. In *ICLR*.

[14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *CVPR*.

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*.

[16] Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. In *NeurIPS*.

[17] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*.

[18] Philip Gage. 1994. A new algorithm for data compression. *C Users Journal*, 12(2):23–38.

[19] Yuan Gong, Yu-An Chung, and James Glass. 2021. Ast: Audio spectrogram transformer. *arXiv preprint arXiv:2104.01778*.

[20] Yuan Gong, Cheng-I Jeff Lai, Yu-An Chung, and James Glass. 2021. Ssast: Self-supervised audio spectrogram transformer. *arXiv preprint arXiv:2110.09784*.

[21] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *CVPR*, pages 6904–6913.

[22] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, et al. 2020. Conformer: Convolution-augmented transformer for speech recognition. In *Interspeech*, pages 5036–5040.

[23] David Harwath and James R Glass. 2017. Learning word-like units from joint audio-visual analysis. *arXiv preprint arXiv:1701.07481*.

[24] David Harwath, Adria Recasens, Dídac Surís, Galen Chuang, Antonio Torralba, and James Glass. 2018. Jointly discovering visual objects and spoken words from raw sensory input. In *Proceedings of the European conference on computer vision (ECCV)*, pages 649–665.

[25] David Harwath, Antonio Torralba, and James Glass. 2016. Unsupervised learning of spoken language with visual context. *Advances in Neural Information Processing Systems*, 29.

[26] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. 2021. Masked autoencoders are scalable vision learners. *arXiv preprint arXiv:2111.06377*.

[27] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460.

[28] Drew A. Hudson and Christopher D. Manning. 2019. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*.

[29] Oana Ignat, Laura Burdick, Jia Deng, and Rada Mihalcea. 2019. Identifying visible actions in lifestyle vlogs. *arXiv preprint arXiv:1906.04236*.

[30] Gabriel Ilharco, Yuan Zhang, and Jason Baldridge. 2019. Large-scale representation learning from visually grounded untranscribed speech. *CoRR*.

[31] Eugene Kharitonov, Ann Lee, Adam Polyak, Yossi Adi, Jade Copet, Kushal Lakhotia, Tu Anh Nguyen, Morgane Riviere, Abdelrahman Mohamed, Emmanuel Dupoux, and Wei-Ning Hsu. 2022. Text-free prosody-aware generative spoken language modeling. In *ACL*, pages 8666–8681, Dublin, Ireland. Association for Computational Linguistics.

[32] Wonjae Kim, Bokyung Son, and Ildoo Kim. 2021. ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision. In *ICML*.

[33] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. In *ICLR*.

[34] Bruno Korbar, Du Tran, and Lorenzo Torresani. 2018. Cooperative learning of audio and video models from self-supervised synchronization. In *NeurIPS*.

[35] Felix Kreuk, Adam Polyak, Jade Copet, Eugene Kharitonov, Tu Nguyen, Morgane Rivière, Wei-Ning Hsu, Abdel rahman Mohamed, Emmanuel Dupoux, and Yossi Adi. 2021. Textless speech emotion conversion using decomposed and discrete representations. *ArXiv*, abs/2111.07402.

[36] Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *EMNLP*, pages 66–71.

[37] Kushal Lakhotia, Eugene Kharitonov, Wei-Ning Hsu, Yossi Adi, Adam Polyak, Benjamin Bolte, Tu-Anh Nguyen, Jade Copet, Alexei Baevski, Abdelrahman Mohamed, and Emmanuel Dupoux. 2021. On generative spoken language modeling from raw audio. *Transactions of the Association for Computational Linguistics*, 9:1336–1354.

[38] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations. In *ICLR*.

[39] Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L. Berg, Mohit Bansal, and Jingjing Liu. 2021. Less is more: Clipbert for video-and-language learningvia sparse sampling. In *CVPR*.

[40] Gondy Leroy and David Kauchak. 2019. A comparison of text versus audio for information comprehension with future uses for smart speakers. *JAMIA Open*, 2(2):254–260.

