Audiovisual Masked Autoencoders

Mariana-Iuliana Georgescu[1, 2*†]   Eduardo Fonseca[1]   Radu Tudor Ionescu[2]

Mario Lucic[1]   Cordelia Schmid[1]   Anurag Arnab[1*]

[1]Google Research   [2]University of Bucharest
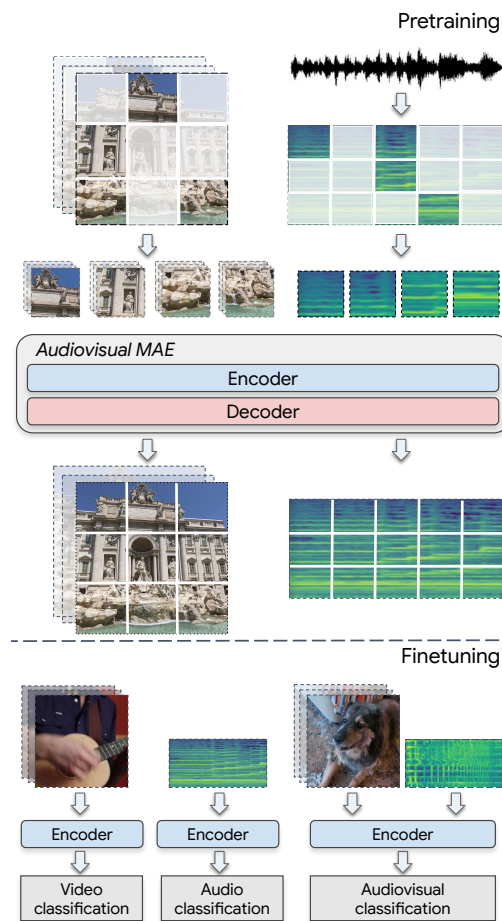
Pretraining / Finetuning diagram

Figure 1: Overview of our Audiovisual Masked Autoencoder. We jointly encode and reconstruct audiovisual inputs, to leverage the correlations between the two modalities to learn stronger representations of the data. Our pretrained encoder can then be used for audiovisual, audio-only and video-only downstream tasks.

Abstract

Can we leverage the audiovisual information already present in video to improve self-supervised representation learning? To answer this question, we study various pretraining architectures and objectives within the masked autoencoding framework, motivated by the success of similar methods in natural language and image understanding. We show that we can achieve significant improvements on audiovisual downstream classification tasks, surpassing the state-of-the-art on VGGSound and AudioSet. Furthermore, we can leverage our audiovisual pretraining scheme for multiple unimodal downstream tasks using a single audiovisual pretrained model. We additionally demonstrate the transferability of our representations, achieving state-of-the-art audiovisual results on Epic Kitchens without pretraining specifically for this dataset.

1. Introduction

The computer vision community has witnessed rapid progress across a wide range of tasks and domains, driven by progressively larger models and datasets [11, 24, 58, 63, 78]. However, pretraining models on large labelled datasets [21, 50, 67] and then finetuning on smaller target datasets is not scalable: Annotating large pretraining datasets is expensive and time-consuming, and larger, more performant models require more pretraining data [24]. This has led to growing research in self-supervised pretraining methods which learn feature representations from unlabelled data, and has been extremely successful in natural language processing (NLP) for developing large language models [13, 22, 65]. More recently, similar methods have been adopted in the vision community as well [10, 37, 76].

In this paper, we propose to leverage the audiovisual information present in video to improve self-supervised representation learning. Despite recent advances in self-supervised image- [10, 37] and video-representation learning [26, 70, 76], these works still ignore the additional auditory information that is already present in their pretraining

sets. Intuitively, we aim to exploit the correlations between the modalities already present in video to learn stronger representations of the data for unimodal and multimodal downstream tasks. Our approach is further motivated by the fact that the world, and human perception of it, is inherently multimodal [61, 64].

Our approach is inspired by the masked autoencoding framework [10, 37], which itself is based on similar masked data modelling approaches in NLP [22] and earlier works

---

*Equal contribution. Correspondence to aarnab@google.com.
†Work done during an internship at Google.

on denoising autoencoders [55, 72]. We develop multiple pretraining architectures to jointly encode and reconstruct audiovisual inputs, and conduct thorough ablation studies to verify our design choices. To encourage further cross-modal information modelling, we also propose a novel "in-painting" objective which tasks our transformer model with predicting audio from video tokens and vice versa.

Our audiovisual pretraining enables us to achieve state-of-the-art results in downstream, audiovisual datasets such as VGGSound and AudioSet. Moreover, we show how we can reuse our audiovisual pretraining for unimodal, *i.e.* audio-only or video-only downstream classification tasks. Furthermore, we show how the representations learned by our model transfer between different pretraining and downstream finetuning datasets, enabling us to achieve state-of-the-art art results on the Epic Kitchens dataset.

2. Related Work

Early self-supervised learning works in vision were based on solving hand-designed pretext tasks such as relative patch prediction [23] and colourisation [81]. Subsequent works converged on contrastive [17, 36, 38, 53], self-distillation [15, 33, 47] or clustering-based [8, 14] objectives that encouraged a neural network to learn feature representations that were invariant to a predefined set of data transformations. These ideas were extended to multimodal scenarios as well, by predicting whether visual and audio signals come from the same video [4, 5, 46, 54], by audiovisual clustering [3, 7] or by using contrastive losses to encourage different modalities from the same input to have similar learned embeddings [1, 2, 51, 56, 73, 79].

Our approach, in contrast, is based on masked data modelling – a paradigm which removes part of the input data and then learns to predict this removed content – and has gained traction due to the success of BERT [22] in NLP. BEIT [10] notably adopted BERT-style pretraining for images, using a discrete variational autoencoder [60] to produce a vocabulary of image tokens, and then predicting these tokens for masked image patches using a cross-entropy loss. Masked Autoencoders (MAE) [37] further showed that simply regressing to the original inputs in pixel-space was just as effective, and by only processing unmasked tokens in the encoder, training could be significantly accelerated. MAE has recently been extended to video [26, 70, 74] and audio [18, 41]. Our work also uses the masked autoencoding framework, but jointly models both audio and video, and is demonstrated on both unimodal (*i.e.* video-only and audio-only) and audiovisual downstream tasks where it outperforms supervised pretraining.

