TVLT: Textless Vision-Language Transformer

Anonymous Author(s)
Affiliation
Address
email

Abstract

Long before the emergence of written symbols, speech has been the main modality of verbal communication among humans. In this work, we present the Textless Vision-Language Transformer (TVLT), a transformer model that takes raw audio and visual inputs for vision-and-language representation learning with minimal modality-specific design, and does not use extra text-specific modules such as tokenization or automatic speech recognition (ASR). We show that TVLT attains results comparable to its text-based counterpart, on various multi-modal tasks such as video retrieval, image retrieval, visual question answering, and multimodal sentiment analysis, while being 50x faster during inference and with only one-third of the parameters. Our findings suggest the promising possibility of obtaining compact and efficient visual-linguistic representations by learning directly from low-level visual and audio perception signals.

1 Introduction

Humans perceive and learn the external world through signals from multiple modalities. In pursuit of embodying 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 [2]. Most of such VL models use written language rather than spoken language for the main verbal communication channel. However, the default communication modality among humans has been speech, since circa 100,000 BCE [52]. Written language is relatively recent; as cuneiform script, the earliest writing system was developed circa 3,200 BCE [44]. In addition, there is a substantial portion of the population who are illiterate or born with visual disabilities. They have limited understanding of written texts, but can easily communicate through spoken language. Furthermore, speech is increasingly used to access information on the Internet through virtual assistants and smart speakers [28].

Recent VL research has seen great success in visual-linguistic representation learning [50; 6; 34; 49] by adopting text-based transformers [54; 11] on text-annotated images or videos. However, only a few works [42; 43] have investigated the possibility of applying similar representation learning from low-level visual and acoustic inputs with no assumption on the prior existence of written language. The challenge is in the difference between text and perception-level visual/acoustic signals; text is discrete and dense in information, while visual/acoustic signals are continuous and sparse in information [21]. Therefore, modality-specific architectures have been used to model these different data; e.g., convolutional networks for visual/acoustic signal, recurrent networks for text. It is only recently that researchers started using modality-agnostic architectures to learn representations of different uni-modal [13; 14; 4] or vision-text data [24; 37]. However, to the best our knowledge, no previous work has explored a single architecture that learns visual-linguistic representations directly from perception-level visual and acoustic input.

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

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

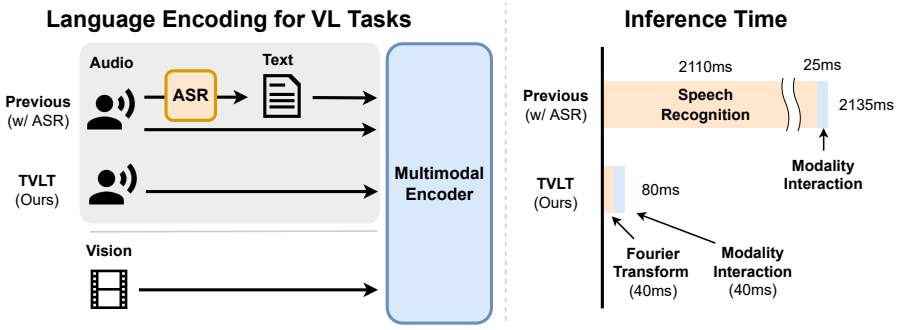

Figure 1: Comparison of previous VL architectures and our proposed textless framework TVLT. We have completely removed automatic speech recognition (ASR) from the VL pipeline while maintaining competitive performance on diverse image-language and video-language downstream tasks.

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. More importantly, TVLT makes no assumptions about the prior existence of written language and does not involve explicit modeling of text input, such as automatic speech recognition (ASR) or tokenization, which are otherwise crucial sub-modules in the success of existing VL models to align written concepts with visual clues.