[41] Gen Li, Nan Duan, Yuejian Fang, Ming Gong, Daxin Jiang, and Ming Zhou. 2020. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. In *AAAI*, pages 11336–11344.

[42] Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. 2020. Hero: Hierarchical encoder for video+ language omni-representation pre-training. In *EMNLP*.

[43] Wei Li, Can Gao, Guocheng Niu, Xinyan Xiao, Hao Liu, Jiachen Liu, Hua Wu, and Haifeng Wang. 2020. Unimo: Towards unified-modal understanding and generation via cross-modal contrastive learning. *arXiv preprint arXiv:2012.15409*.

[44] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755. Springer.

[45] Yan-Bo Lin, Jie Lei, Mohit Bansal, and Gedas Bertasius. 2022. Eclipse: Efficient long-range video retrieval using sight and sound. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

[46] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

[47] Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *ICLR*.

[48] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*.

[49] Shuang Ma, Zhaoyang Zeng, Daniel McDuff, and Yale Song. 2021. Active contrastive learning of audio-visual video representations. In *ICLR*.

[50] Brian McFee, Alexandros Metsai, Matt McVicar, Stefan Balke, Carl Thomé, Colin Raffel, Frank Zalkow, Ayoub Malek, Dana, Kyungyun Lee, Oriol Nieto, Dan Ellis, Jack Mason, Eric Battenberg, Scott Seyfarth, Ryuichi Yamamoto, viktorandreevichmorozov, Keunwoo Choi, Josh Moore, Rachel Bittner, Shunsuke Hidaka, Ziyao Wei, nullmightybofo, Adam Weiss, Darío Hereñú, Fabian-Robert Stöter, Pius Friesch, Matt Vollrath, Taewoon Kim, and Thassilo. 2022. librosa/librosa: 0.9.1.

[51] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. 2020. End-to-end learning of visual representations from uncurated instructional videos. In *CVPR*.

[52] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. 2019. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, pages 2630–2640.

[53] Pedro Morgado, Ishan Misra, and Nuno Vasconcelos. 2021. Robust audio-visual instance discrimination. In *CVPR*.

[54] Yixin Nie, Linjie Li, Zhe Gan, Shuohang Wang, Chenguang Zhu, Michael Zeng, Zicheng Liu, Mohit Bansal, and Lijuan Wang. 2021. MLP architectures for vision-and-language modeling: An empirical study. *CoRR*, abs/2112.04453.

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

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

[57] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: an asr corpus based on public domain audio books. In *ICASSP*, pages 5206–5210. IEEE.

[58] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL*.

[59] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*.

[60] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*.

[61] Wasifur Rahman, Md Kamrul Hasan, Amir Zadeh, Louis-Philippe Morency, Mohammed Ehsan Hoque, Sangwu Lee, AmirAli Bagher Zadeh, Chengfeng Mao, Louis-Philippe Morency, and Ehsan Hoque. 2020. Integrating Multimodal Information in Large Pretrained Transformers. In *ACL*, pages 2359–2369.

[62] Mirco Ravanelli, Titouan Parcollet, Peter Plantinga, Aku Rouhe, Samuele Cornell, Loren Lugosch, Cem Subakan, Nauman Dawalatabad, Abdelwahab Heba, Jianyuan Zhong, Ju-Chieh Chou, Sung-Lin Yeh, Szu-Wei Fu, Chien-Feng Liao, Elena Rastorgueva, François Grondin, William Aris, Hwidong Na, Yan Gao, Renato De Mori, and Yoshua Bengio. 2021. SpeechBrain: A general-purpose speech toolkit. ArXiv:2106.04624.

[63] Andrew Rouditchenko, Angie Boggust, David Harwath, Brian Chen, Dhiraj Joshi, Samuel Thomas, Kartik Audhkhasi, Hilde Kuehne, Rameswar Panda, Rogerio Feris, et al. 2020. Avlnet: Learning audio-visual language representations from instructional videos. *arXiv preprint arXiv:2006.09199*.