A few recent works have also addressed multimodal pre-training with MAE: OmniMAE [28] trains a single MAE model to reconstruct both images and video with shared weights among the modalities. The model is trained in a multi-task setting, where it can process either images or videos, but only one modality at a time. It is thus a self-supervised equivalent of multi-task models like [29, 48] which can perform multiple classification tasks, but only whilst processing a single task from a single modality at a time. Our model, in contrast, is developed to fuse information from different modalities, for both multimodal and unimodal downstream tasks. Bachmann *et al.* [9] develop an MAE model for dense prediction tasks, where the model reconstructs images, depth maps and segmentation maps of the image. Training this model, however, requires real or pseudo-labels for segmentation and depth, and hence, the model is not purely self-supervised like our approach.

We also note that concurrent works, CAV-MAE [32] and MAViL, [40], have also recently explored joint masked autoencoding of audio and video. We observe two main differences with respect to our current study. Here, we focus on a thorough comparison of multiple architectures and objectives for audiovisual masked autoencoders. In contrast, those works explore more complex systems featuring additional objectives and training methodologies such as contrastive learning [32, 40] and/or additional distillation [40] training stages, whilst we perform a single-stage of self-supervised pretraining. Second, in contrast to [32, 40], our representations are learned from random initialisation and not from ImageNet-pretrained checkpoints.

Finally, we note that vision transformer architectures [24, 71] have also been extended to fusing multiple modalities [43, 52, 59]. Our work focuses on pretraining audiovisual models, and can in fact be used to initialise MBT [52] to improve the results that the original authors achieved with supervised pretraining.

3. Audiovisual Masked Autoencoders

We extend the masked-autoencoding framework [37] to learn audiovisual feature representations that can be leveraged for both multimodal and unimodal downstream tasks. This is done by jointly modelling both modalities to learn synergies between them. We begin with an overview of masked autoencoders [37] and transformers in vision in Sec. 3.1. Thereafter, we detail our different encoders (Sec. 3.2), decoders (Sec. 3.3) and objective functions (Sec. 3.4) as summarised in Fig. 2 and 3.

3.1. Background

Transformers are a generic architecture that operate on any input that can be converted into a sequence of tokens. Images are typically tokenised by performing a linear projection of non-overlapping "patches" of the input, which corresponds to a 2D convolution [24]. For videos, a common method is to linearly project spatio-temporal "tubelets" which is equivalent to a 3D convolution [6]. Audio inputs are commonly represented as spectrograms, which are 2-

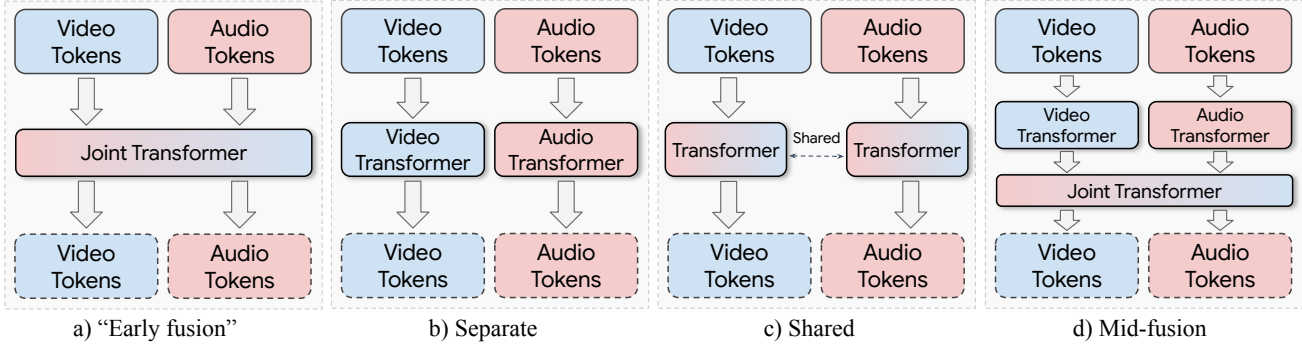| a) "Early fusion" | b) Separate | c) Shared | d) Mid-fusion |

Figure 2: Transformer architectures for performing audiovisual fusion. Concatenating the tokens before passing them through the transformer corresponds to "early fusion" (a), whilst using two separate encoders (b) can be used to perform "late fusion" in the subsequent decoder. An alternate method of coupling modalities together is by sharing weights between the two encoders (c). Finally, mid-fusion (d) represents a balance between "early" and "late" fusion.

dimensional representations (along the time and frequency axes) in the Fourier domain of the input waveform, and can thus be treated as images with a single channel [30] (in fact, ImageNet pretrained weights have also been shown to be effective for initialising audio models [30, 34, 35]).

Dosovitskiy *et al.* [24] showed that transformers, using the same original architecture as [71], excelled at image recognition tasks when pretrained on large, supervised datasets such as ImageNet-21K [21] and JFT [67]. More recently, approaches such as masked autoencoders [37] have demonstrated how vision transformers can be pretrained with only self-supervision on smaller datasets.

In the masked autoencoding framework [37], the input, $\mathbf{x}$, is tokenised following previous supervised learning setups [6, 24, 30]. We denote the resulting tokens as $\mathbf{v} = \text{Tokenise}(\mathbf{x}) + \mathbf{p}$, where $\mathbf{p}$ denotes the positional embeddings, and $\mathbf{v} \in \mathbb{R}^{n \times d}$ where n is the total number of tokens, and d is their hidden dimensionality. A random subset of these tokens are then masked, and only the unmasked tokens are processed by the transformer encoder. We denote these steps as $\mathbf{u} = \text{Mask}(\mathbf{v}; \alpha)$ and $\mathbf{e} = \text{Encode}(\mathbf{u})$, where $\mathbf{u}$ and $\mathbf{e} \in \mathbb{R}^{u \times d}$. Here α is the masking ratio, and $u = (1 - \alpha) \cdot n$ is the number of unmasked tokens.

Learned mask tokens, $m \in \mathbb{R}^d$ are then inserted back into the input token sequence whilst also adding new positional embeddings, which we denote as $\mathbf{z} = \text{Unshuffle}(\mathbf{e}, m) \in \mathbb{R}^{n \times d}$. Finally, a transformer decoder (which has the same structure as the encoder) processes these tokens, and the entire network is trained to reconstruct the original inputs corresponding to the tokens in pixel space, $\tilde{\mathbf{x}}$, with a mean-squared error objective. Note that $\tilde{\mathbf{x}}$ denotes the "patchified" version of the input $\mathbf{x}$ used as the reconstruction target. For example, for an image $\mathbf{x} \in \mathbb{R}^{H \times W \times 3}$, $\tilde{\mathbf{x}} \in \mathbb{R}^{H/p_h \cdot W/p_w \times p_h \cdot p_w \cdot 3}$, where H and W are the height and width of the image, and p_h and p_w are the patch sizes used for tokenisation. Additional standardisation may also be applied to $\tilde{\mathbf{x}}$ [37, 70].