Despite the removal of text-based modules and modality-specific designs, TVLT achieves comparable results with its text-based counterparts on multimodal tasks such as video retrieval, image retrieval, visual question answering, and multimodal sentiment analysis, while being computationally efficient with fewer model parameters and a magnitude shorter inference time, as illustrated in Fig. 1. This indicates that the removal of text-specific modules such as ASR in the vision-and-language modeling helps to reduce computational redundancy in existing pipelined learning paradigms, where text is first extracted via ASR and then additionally processed by a text-based VL model. Moreover, we also show that TVLT is able to capture acoustic information beyond speech and is more effective in multimodal emotion classification than the text-based TVLT 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 [39] proposes to pretrain and finetune a large recurrent language model, which improves performance on a diverse set of downstream natural language processing tasks. BERT [11] improves the scalability of the pretrain-then-finetune paradigm by using a transformer [54] model with a masked language modeling objective. Since then, the pre-training of transformers has been extensively explored for transfer learning in language [32; 56; 27; 12; 48; 40; 9]. In those methods, the learning is focused on eliciting high-level linguistic semantics and structures from unlabeled written texts or natural sequence 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 works mainly use audio-specific recurrent neural networks [8; 7; 47] and convolution networks [45] for audio encoding. To leverage the proven expressiveness and genericness of transformers, more recent works proposed to use audio spectrograms [14; 15; 3] as an image input, then encode the patches of such images with a transformer, following the same methodology in computer vision [13]. The pretraining objectives for transformers range from classification [14] to masked audio modeling [15; 3]. In these works, the learning is focused on transcription of texts or understanding acoustic semantics from uni-modal raw audio signals or low-level spectrogram inputs.

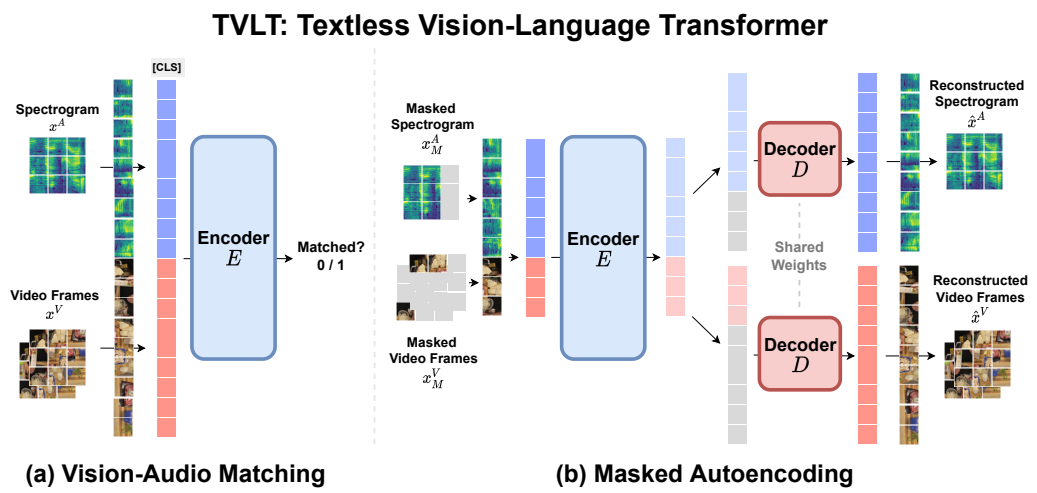

Figure 2: TVLT is pretrained with two objectives: (a) vision-audio matching (Sec. 4.1) and (b) masked autoencoding (Sec. 4.2). The model takes video frames and audio spectrogram as inputs and does not use text input and completely removes text from the pipeline.

Vision-and-Language Representation Learning. Following the success of uni-modal language pretraining with transformer models, pretraining of image-text [50; 34; 6; 31; 62; 29] and video-text [49; 36; 65; 35; 30; 51] multimodal transformers have recently achieved improvements in numerous multimodal downstream tasks [2; 55; 63]. All these existing methods use text, such as written captions and transcripts from ASR models, as the input of the language channel. The learning is focused on connecting visual semantics, including objects, attributes, parts, spatial relationships, actions, and intentions, with high-level written concepts where natural language words were taken for granted. Similarly to our work, AVLnet [42] encodes raw audio and video signals input using modality-specific models and optimizes the models only for audio-video matching. We simplify the modeling of audio and video with a homogeneous and modality-agnostic transformer and unify visual-linguistic representation learning via masked autoencoding and contrastive modeling objectives.

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 summation 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 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 [13]-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-dimension patch embedding. For a video clip with N frame samples, the input tensor with the shape $N \times 224 \times 224 \times 3$ (time axis $\times$ height $\times$ width $\times$ 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 patches. We treat image input as a single-frame video so that our model can handle both image and video tasks without

architecture modification [5]. 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 raw audio signal to 128-dimensional log Mel-spectrogram having a dimension of $T \times 128$ (time axis $\times$ frequency axis).[1] Then, we treat the audio spectrogram as an image, divide the spectrogram images into patches, and applied a liner projection layer on each patches to obtain a 768-dimension patch embedding. This is to follow audio embedding methods in recent work [14; 15; 3] where a similar modality-agnostic transformer is used for modeling 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.[2]

3.2 Multimodal Encoder-Decoder

The main architecture of TVLT is a transformer [54] consisting of a 12-layer encoder (hidden size 768), E, and an 8-layer decoder (hidden size 512), D. We follow He et al. [21] and use a shallow decoder that only serves for masked autoencoding objective (Sec. 4.2) and has much fewer computations than encoder. After pretraining, we only use the encoder representation for finetuning on downstream tasks.