[64] Ramon Sanabria, Austin Waters, and Jason Baldridge. 2021. Talk, Don't write: A study of direct speech-based image retrieval. In *INTERSPEECH*.

[65] Denise Schmandt-Besserat. 2014. The evolution of writing. *International Encyclopedia of Social and Behavioral Sciences*, pages 1–15.

[66] Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. 2019. wav2vec: Unsupervised pre-training for speech recognition. *arXiv preprint arXiv:1904.05862*.

[67] Roy Schwartz, Jesse Dodge, Noah A Smith, and Oren Etzioni. 2020. Green ai. *Communications of the ACM*, 63(12):54–63.

[68] Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *ACL*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

[69] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*.

[70] Aman Shenoy and Ashish Sardana. 2020. Multilogue-net: A context-aware RNN for multi-modal emotion detection and sentiment analysis in conversation. In *ACL Workshop*, pages 19–28, Seattle, USA. Association for Computational Linguistics.

[71] Bowen Shi, Wei-Ning Hsu, Kushal Lakhotia, and Abdelrahman Mohamed. 2022. Learning audio-visual speech representation by masked multimodal cluster prediction. In *ICLR*.

[72] Brendan Shillingford, Yannis Assael, Matthew W. Hoffman, Thomas Paine, Cían Hughes, Utsav Prabhu, Hank Liao, Hasim Sak, Kanishka Rao, Lorrayne Bennett, Marie Mulville, Misha Denil, Ben Coppin, Ben Laurie, Andrew Senior, Nando De Freitas, and Nando De Freitas. 2019. Large-scale visual speech recognition. In *INTERSPEECH*.

[73] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. Mass: Masked sequence to sequence pre-training for language generation. In *ICML*.

[74] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. 2019. Videobert: A joint model for video and language representation learning. In *ICCV*.

[75] Dídac Surís, Amanda Duarte, Amaia Salvador, Jordi Torres, and Xavier Giró i Nieto. 2018. Cross-modal embeddings for video and audio retrieval. In *ECCV Workshop*.

[76] Hao Tan and Mohit Bansal. 2019. Lxmert: Learning cross-modality encoder representations from transformers. In *EMNLP*.

[77] Zineng Tang, Jie Lei, and Mohit Bansal. 2021. Decembert: Learning from noisy instructional videos via dense captions and entropy minimization. In *NAACL-HLT*, pages 2415–2426.

[78] Ian Tattersall, A Sophie, Frederick L Coolidge, and Thomas Wynn. 2009. *Cognitive Archaeology and Human Evolution*. Cambridge UP.

[79] Yao-Hung Hubert Tsai, Shaojie Bai, Paul Pu Liang, J. Zico Kolter, Louis-Philippe Morency, and Ruslan Salakhutdinov. 2019. Multimodal Transformer for Unaligned Multimodal Language Sequences. In *ACL*, pages 6558–6569.

[80] Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alexander Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. 2016. Wavenet: A generative model for raw audio. In *Arxiv*.

[81] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*.

[82] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*.

[83] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *NeurIPS*.

[84] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.

[85] Amir Zadeh and Paul Pu. 2018. Multimodal language analysis in the wild: Cmu-mosei dataset and interpretable dynamic fusion graph. In *ACL*.

[86] Rowan Zellers, Jiasen Lu, Ximing Lu, Youngjae Yu, Yanpeng Zhao, Mohammadreza Salehi, Aditya Kusupati, Jack Hessel, Ali Farhadi, and Yejin Choi. 2022. MERLOT Reserve: Neural Script Knowledge through Vision and Language and Sound. In *CVPR*.

[87] Rowan Zellers, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. 2021. Merlot: Multimodal neural script knowledge models. *Advances in Neural Information Processing Systems*, 34.

[88] Ted Zhang, Dengxin Dai, Tinne Tuytelaars, Marie-Francine Moens, and Luc Van Gool. 2017. Speech-based visual question answering. *arXiv preprint arXiv:1705.00464*.

[89] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. 2014. Learning deep features for scene recognition using places database. *Advances in neural information processing systems*, 27.