3.2. Audiovisual encoders

As shown in Fig. 2, we consider different encoders which fuse audio and visual information at different stages. In all cases, we use the standard transformer architecture [24, 71].

Early fusion We first concatenate the unmasked tokens from the respective modalities, before passing them to a single transformer (Fig. 2a). This method is thus an "early" fusion [44] of audio and video. Due to the high masking ratios, α_a and α_v used for audio and video respectively, this is still computationally efficient and allows the encoder to model joint interactions between both modalities.

Separate This variant, in contrast, encodes audio and video tokens with two separate encoders each with different parameters. When using such an encoder, a "late fusion" [44, 62] of audio and video is performed in the decoder when reconstructing the tokens. This strategy, however, allows us to obtain separate encoders for each modality after pretraining, which may be advantageous for finetuning on a unimodal (*i.e.* audio-only or video-only) downstream task.

Mid-fusion Here, we perform a middle-ground between the previous two approaches (Fig. 2d). Denoting the total number of transformer layers as L, the first $L - S$ layers are separate for each modality as in the "Separate" encoding approach. The tokens are then concatenated into a single sequence, and then forwarded through a further $S \geq 1$ layers which jointly process both modalities.

Shared Finally, we explore coupling the two modalities together via parameter-sharing. As shown in Fig. 2c, the unmasked tokens for audio and video are encoded separately, but using the same transformer parameters in both cases. This is therefore equivalent to the "Separate" strategy, where the weights are tied between the encoders.

3.3. Decoders

In the masked autoencoding framework [10, 26, 37], the decoder is another transformer that reconstructs the masked tokens given the encoded tokens as context. The decoder has less capacity than the encoder, to force the encoder to
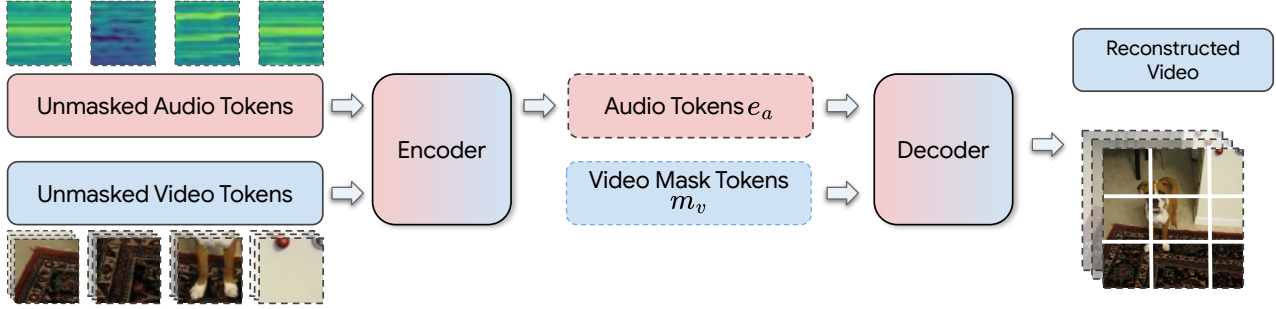
Figure 3: Overview of modality inpainting for reconstructing video from audio. We initially jointly encode unmasked tokens from both audio and video. Then, we use all the encoded tokens of one modality (*i.e.* audio), and mask tokens from the other (*i.e.* video), to reconstruct the masked modality (*i.e.* video). Note that we can reconstruct all combinations of modalities, and show one for clarity.

learn discriminative features which can be used for reconstruction. Moreover, this also improves training efficiency, as mask tokens are also processed by the decoder.

Our decoder can follow any of the encoder structures described previously, whilst also being shallower. Note that when the "separate" encoding strategy is used, the same decoding strategy should not be used as this would amount to two separate, uncoupled masked autoencoder models.

3.4. Objective

After encoding the unmasked tokens, and inserting the mask tokens back into the sequence, we obtain the inputs to our decoder, namely $\mathbf{z}_a \in \mathbb{R}^{n_a \times d}$ and $\mathbf{z}_v \in \mathbb{R}^{n_v \times d}$. Here, $\mathbf{z}_a$ and $\mathbf{z}_v$ correspond to the audio and video modalities, respectively. Similarly, n_a and n_v are the total number of audio and video tokens respectively.

Joint Reconstruction This is a straightforward extension of the masked autoencoding objective [37]. We reconstruct the original audio and video inputs that correspond to the mask tokens in $\mathbf{z}_a$ and $\mathbf{z}_v$ respectively, with the mean square error. This is denoted for audio as

$$\mathcal{L}_a(\mathbf{z_a}, \tilde{\mathbf{x}}_a) = \frac{1}{\alpha_a n_a} \sum_{i \in \mathcal{M}_a} ||\text{Decode}(\mathbf{z}_{a,i}) - \tilde{\mathbf{x}}_{a,i}||^2, \quad (1)$$

where $\mathcal{M}_a$ denotes the set of mask token indices, as we only compute the reconstruction loss on these tokens like [26, 37, 76]. The reconstruction loss for the video modality is defined similarly, and to avoid additional hyperparameters, we use equal loss weights for both audio and video.

Modality inpainting This objective aims to encourage stronger coupling between the two modalities by reconstructing one modality from the other. As shown in Fig. 3, we reconstruct the input video tubelets, $\tilde{\mathbf{x}}_v$ using the encoded audio, $\mathbf{e}_a$ and video mask tokens, m_v. Similarly, we reconstruct the input audio tokens, $\tilde{\mathbf{x}}_a$, using the encoded video $\mathbf{e}_v$ and audio mask tokens, m_a. Note that in order to use this objective, we require cross-modal encoding of the audio and video tokens, for example by using the "Early" or "Mid-fusion" encoding methods. Otherwise, the video tokens will not have the information necessary to reconstruct the audio tokens, and vice versa.

Another consideration is that there is no correspondence between the two sets of tokens (*i.e.* the ith audio token does not correspond to the ith video token, and the respective sequence lengths of the modalities, n_a and n_v, are also typically different). This poses a challenge for reconstruction, as for example, we do not know which video tokens need to be reconstructed given the encoded audio- and mask video-tokens. To resolve this, we create an ordering in the sequence fed to our decoder, $\mathbf{z}$, by placing all the encoded tokens at the beginning of the sequence, and all of the mask tokens at the end (Fig. 3), and then only computing the reconstruction error (Eq. 1) on the mask tokens.