4 Pretraining Objectives

By the 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 vision-audio matching (VAM) objective to learn 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 prior vision-and-language transformers [50; 6; 34; 24], 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 for learning cross-modal representation, we also use the masked autoencoding (MAE) objective to refine the uni-modal representations in the vision-and-language 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 back the dropped embeddings as a 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

[1] We include audio processing details in appendix.

[2] In our audio processing, a 20-second audio will have a spectrogram with shape 626×128 (time axis $\times$ frequency axis), resulting in $40 \times 8 = 320$ patches.

frequency embeddings for encoder and decoder are separately parameterized. We calculate the mean squared error between the reconstructed and original video frames and spectrograms:

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

where N_M^V and N_M^A are the number of the masked patches for vision and audio, respectively. We compute the loss only on masked patches, similar to BERT [11].

To save computation, we slice the audio and video parts of the encoder output and feed them separately to the decoder, instead of 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 [21], we randomly mask 75% of the visual patches and the masking is applied for each video frame independently.

Audio Masking. Following MAE-AST [3], we randomly mask 75% of the spectrogram patches. In order to better capture speech-related audio representation, we emphasize the audio masking on speech audios. We use Audiotok [1], an audio activity detection tool, to determine the 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 include the details of speech span detection in appendix.

5 Experimental Setup

To compare the audio-based and text-based visual-linguistic representations for vision-and-language learning, 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 [26] 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 langauge modeling [11] to use an affine layer as the decoder to recover the 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 [36], a dataset containing 136M video clips of total 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.

YTTemporal180M. YTTemporal180M [59] includes 180M video segments from 6M YouTube videos that spans multiple domains, and topics, including instructional videos from HowTo100M [36], lifestyle vlogs of everyday events from the VLOG dataset [22], and YouTube's auto-suggested videos for popular topics like 'science' or 'home improvement'. We sample 20% of YTTemporal180M such that the resultant subset will have roughly the same size as that of HowTo100M and pretrain our model on both the subset of YTTemporal180M and HowTo100M to test the domain adaptability of our model. In section Sec. 6.5, we show that TVLT is able to learn better representation when immersed with multi-domain YTTemporal180M during pretraining.

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 [55; 64; 66] and multimodal sentiment analysis [58]. For image-based tasks, we experiment with image retrieval [57] and visual question answering [2; 16]. While audio naturally comes with video, image-based VQA do not include audio. Thus, we obtain audio VQA via text-to-speech (TTS) synthesis method (Sec. 5.4).

Audio-to-Video Retrieval. Following AVLnet [42], we use MSR-VTT [55], Youcook2 [64], and CrossTask [66] for audio-to-video retrieval. We also follow the same data split in AVLnet [42] to finetune our models on their respective training set. We report test result for MSR-VTT and validate result for CrossTask and Youcook2. Note that CrossTask and Youcook2 are instructual videos while MSR-VTT contains general YouTube video with no specific domain constraints. The details of the three datasets can be found in Appendix.

Multimodal Sentiment / Emotion Analysis. We use CMU-MOSEI [58] 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 covers 250 distinct topics. Each video clip also comes with ground truth transcription written by the author of the video. Following prior study, 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) [20; 18; 19] 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 [61] 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.

Speech-based Visual Question Answering. We use VQAv1 and VQAv2 [2; 16] for speech-based visual question answering. VQAv1 contains 204,721 images from COCO and abstract scenes with 430,725 question samples. VQAv2 is a newer version of VQAv1. It contains 265,016 images from COCO and abstract scenes with 1,105,904 questions. We generate speech audio of textual questions using TTS (Sec. 5.4) and report test-dev results for both tasks.