[90] Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason J Corso, and Jianfeng Gao. 2020. Unified vision-language pre-training for image captioning and vqa. In *AAAI*.

[91] Luowei Zhou, Chenliang Xu, and Jason J Corso. 2018. Towards automatic learning of procedures from web instructional videos. In *AAAI*.

[92] Linchao Zhu and Yi Yang. 2020. Actbert: Learning global-local video-text representations. In *CVPR*.

[93] Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. 2019. Cross-task weakly supervised learning from instructional videos. In *CVPR*, pages 3537–3545.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See supplementary material
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See supplementary material
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See supplemental material
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Sec. 5
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Sec. 5.4
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes] See supplementary material

(c) Did you include any new assets either in the supplemental material or as a URL? [Yes]

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

In this appendix, we include the pretraining dataset combination experiment (Appendix A), TTS-based text-to-video retrieval experiment (Sec. 6.3), ASR quality experiment (Appendix B), implementation details (Appendix C), finetuning on unimodal ASR task (Appendix D), visualization of MAE reconstruction (Appendix E), limitations and potential negative impacts (Appendix F), and licenses (Appendix G).

A Combination of Pretraining Datasets

Table 1 and Table 2 in the main paper show that TVLT either pretraining on HowTo100M [52] or YTT-S [87] can outperform random initialization across the board. Among the two pretraining datasets, models pretrained on YTT-S achieve higher performance than models pretrained on HowTo100M. The relative improvement is consistent with the findings of Zellers et al. [87], and we suspect that coverage of a wider range of video topics improves overall performance. We also experiment with pretraining TVLT with the combination of HowTo100M and YTT-S. The total size of the pretraining dataset size is 1.85M = (0.92M + 0.93M) videos, and we pretrain the model for 200k steps. As shown in Table 12, pretraining on the combination of both datasets achieves better finetuning performance than single-dataset pretraining on both MSR-VTT audio-to-video retrieval and VQAv2. The results indicate that TVLT can take advantage of the domain diversity of YTT-S and that pretraining with data from a diverse range of domains can result in a more adaptable representation.

Table 12: Finetuning performance of TVLT pretrained on different datasets.

Method	Input Mod.			Pretrain Datasets	Audio-to-Video Retrieval (R@1) ↑	Visual QA (Acc.) ↑
	V	T	A		MSR-VTT	VQAv2
TVLT	✓		✓	HowTo100M	22.6	60.8
TVLT	✓		✓	YTT-S	23.8	61.0
TVLT	✓		✓	HowTo100M+YTT-S	**25.0**	**61.4**

B Impact of ASR quality

Table 13 shows the results of TVLT on CMU-MOSEI sentiment analysis with the following different inputs: audio, ASR-based text, and ground-truth text transcriptions. ASR-Google and ASR-SpeechBrain refer to Google Cloud API and SpeechBrain, respectively (see main paper Sec. 5.4). Although TVLT pretrained on HowTo100M underperform the text variant with SpeechBrain ASR input, TVLT pretrained on YTT-S (76.8) achieves comparable results to those of the text variant with SpeechBrain ASR (76.6), which sheds light on the effectiveness of TVLT. Although there is still a gap between TVLT and text-based TVLT with higher quality ASR or ground truth transcript input, we expect that TVLT can be further improved with larger-scale pretraining (e.g., full YTTemporal180M dataset) on raw video signals.

Table 13: TVLT with audio/text on CMU-MOSEI.

Language Input	CMU-MOSEI (A2) ↑	
	HT100M	YTT-S
Audio	75.3	**76.8**
Text (ASR-SpeechBrain)	**76.5**	76.6
Text (ASR-Google)	77.1	77.8
Text (GT Transcripts)	78.9	79.1

To better understand the impact of ASR on downstream tasks, we show two examples of the CMU-MOSEI sentiment analysis task in Table 14. For example (a), ASR-Google Cloud provides more accurate transcription than ASR-SpeechBrain, resulting in more accurate sentiment estimation (ASR-SpeechBrain: -1.0 vs. ASR-Google: 0.0; label: 0.0). For example (b), ASR-Google Cloud and ASR-SpeechBrain provide similar transcription quality, resulting in the same sentiment estimation (ASR-SpeechBrain: 2.0 vs ASR-Google: 2.0; label: 1.0).