Concretely, this means that for reconstructing video, the token sequence is $\mathbf{z}_v = [\mathbf{e}_a || \hat{\mathbf{m}}_v]$, where $[\cdot || \cdot]$ denotes concatenation. Note that the mask video token, m_v, is repeated $n_v - (1 - \alpha_a) \cdot n_a$ times to form, $\hat{\mathbf{m}}_\mathbf{v}$ and the reconstruction error is only computed on these tokens once they are decoded. Similarly, we reconstruct audio tokens by decoding the sequence $\mathbf{z}_a = [\mathbf{e}_v || \hat{\mathbf{m}}_a]$ where there are $n_a - (1 - \alpha_v) \cdot n_v$ mask tokens.

We experimentally evaluate our proposed pretraining architectures and objectives next.

4. Experimental Evaluation

We first describe our experimental setup in Sec. 4.1, before presenting ablation studies (Sec. 4.2) and comparisons to the state-of-the-art in Sec. 4.3. For reproducibility, we will release code and trained models on acceptance.

4.1. Experimental setup

Datasets We select 3 audiovisual datasets, where the audio and video signals are correlated and have been shown to be complementary [43, 52, 75]: VGGSound [16], AudioSet [27] and Epic Kitchens [19]. We follow standard protocols for these datasets, which we describe below.

VGGSound [16] is an audiovisual benchmark of almost 200 000 video clips, that are 10 seconds long and collected from YouTube. The videos were selected such that the object that is generating the sound is also visible in the video. Each video is annotated with one of 309 classes. After removing the unavailable URLs from the data set, we end up

with $170\,000$ samples for training and $14\,448$ for testing.

AudioSet [27] is the largest audiovisual dataset, consisting of almost 2 million videos from YouTube which are 10 seconds long. There are 527 classes, and the mean Average Precision (mAP) is the standard metric as the dataset is multilabel. The dataset is released as a set of URLs, and after accounting for videos that have been removed, we obtain 1.8 million training videos and $18\,589$ validation examples. AudioSet is an extremely imbalanced dataset, and numerous works reporting on it have used complex, batch sampling strategies [30, 31, 41] or trained on a smaller, but more balanced subset [52] of $500\,000$ training videos which we denote as AS500K. For self-supervised pretraining, we use the full, imbalanced AudioSet dataset, which we denote as AS2M, whereas for finetuning we use AS500K.

Epic Kitchens-100 [19] consists of $67\,000$ egocentric videos spanning 100 hours. We report results following the standard "action recognition" protocol. Here, each video is labelled with a "verb" and a "noun", and the top-scoring verb and noun pair predicted by the network form an "action", and action accuracy is the primary metric. Following [6, 52, 77], we predict both verb and noun classes with a single network with two "heads", and equal loss weights.

In contrast to the other datasets, Epic Kitchens does not use YouTube videos, and as the videos are egocentric, there is a substantial domain shift between them. Therefore, Epic Kitchens is well-suited for evaluating the transferability of the feature representations learned on the other datasets.

Implementation details Our training hyperparameters are based on the public implementations of unimodal MAE models [37, 70]. We use a masking ratio of $\alpha_v = 0.9$ for video and $\alpha_a = 0.5$ for audio. For both modalities, we use random token masking as prior works found this to outperform other masking patterns [26, 37, 41].

Following [30, 52], audio for all datasets is sampled at 16 kHz and converted to a single channel. We extract log mel-spectrograms, using 128 frequency bins computed using a 25ms Hamming window with a hop-length of 10ms. Therefore the input dimension is $128 \times 100t$ for t seconds of audio. Since we use 8 seconds of audio as [52], and a spectrogram patch size of 16×16, we have $n_a = 400$ audio tokens. For video, following [26, 70], we randomly sample 16 frames, which we tokenise with tubelets of size $16 \times 16 \times 2$. We implemented our model using the Scenic library [20] and JAX [12], and include exhaustive details in the supplementary.

Network architecture We use a standard vision transformer [24], namely the ViT-Base and Large models, following the same configurations as [22, 24]. When using the "mid-fusion" encoder, we use $S = 2$ shared layers between the modalities, and ablate this choice in the supplementary.

Finetuning We first pretrain our network architectures from Sec. 3, and then evaluate the learned representations

Table 1: Ablation study of different pretraining architectures. Models are pretrained from random initialisation for 400 epochs on VGGSound, and then finetuned on the same dataset.

Encoder	Decoder	Audio-only	Video-only	Audiovisual
Early fusion	Shared	55.5	46.5	62.2
Shared	Shared	55.5	44.7	62.5
Separate	Shared	55.4	48.9	63.0
Mid-fusion	Shared	55.8	48.5	63.5
Mid-fusion	Early	55.5	48.5	63.3
Mid-fusion	Separate	55.5	47.4	63.4
Mid-fusion	Shared	55.8	48.5	63.5

by finetuning on target datasets. The decoder of the model is only used during pretraining, and is discarded for finetuning as shown in Fig. 1.

For unimodal finetuning, we simply reuse the pretrained encoder of our model, and feed it a sequence of either audio or video tokens. For audio, this corresponds to AST [30]. While for video, this is the unfactorised ViViT model [6].

For audiovisual finetuning, we can reuse the same encoder structure as pretraining for encoding both modalities. Another alternative is to reuse the current state-of-the-art model, MBT [52], and initialise it with our self-supervised pretrained parameters. The MBT model is analogous to our "Mid-fusion" approach, it consists of separate transformer encoders for the audio and video modalities, with lateral connections between the two modalities after a predefined number of layers. When we use the "Separate" encoding strategy, we can initialise each stream of MBT with the corresponding encoder from the pretrained model. And when we use the "Early fusion" or "Shared" encoding methods, we can initialise each stream of MBT with the same encoder weights which are then untied from each other during training. In fact, the MBT authors used this method for initialising from ImageNet-21K with supervised pretraining [52].

As detailed in the supplementary, and perhaps surprisingly, we found that audiovisual finetuning with the MBT model performed the best, regardless of the pretraining architecture. We therefore use this approach in all our audiovisual finetuning experiments. Morever, as MBT is the current state-of-the-art multimodal fusion model, it allows us to fairly compare our audiovisual self-supervised pretraining to supervised pretraining in state-of-the-art comparisons.