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 `asr-crdnn-rnnlm-librispeech` ASR model from Speechbrain package [41]. The model is based on RNN language model and CRDNN encoder-CTC/Attention decoder architecture and is trained LibriSpeech [38]. We also experiment with the Google Cloud Speech-to-Text API which uses Conformer [17] as the backend model.[3]

Text-to-Speech (TTS). We use `WaveNet` [53] Google Cloud Text-to-Speech API[4] 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 `mp3` audio format with sample rate of 44.1kHz to match the audio configuration used in the pretraining.

Pretraining Details. We train TVLT and text-based TVLT counterpart for 200k steps using Adam optimizer [25] with a learning rate of 3e-5, batch size 4096, and a decay rate of 0.001 with a cosine schedule [33]. We initialize the weights of both models with the masked autoencoder transformer in He et al. [21] which is pretrained on ImageNet [10]. 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 sample 4 frames with a uniform interval between them. Pretraining takes 2 weeks with 4 NVIDIA RTX A6000 GPUs (each 49GB memory).

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

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

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	2135.0
TVLT	✓		✓	-	4.3	4.7	2.7	65.7	43.0
TVLT	✓	✓		HT100M	17.1	24.9	11.1	76.5	2135.0
TVLT	✓		✓	HT100M	22.6	31.8	14.9	75.3	43.0
TVLT	✓	✓		YTT-S	19.3	26.3	12.2	76.6	2135.0
TVLT	✓		✓	YTT-S	**23.8**	**32.8**	**15.3**	**76.8**	43.0

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	SpeechVQA (Acc.) ↑	Latency ↓
	V	T	A		Places-400k (R@1 / R@5 / R@10) ↑	VQAv2	(ms)
TVLT	✓	✓		-	13.0 / 35.9 / 49.7	47.0	2010.0
TVLT	✓		✓	-	12.7 / 33.3 / 48.0	46.7	12.0
TVLT	✓	✓		HT100M	50.4 / 78.2 / 87.0	62.1	2010.0
TVLT	✓		✓	HT100M	48.7 / 77.9 / 86.0	60.8	12.0
TVLT	✓	✓		YTT-S	**54.3 / 78.9 / 88.8**	**63.2**	2010.0
TVLT	✓		✓	YTT-S	49.0 / 78.2 / 86.8	61.0	12.0

Finetuning on Downstream Tasks. We use a learning rate of 3e-5, batch size 256, and a decay rate of 0.001 with a cosine schedule for all tasks. For each video clip, we sample 4 frames with uniform interval between them. 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 on 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 speech-based visual question answering, TVLT is still able to achieve decently comparable results and remain competitive while being 50x faster during inference because of the removal of ASR in the processing pipeline. More details regarding efficiency analysis are in Sec. 6.2. The results provide evidence on the possibility of learning a more compact and efficient vision-and-language representation from raw visual and audio signals comparing to the prevailing VL learning paradigms with explicit text-based modules in the pipeline.

6.2 Efficiency Comparison

To test the inference latency, we sample 100 videos in CMU-MOSEI[5] and crop them to 10 and 20 seconds, a representative range of raw input length. Then, we calculate the processing time of SpeechBrain, TVLT, text-based TVLT, and AVLNet on the sampled inputs. SpeechBrain [41] (ASR-SpBr) is the default ASR module we used in our text-based counterpart pipeline (see Sec. 5.4).

Table 3: Latency of ASR and VL Models.

Model	# Param	Video length	Latency (ms) ↓		
			ASR	VL	Total
ASR-SpBr	195M	10s	2110	-	-
		20s	2890	-	-
TVLT	88M	10s	-	40	40
		20s	-	43	43
TVLT + text	88M + 195M	10s	2110	25	2135
	88M + 195M	20s	2890	26	2916
AVLnet	158M	10s	-	208	208
AVLnet + text	158M + 195M	10s	2110	206	2316

As shown in Table 3, we find that ASR dominates the inference time for text-based models. Although ASR help reduce the input embedding

[5]The average video length in CMU-MOSEI dataset is 12 seconds.

Table 4: 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**

Table 5: 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 [47]	✓		✓	-	-	-	-	75.2
AVLnet [42]	✓		✓	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**

length in transformers (as indicated by the processing time of VL sub-module), the vanilla TVLT is more than 50x and 65x faster than the text-based TVLT for inference on 10 and 20 seconds of video inputs respectively while with only one-third 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 5x faster than audio-based AVLnet which contains audio-specific convolution sub-modules.