Table 14: Comparison of different ASR models on CMU-MOSEI sentiment analysis task. Sentiment label has range $[-3, 3]$, where -3 and 3 corresponds to negative and positive, respectively. We use TVLT pretrained on HowTo100M.

	GT Transcripts	ASR-SpeechBrain	ASR-Google Cloud	Pred (GT Transcripts)	Pred (ASR-SpBr)	Pred (ASR-GC)	Label
(a)	This is a new movie (uhh) in which a character is confined to his house, he is under house arrest, and his mother takes away his Xboxes and TV as sort of a little bit of additional punishment	communicate of additional punishment thoroughly	Serbia this is a new movie and which a character is confined to his house. He is under house arrest and his mother takes away his Xboxes and TVs is sort of a little bit of additional punishment.	0.0	-1.0	0.0	0.0
(b)	The club that I'm part of that organize that has about currently 40 some students and then last year we had 260-something come out to the dance.	well the club that i'm part of that organizes it has about currently forty some students and then flashed here we had two hundred and sixteen something come out to the dance	The club that I'm part of that organizes it has about currently 40 some students. And then last year we had 260 something come out to the dance	1.0	2.0	2.0	1.0

C Implementation Details

C.1 Speech Span Detection

For the speech span detection mentioned in the main paper Sec. 4.3, we use the Audiotok [3] word-level speech event detector. We use the configurations as follows: (1) We set a single speech event to have a duration within [0.3s, 1.2s], so that an event is likely to cover a single word. (2) We set `max_silence` $= 0.05s$. `max_silence` refers to the maximum silence gap between two speech spans. If the silence gap is too large, it is usually a stop between two words. Therefore, setting a low value ensures that we do not detect two words as a single word. (3) We use an energy threshold of 70, which is higher than the default value of 55, to avoid false positives of detecting noise. This is because real-world audio contains natural sounds and noises that usually come with a high level of audio signal energy. In the speech spans detected on HowTo100M, each word has an average length of 15 in our audio spectrogram (Sec. 3.1). As this is similar to the size of a single audio patch (16x16), masking an audio patch usually covers a word in speech.

Table 15: Audio Pipeline Latency.

Audio Length	CPU Latency (ms) ↓			GPU Latency (ms) ↓	
	Data Loading	Fast Fourier Transform	Speech Span Detection	ASR	VL Model
10s	60	40	130	2110	40
20s	110	60	170	2890	43

C.2 Audio Pipeline latency

In Table 15, we show the detailed latency for each audio processing pipeline for two different audio length settings: 10s and 20s. In both settings, ASR takes significantly longer processing time than all other modules and becomes the bottleneck of the entire vision-and-language pipeline.

D Finetuning on Unimodal ASR Task

Table 16: Finetuning on ASR.

Encoder PT	WER (%) ↓	
	dev-clean	dev-other
No-pretrain	3.1	6.0
V+A pretrain	**2.3**	**4.7**

To explore whether the cross-modal representation of TVLT is useful for unimodal tasks, we experiment using TVLT as an audio encoder for an ASR model. Specifically, we construct a 4-layer transformer language model that attends to TVLT encoder outputs via cross-attentions and jointly train the encoder and decoder. We experiment with two settings: where