4.2. Ablation Studies

Pretraining architecture Table 1 compares the different pretraining architectures, described in Sec. 3.2 and 3.3. In this experiment, we pretrained all models from random initialisation on both audio and video on the VGGSound dataset using the "Joint Reconstruction" objective (Sec. 3.4) and the ViT-Base backbone for 400 epochs. We then evaluated the learned representations by finetuning on VGGSound. As shown in Tab. 1, we performed audio-only, video-only and audiovisual finetuning (where we fuse both

Table 2: Ablation study of different pretraining objectives. Models are pre-trained for 400 epochs on VGGSound with the "Early fusion" encoder, and "Shared" decoder architecture.

Objective	Audio-only	Video-only	Audiovisual
Joint reconstruction	55.5	46.5	62.2
Inpainting (video from audio)	51.5	39.9	58.4
Inpainting (audio from video)	52.5	38.1	58.2
Inpainting (both modalities)	54.1	38.6	58.4

Table 3: Comparison of single-modality and audiovisual pretraining on VGGSound. We use a ViT-Base model with 400 epochs of pretraining. AudioMAE and VideoMAE refer to an MAE pretrained only on audio and video respectively.

Pretraining	Audio only	Video only	Audiovisual
AudioMAE	55.7	42.1	58.3
VideoMAE	52.8	49.3	62.1
Audiovisual MAE	55.8	48.5	63.5

modalities) using the same pretrained model in all cases.

We observe that the different encoder architectures perform similarly for audio-only finetuning. However, there is more of a difference for video-only finetuning, with "Separate" and "Mid-fusion" performing the best. For audiovisual finetuning, we also find that the "Separate" and "Mid-fusion" encoder strategies perform the best, with a sizable difference to the other approaches. A reason for this is that these encoding strategies closely follow the architecture of MBT [52], the current state-of-the-art multimodal fusion model, which we initialise with our pretrained weights. Moreover, separate parameters and processing streams for each modality increase model capacity.

In terms of the decoder architecture, the bottom part of Tab. 1 shows that the "Shared" decoder strategy outperforms the "Early" and "Separate" approaches by a small margin. Parameter sharing between the two modalities (as in the "Shared" approach) may be more beneficial in the decoder than in the encoder, as it allows better coupling between audio and video: A shared decoder requires the encoded audio tokens to contain information from the video tokens (and vice versa), in order to be able to reconstruct both sets of tokens using the same parameters. Based on Tab. 1 (and additional AudioSet experiments in the supplementary), we use the "Mid-fusion" encoder, and "Shared" decoder for future experiments unless otherwise stated.

Objective Table 2 compares our "Joint reconstruction" and "Modality inpainting" objectives (Sec. 3.4). In this experiment, we fix our architecture to the "Early fusion" encoder and "Shared" decoder for simplicity. For modality inpainting, as we reconstruct audio tokens from the encoded video tokens, and vice versa, we cannot use "Separate" encoders – it is necessary for the encoded audio tokens to contain information about the video tokens and vice versa. And indeed, training did not converge in this setting.

Table 2 shows that using the "Joint reconstruction" objective outperforms "Modality inpainting". A possible explanation is that "Modality inpainting" is more challenging, as the model is tasked with cross-modal reconstruction without using any encoded tokens of the target modality.

Nevertheless, analysis of the inpainting objective revealed interesting insights. Reconstructing video from audio results in better video representations (as shown by the higher video-only finetuning number). Similarly, reconstructing audio from video yields better audio-only repre-

sentations. Inpainting both audio from video, and video from audio, was difficult to train, and also required careful learning rate tuning to converge (as illustrated by learning curves in the supplementary). Inpainting both modalities however yields better audio representations, but slightly worse video representations, than inpainting a single modality. Note that we use equal weights for the video inpainting and audio inpainting losses in order to reduce the number of pretraining hyperparameters. Future work remains to further improve the potential of this objective, for example by using a combination of our "Joint Reconstruction" and "Modality inpainting" losses instead.

As the "Joint reconstruction" objective outperforms "Modality inpainting", and is also simpler to implement and train, we use it for the remainder of our experiments.

Comparing multimodal and unimodal pretraining Table 3 compares our Audiovisual MAE to pretraining on audio only (AudioMAE), and video only (VideoMAE).

When finetuning with audio-only or video-only, our Audiovisual MAE performs similarly to AudioMAE and VideoMAE, respectively. However, we observe substantial benefits for audiovisual finetuning tasks where we improve by 5.2 and 1.4 points compared to AudioMAE and VideoMAE respectively. Note that when performing audiovisual finetuning with a single-modality MAE we initialise both modality-streams of MBT [52] with the parameters of either AudioMAE or VideoMAE, analogously to how the original authors initialised from ImageNet-pretrained models.

Therefore, we conclude that audiovisual pretraining is especially beneficial for audiovisual finetuning, and still effective if one is interested in unimodal downstream tasks.

Transferability of learned representations Our experiments thus far have pretrained and finetuned on the same dataset. To evaluate the transferability of learned representations, we pretrain and finetune across different datasets as shown in Tab. 4. In this experiment, we pretrain for an equivalent number of iterations on both datasets. This is 800 epochs on VGGSound, and 80 epochs on AudioSet as the dataset is 10 times larger than VGGSound.

When evaluating on either VGGSound or AudioSet, we observe small differences between pretraining on either dataset, indicating that the learned representations transfer across both datasets. As expected, pretraining and finetuning on the same dataset still produces the best results.

Table 4: Transferability of learned representations by pretraining and finetuning across different datasets. We pretrain for the equivalent number of iterations on both datasets using ViT-Large as the backbone. Following standard protocol, we report Top-1 accuracy for VGGSound, mean Average Precision for AudioSet, and Top-1 action accuracy for Epic Kitchens.

Pretrain \ Finetune	VGGSound	AudioSet	Epic Kitchens
VGGSound	**65.0**	51.2	**45.5**
AudioSet	64.7	**51.3**	43.5

Table 5: Effect of number of pretraining epochs, using ViT-Large. We pretrain on VGGSound, and evaluate with audiovisual finetuning on both VGGSound and Epic Kitchens, to evaluate the transferability of the learned representation.

Epochs	200	400	800	1200
VGGSound	63.2	63.9	**65.0**	64.9
Epic Kitchens	41.8	42.5	45.5	**46.0**

However, we observe larger differences on Epic Kitchens, where pretraining on VGGSound improves action accuracy by 2% compared to pretraining on AudioSet. Note that the domain of Epic Kitchens, which consists of videos taken by egocentric cameras in household environments, is quite different to that of the YouTube videos which comprise VGGSound and AudioSet. Therefore, evaluating transfer performance on Epic Kitchens is a challenging task.