6.3 Emotion Analysis

Since TVLT takes raw visual and audio input instead of solely relying on text as in the text-based TVLT, we further investigate what type of information TVLT is able to learn beyond speech on CMU-MOSEI emotion classification task. As shown in Table 4, TVLT outperforms the text-based counterpart in most emotion categories, except for 'Disgust'. We suspect that TVLT is able to capture speech-related acoustic information, such as tone and loudness, which is helpful to recognize these emotions, while such ability is absent from the text-based ASR-dependent models.

6.4 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 involves audio-specific designs in their networks. As shown in Table 5, TVLT outperforms AVLnet [42] on three audio-to-video retrieval (MSR-VTT, Youcook2, CrossTask) tasks and outperform Multilogue-Net [47] on multimodal sentiment analysis (CMU-MOSEI) task with a simple modality-agnostic design. Similarly, Table 6 shows that TVLT achieves competitive results with AVLnet [42] and MILAN [43] on audio-to-image retrieval (Places-400k). Note that MILAN[6] is pretrained on Conceptual Spoken Caption [23] which contains 3.3M well-aligned image-speech pairs taken from Conceptual Captions [46] 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. [60] on speech-based visual question answering (SpeechVQA).

6.5 Impact of Pretraining

Table 1 and Table 2 show that TVLT either pretraining on HowTo100M or YTT-S 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. [59], and we suspect that coverage of a wider range of video topics improves overall performance.

[6]The dataset is also not publicly available.

Table 6: Finetuning performance on audio-to-image retrieval and visual question answering. For SpeechVQA, 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	SpeechVQA (Acc.) ↑
	V	T	A		Places-400k (R@1 / R@5 / R@10) ↑	VQAv1 / VQAv2
TextMod [60]	✓	✓		-	-	56.7 / -
SpeechMod [60]	✓		✓	-	-	47.0 / -
AVLnet [42]	✓		✓	HT100M	44.8 / 76.9 / 86.4	-
MILAN [43]	✓		✓	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**

6.6 Ablation Studies

In the following, we show the results of the ablation study on TVLT training details: the audio masking strategy, the modality-specific decoder, and pretraining objectives.

Audio Masking Strategy. In Table 7, 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. We use 16×16 with speech span detection for our default pretraining configuration, since it is a modality-agnostic patch size that is also used in visual embedding and the speech span detection improves performance with minimal additional computation (see appendix).

Table 7: 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**

Modality Specific Decoder. 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 to joint decoding, where we feed the concatenated encoder outputs to decoder and jointly reconstruct the video frames and spectrogram. Table 8 shows that pretraining with separate decoder outperforms joint decoder on finetuning performance, while being more efficient as well.

Table 8: 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 9 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 9: 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 audio-based video and image retrieval, multimodal sentiment analysis, and speech VQA. We also show that by eliminating the need of expensive ASR in the model pipeline, TVLT can be 50x 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 on simple and efficient vision-and-language frameworks with low-level perception signals.

References

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

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

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

[4] 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.

[5] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1728–1738.

[6] 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*.

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

[8] 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*.

[9] 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*.

[10] 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*.

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

[12] 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*.

[13] 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*.

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

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

[16] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913.

[17] 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. *Proc. Interspeech 2020*, pages 5036–5040.

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

[19] 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.

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

[21] 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*.

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

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

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

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

[26] Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71.

[27] 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*.

[28] 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.

[29] 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.

[30] 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*.

[31] 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*.

[32] 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*.

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

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

[35] 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*.

[36] 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.

[37] 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.

[38] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5206–5210. IEEE.

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

[40] 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*.

[41] 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.

[42] 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*.

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

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

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

[46] 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*.

[47] Aman Shenoy and Ashish Sardana. 2020. Multilogue-net: A context-aware RNN for multi-modal emotion detection and sentiment analysis in conversation. In *Second Grand-Challenge and Workshop on Multimodal Language (Challenge-HML)*, pages 19–28, Seattle, USA. Association for Computational Linguistics.

[48] 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*.

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

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

[51] 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.

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

[53] 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*.

[54] 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*.

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

[56] 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*.

[57] 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.

[58] Amir Zadeh and Paul Pu. 2018. Multimodal language analysis in the wild: Cmu-mosei dataset and interpretable dynamic fusion graph. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Long Papers)*.

[59] 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.

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

[61] 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.

[62] 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*.

[63] Luowei Zhou, Chenliang Xu, and Jason J Corso. 2017. Towards automatic learning of procedures from web instructional videos. In *AAAI*.

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

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

[66] 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]