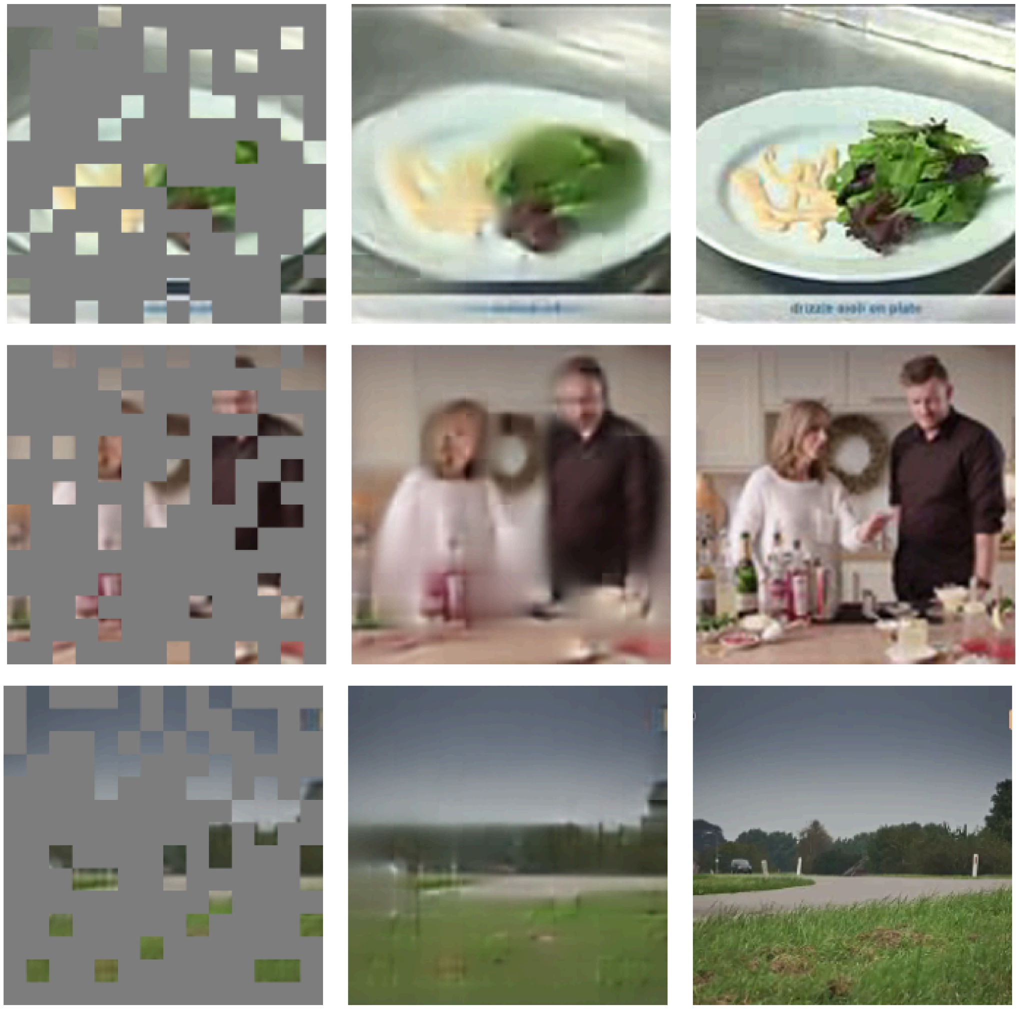


Figure 3: Visualization on video frames reconstruction (single frame): masked frames (left), reconstruction (middle), and original frames (right).

the TVLT encoder is randomly initialized or initialized with V+A pretraining. We train the models on LibriSpeech [57], a widely used ASR corpus with 960 hours of English audiobooks, and evaluate them on its two dev sets, dev-clean and dev-other. As shown in Table 16, our ASR model with V+A pretrained TVLT encoder outperforms the No-pretrain baseline by 0.8 (dev-clean) and 1.3 (dev-other) in Word Error Rate (WER), respectively. The results show that the cross-modal representation learned by TVLT could also be helpful for ASR, a unimodal task.

E MAE Reconstruction Visualization

In Figure 3 and Figure 4, we show the reconstruction results with the MAE head, described in the main paper Sec. 4.2. In each figure, the left column shows the masked input, the middle column shows the reconstruction, and the right column shows the target. We use masking ratio 0.75, image size 224 × 224, and audio spectrogram size 176 × 128 (time × frequency) for this visualization.