The fact that pretraining on VGGSound performs better than the substantially larger AudioSet dataset suggests that the number of iterations of pretraining are more important than the actual size of the pretraining dataset, in line with some of the observations made by [70], and an additional experiment in our supplementary.

Effect of pretraining epochs Table 5 compares the effect of the number of pretraining epochs on downstream, audiovisual finetuning accuracy. When pretraining and finetuning on the same VGGSound dataset, we observe that accuracy saturates at 800 pretraining epochs. However, when finetuning on the Epic Kitchens dataset, we can see that pretraining for longer, up to 1200 epochs, is still beneficial and we do not yet see saturation. The consistent improvements from pretraining for longer are in line with previous works on unimodal masked autoencoders [26, 37, 70].

Scaling up the backbone Table 6 compares the ViT-Base and ViT-Large backbones for audiovisual finetuning on VGGSound. We consider two scenarios: First, pretraining with our proposed Audiovisual MAE on VGGSound, and second, using supervised pretraining with ImageNet-21K as done by the current state-of-the-art, MBT [52].

With our audiovisual pretraining, we observe a solid improvement from scaling up our model backbone from ViT-Base to ViT-Large. MBT [52] which uses supervised pretraining does not benefit from increasing the model back-

Table 6: Effect of scaling up the backbone architecture from ViT-Base to ViT-Large, for both supervised and self-supervised pretraining. We report audiovisual accuracy on VGGSound. Supervised pretraining initialisation results are obtained from MBT [52].

Initialisation	Base	Large
ImageNet-21K [52] (Supervised)	64.1	61.4
Audiovisual MAE (Self-supervised)	**64.2**	**65.0**

bone to ViT-Large, and in fact, experiences heavy overfitting which reduces its performance. Note that our improved accuracy is not due to additional regularisation during finetuning to counter overfitting, since we followed the public MBT code and used the same regularisers (stochastic depth [39], mixup [80], label smoothing [68]) as detailed in the supplementary. Our finding that masked pretraining produces more generalisable representations for finetuning larger models is also in line with [37].

Additional baseline We consider another baseline, where we train two separate MAE models, one audio-only and the other video-only on VGGSound for 800 epochs, and use this to initialise an audiovisual MBT model which we finetune on VGGSound. As detailed in the supplementary, this baseline obtains an audiovisual accuracy of 63.3 on VGGSound, compared to our Audiovisual MAE model which achieves 64.2, thereby showing the benefits of joint pretraining of both audio and video.

Qualitative Results We include visualisations of our reconstructions from pretraining in the supplementary.

4.3. State-of-the-art comparisons

We now compare our best models, using a ViT-Large backbone, on the audiovisual VGGSound, Epic Kitchens and AudioSet datasets. Note that these are only system-level comparisons as most prior, published works use supervised pretraining. In particular for unimodal finetuning, our architecture is a vanilla ViT model without the modality-specific designs used in other works. We are nevertheless able to achieve results surpassing, or competitive with, the state-of-the-art on a number of domains and modalities, showing the promise of our self-supervised pretraining. We also include additional state-of-the-art results on audiovisual event localisation [69] in the supplementary.

VGGSound Table 7a shows that we outperform published methods on the VGGSound dataset. Prior works on this dataset use supervised pretraining on ImageNet-21K. In contrast, we use self-supervised pretraining for 800 epochs, and do not use any additional labelled data. In the audio-only setting, we improve by 2.1 points over PolyViT [48] which was also trained on AudioSet, and by 4.9 points over MBT. In addition, we also achieve 65.0% Top-1 accuracy on audiovisual finetuning, improving upon [52] by 0.9 points.

Epic Kitchens We now transfer our VGGSound pretrained model from the previous experiment to the Epic

Table 7: State-of-the-art comparisons on audiovisual datasets. Previous methods use supervised pretraining on additional data including ImageNet-21K (Im21K), ImageNet-1K (Im1K), AudioSet (AS) and Kinetics 400 (K400). Our approach, in contrast, is self-supervised, and uses no labelled data beyond the target dataset. We denote audio-only, video-only and audiovisual finetuning as "A", "V" and "AV" respectively. The top-scoring entry is in bold, whilst the second-highest is underlined.

(a) VGGSound. We report Top-1 accuracy.

Epochs	Pretraining	A	V	AV
Kazakos *et al.* [45]	Sup. Im1K	52.5	–	–
PlayItBack [66]	Sup. Im21K	53.7	–	–
PolyViT [48]	Sup. Im21K, AS	55.1	–	–
MBT [52]	Sup. Im21K	52.3	**51.2**	64.1
Ours	SSL VGGSound	**57.2**	50.3	**65.0**

(b) AudioSet. We report the mAP for audiovisual fusion models.

Epochs	Pretraining	Training set	A	V	AV
GBlend [75]	Im1K	AS-2M	32.4	18.8	41.8
Perceiver [43]	None	AS-2M	38.4	25.8	44.2
PerceiverIO [42]	None	AS-2M	–	–	44.9
Fayek *et al.* [25]	Im1K	AS-2M	38.4	25.7	46.2
MBT [52]	Im21K	AS-500K	41.5	**31.3**	49.6
Ours	SSL AS-2M	AS-500K	**46.6**	31.1	**51.8**

(c) Epic Kitchens. We report Top-1 accuracies for verbs, nouns and actions (pairs of verbs and nouns).

Method	Pretraining	Audio			Video			Audiovisual		
		Verb	Noun	Action	Verb	Noun	Action	Verb	Noun	Action
Damen *et al.* [19]	Sup. Im1K	42.6	22.4	14.5	–	–	–	–	–	–
Kazakos *et al.* [45]	Sup. VGGSound	46.1	23.0	15.2	–	–	–	–	–	–
PlayItBack [66]	Sup. Im21K	47.0	23.1	15.9	–	–	–	–	–	–
TSM [49]	Sup. Im1K + K400	–	–	–	67.9	49.0	38.3	–	–	–
ViViT-L Fact. Encoder [6]	Sup. Im21K + K400	–	–	–	66.4	56.8	44.0	–	–	–
MotionFormer [57]	Sup. Im21K + K400	–	–	–	67.0	58.5	44.5	–	–	–
MTV [77]	Sup. Im21K + K400	–	–	–	67.8	**60.5**	**46.7**	–	–	–
MBT [52]	Sup. Im21K	44.3	22.4	13.0	62.0	56.4	40.7	64.8	**58.0**	43.4
Ours	SSL VGGSound	**52.7**	**27.2**	**19.7**	**70.8**	55.9	45.8	**71.4**	56.4	**46.0**

Kitchens dataset. Epic Kitchens consists of egocentric videos, and hence it presents a challenging domain shift compared to the YouTube videos in our pretraining dataset. Nevertheless, Tab. 7c shows that we outperform MBT substantially by 2.6 points on audiovisual finetuning. On audio, we substantially outperform recent work [66] by 3.8 points. This is likely because our self-supervised pretraining provides better initialisation than the ImageNet-pretrained weights typically used by the audio community.