F Limitations

Green AI. A key barrier to the adoption of Green AI [67] has been the incentive to use massive computational power for pretraining. As shown in our main paper, TVLT is also subject to pretraining

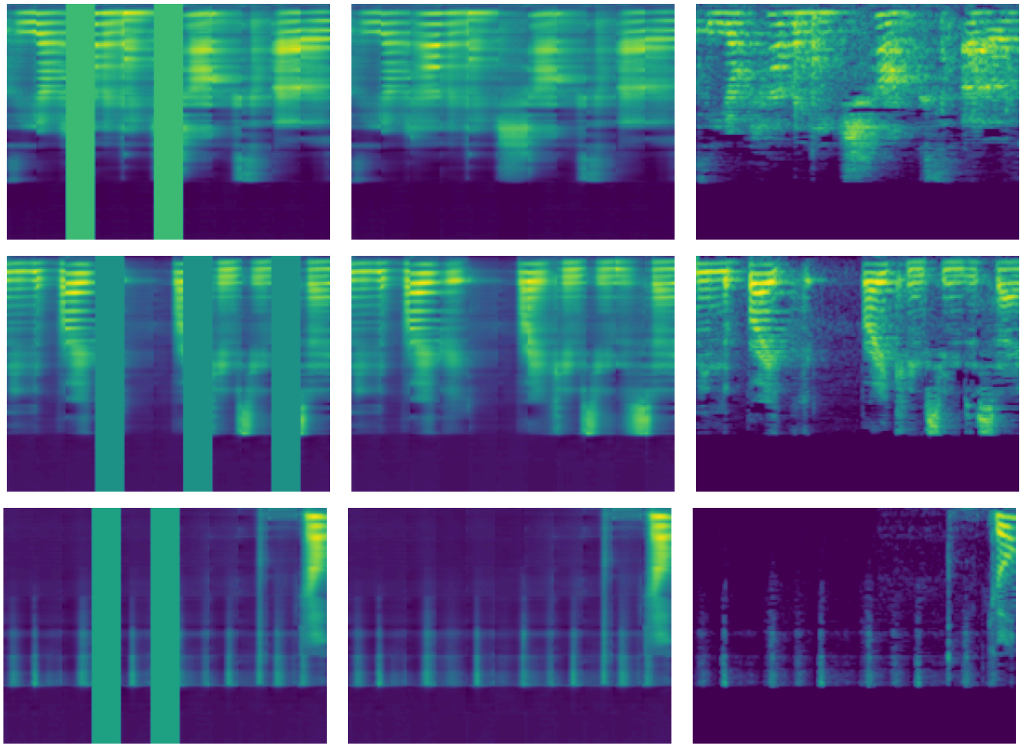


Figure 4: Visualization on video frames reconstruction: masked audio spectrogram (left), reconstruction (middle), and original audio spectrogram (right).

in order to achieve decent performance on visual linguistic tasks. While TVLT is substantially faster than vision-and-language models with explicit text-based modules that can help reduce pretraining computation, there is still scope for future work on energy-efficient training to alleviate the heavy reliance on large-scale pretraining.

English-only Datasets. We perform transfer learning with TVLT pretrained with HowTo100M and YTTemporal180M datasets. Both datasets mostly contain content in English, since HowTo100M [52] videos are obtained from English queries, and the authors of YTTemporal180M [87] filtered out videos with non-English ASR results. Therefore, our models pretrained with the two datasets might not have a good performance on non-English tasks without additional pretraining.

Note that the TVLT framework is a language-agnostic method, so one can adapt our model to a non-English dataset without any architectural change. Furthermore, our architecture eliminates the need for external ASR modules, which reduces the computation of the typical vision-and-language pipeline. To reduce environmental damage, we will publicly release our code and pretrained checkpoint.

G License

We will publicly release our code and models. We use standard licenses from the community and provide the following links to the licenses for the datasets, codes, and models that we used in the project. For more details, see the individual link.

HowTo100M: Apache

YTTemporal180M: MIT

PyTorch: BSD-style

Huggingface Transformers: Apache

Torchvision: BSD 3-Clause