For video-only finetuning, our architecture corresponds to an unfactorised ViViT [6]. We, however, still outperform the ViViT Factorised Encoder [6] which the authors showed was more accurate than the unfactorised model. The key difference is our initialisation – our model is initialised with self-supervised pretraining, whereas the other video models in Tab. 7c used supervised pretraining, first on ImageNet-21K and then Kinetics 400 [6, 57, 77]. And it is this self-supervised initialisation which enables better generalisation of our model to Epic Kitchens. Only recent work [77], outperforms our method for video-only finetuning with a specialised, multi-view architecture and additional supervised pretraining (ImageNet-21K and Kinetics 400).

Note that previous transformer models on this dataset [6, 52, 57, 77], are pretrained on ImageNet-21K and have a bias towards "noun" classes, performing well on these, and poorly on "verb" classes, compared to previous CNN models [49]. Intuitively, this is because the ImageNet dataset is labelled with object classes corresponding to nouns, and models finetuned from this initialisation perform well on nouns, but struggle on verbs. Our self-supervised pretraining, in contrast, does not utilise class labels, and performs significantly better on verb classes across all modalities.

AudioSet Table 7b compares our model to other published audiovisual models on AudioSet. We performed self-supervised pretraining on the full AudioSet-2M (AS2M). Following MBT [52], we finetuned on the AS500K training subset, which is slightly more balanced than the full AS2M. Other methods which finetune on the full AS2M have used complex minibatch sampling techniques [75], which we can avoid due to our use of AS500K. We improve substantially upon previous methods on audiovisual finetuning by 2.2 points, and audio-only finetuning by 5.1 points. Note that we have reported the recently corrected results of [52].

5. Conclusion and Future Work

We have proposed Audiovisual MAE, a simple and effective self-supervised approach for learning powerful and generalisable audiovisual representations. The efficacy of our approach is demonstrated by our state-of-the-art results, in both audiovisual and unimodal downstream tasks, on the VGGSound, AudioSet and Epic Kitchens datasets.

In future work, we aim to leverage more powerful backbone architectures than a standard vision transformer [24] and improve our modality inpainting objective, for example, by combining it with the "Joint reconstruction" loss term.

References

[1] Hassan Akbari, Liangzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. VATT: Transformers for multimodal self-supervised learning from raw video, audio and text. In *NeurIPS*, 2021. 2

[2] Jean-Baptiste Alayrac, Adria Recasens, Rosalia Schneider, Relja Arandjelović, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. In *NeurIPS*, 2020. 2

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

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

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

[6] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. ViViT: A video vision transformer. In *ICCV*, 2021. 2, 3, 5, 8

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

[8] Yuki M. Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *ICLR*, 2020. 2

[9] Roman Bachmann, David Mizrahi, Andrei Atanov, and Amir Zamir. MultiMAE: Multi-modal multi-task masked autoencoders. In *ECCV*, 2022. 2

[10] Hangbo Bao, Li Dong, and Furu Wei. BEiT: BERT pretraining of image transformers. In *ICLR*, 2022. 1, 2, 3

[11] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. In *arXiv preprint arXiv:2108.07258*, 2021. 1

[12] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. 5

[13] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *NeurIPS*, 2020. 1

[14] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018. 2

[15] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jegou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021. 2

[16] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. VGGSound: A large-scale audio-visual dataset. In *ICASSP*, 2020. 4

[17] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 2

[18] Dading Chong, Helin Wang, Peilin Zhou, and Qingcheng Zeng. Masked spectrogram prediction for self-supervised audio pre-training. In *arXiv preprint arXiv:2204.12768*, 2022. 2

[19] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, Jian Ma, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Rescaling egocentric vision: Collection, pipeline and challenges for EPIC-KITCHENS-100. *IJCV*, 2022. 4, 5, 8

[20] Mostafa Dehghani, Alexey Gritsenko, Anurag Arnab, Matthias Minderer, and Yi Tay. Scenic: A JAX library for computer vision research and beyond. In *CVPR Demo*, 2022. 5

[21] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009. 1, 3

[22] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019. 1, 2, 5

[23] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *ICCV*, 2015. 2

[24] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 1, 2, 3, 5, 8

[25] Haytham M Fayek and Anurag Kumar. Large scale audiovisual learning of sounds with weakly labeled data. In *arXiv preprint arXiv:2006.01595*, 2020. 8

[26] Christoph Feichtenhofer, Haoqi Fan, Yanghao Li, and Kaiming He. Masked autoencoders as spatiotemporal learners. In *arXiv preprint arXiv:2205.09113*, 2022. 1, 2, 3, 4, 5, 7

[27] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio Set: An ontology and human-labeled dataset for audio events. In *ICASSP*, 2017. 4, 5

[28] Rohit Girdhar, Alaaeldin El-Nouby, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. OmniMAE: Single model masked pretraining on images and videos. In *arXiv preprint arXiv:2206.08356*, 2022. 2

[29] Rohit Girdhar, Mannat Singh, Nikhila Ravi, Laurens van der Maaten, Armand Joulin, and Ishan Misra. Omnivore: A single model for many visual modalities. In *CVPR*, 2022. 2

[30] Yuan Gong, Yu-An Chung, and James Glass. AST: Audio spectrogram transformer. In *Interspeech*, 2021. 3, 5

[31] Yuan Gong, Yu-An Chung, and James Glass. PSLA: Improving audio tagging with pretraining, sampling, labeling, and aggregation. *Transactions on Audio, Speech, and Language Processing*, 2021. 5

[32] Yuan Gong, Andrew Rouditchenko, Alexander H Liu, David Harwath, Leonid Karlinsky, Hilde Kuehne, and James Glass. Contrastive audio-visual masked autoencoder. In *arXiv preprint arXiv:2210.07839*, 2022. 2

[33] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap your own latent - a new approach to self-supervised learning. In *NeurIPS*, 2020. 2

[34] Andrey Guzhov, Federico Raue, Jörn Hees, and Andreas Dengel. ESResNet: Environmental sound classification based on visual domain models. In *ICPR*, 2021. 3

[35] Grzegorz Gwardys and Daniel Grzywczak. Deep image features in music information retrieval. *International Journal of Electronics and Telecommunications*, 2014. 3

[36] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006. 2

[37] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. 1, 2, 3, 4, 5, 7

[38] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 2

[39] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Weinberger. Deep networks with stochastic depth. In *ECCV*, 2016. 7

[40] Po-Yao Huang, Vasu Sharma, Hu Xu, Chaitanya Ryali, Haoqi Fan, Yanghao Li, Shang-Wen Li, Gargi Ghosh, Jitendra Malik, and Christoph Feichtenhofer. Mavil: Masked audio-video learners. In *arXiv preprint arXiv:2212.08071*, 2022. 2

[41] Po-Yao Huang, Hu Xu, Juncheng Li, Alexei Baevski, Michael Auli, Wojciech Galuba, Florian Metze, and Christoph Feichtenhofer. Masked autoencoders that listen. In *arXiv preprint arXiv:2207.06405*, 2022. 2, 5

[42] Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, et al. Perceiver IO: A general architecture for structured inputs & outputs. In *ICLR*, 2022. 8

[43] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *ICML*, 2021. 2, 4, 8

[44] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014. 3

[45] Evangelos Kazakos, Arsha Nagrani, Andrew Zisserman, and Dima Damen. Slow-fast auditory streams for audio recognition. In *ICASSP*, 2021. 8

[46] Bruno Korbar, Du Tran, and Lorenzo Torresani. Cooperative learning of audio and video models from self-supervised synchronization. In *NeurIPS*, 2018. 2

[47] Kuang-Huei Lee, Anurag Arnab, Sergio Guadarrama, John Canny, and Ian Fischer. Compressive visual representations. In *NeurIPS*, 2021. 2

[48] Valerii Likhosherstov, Anurag Arnab, Krzysztof Choromanski, Mario Lucic, Yi Tay, Adrian Weller, and Mostafa Dehghani. PolyViT: Co-training vision transformers on images, videos and audio. In *arXiv preprint arXiv:2111.12993*, 2021. 2, 7, 8

[49] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *CVPR*, 2019. 8

[50] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens Van Der Maaten. Exploring the limits of weakly supervised pretraining. In *ECCV*, 2018. 1

[51] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *CVPR*, 2020. 2

[52] Arsha Nagrani, Shan Yang, Anurag Arnab, Aren Jansen, Cordelia Schmid, and Chen Sun. Attention bottlenecks for multimodal fusion. In *NeurIPS*, 2021. 2, 4, 5, 6, 7, 8

[53] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. In *arXiv preprint arXiv:1807.03748*, 2018. 2

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

[55] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016. 2

[56] Mandela Patrick, Yuki M Asano, Polina Kuznetsova, Ruth Fong, João F Henriques, Geoffrey Zweig, and Andrea Vedaldi. On compositions of transformations in contrastive self-supervised learning. In *ICCV*, 2021. 2

[57] Mandela Patrick, Dylan Campbell, Yuki Asano, Ishan Misra, Florian Metze, Christoph Feichtenhofer, Andrea Vedaldi, and João F Henriques. Keeping your eye on the ball: Trajectory attention in video transformers. In *NeurIPS*, 2021. 8

[58] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 1

[59] Merey Ramazanova, Victor Escorcia, Fabian Caba Heilbron, Chen Zhao, and Bernard Ghanem. OWL (observe, watch, listen): Localizing actions in egocentric video via audiovisual temporal context. In *BMVC*, 2022. 2

[60] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*, 2021. 2

[61] Ladan Shams and Robyn Kim. Crossmodal influences on visual perception. *Physics of life reviews*, 7(3):269–284, 2010. 1

[62] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *NeurIPS*, 2014. 3

[63] Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. Flava: A foundational language and vision alignment model. In *CVPR*, 2022. 1

[64] Linda Smith and Michael Gasser. The development of embodied cognition: Six lessons from babies. *Artificial life*, 11(1-2):13–29, 2005. 1

[65] Shaden Smith, Mostofa Patwary, Brandon Norick, Patrick LeGresley, Samyam Rajbhandari, Jared Casper, Zhun Liu, Shrimai Prabhumoye, George Zerveas, Vijay Korthikanti, et al. Using DeepSpeed and Megatron to train Megatron-Turing NLG 530B, a large-scale generative language model. In *arXiv preprint arXiv:2201.11990*, 2022. 1

[66] Alexandros Stergiou and Dima Damen. Play it back: Iterative attention for audio recognition. In *arXiv preprint arXiv:2210.11328*, 2022. 8

[67] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *ICCV*, 2017. 1, 3

[68] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016. 7

[69] Yapeng Tian, Jing Shi, Bochen Li, Zhiyao Duan, and Chenliang Xu. Audio-visual event localization in unconstrained videos. In *ECCV*, 2018. 7

[70] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. VideoMAE: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *arXiv preprint arXiv:2203.12602*, 2022. 1, 2, 3, 5, 7

[71] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2, 3

[72] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *ICML*, 2008. 2

[73] Luyu Wang, Pauline Luc, Adria Recasens, Jean-Baptiste Alayrac, and Aaron van den Oord. Multimodal self-supervised learning of general audio representations. In *arXiv preprint arXiv:2104.12807*, 2021. 2

[74] Rui Wang, Dongdong Chen, Zuxuan Wu, Yinpeng Chen, Xiyang Dai, Mengchen Liu, Yu-Gang Jiang, Luowei Zhou, and Lu Yuan. Bevt: Bert pretraining of video transformers. In *CVPR*, 2022. 2

[75] Weiyao Wang, Du Tran, and Matt Feiszli. What makes training multi-modal classification networks hard? In *CVPR*, 2020. 4, 8

[76] Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. In *CVPR*, 2022. 1, 4

[77] Shen Yan, Xuehan Xiong, Anurag Arnab, Zhichao Lu, Mi Zhang, Chen Sun, and Cordelia Schmid. Multiview transformers for video recognition. In *CVPR*, 2022. 5, 8

[78] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. In *arXiv preprint arXiv:2111.11432*, 2021. 1

[79] Rowan Zellers, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. MERLOT: Multimodal neural script knowledge models. In *NeurIPS*, 2021. 2

[80] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018. 7

[81] Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization. In *ECCV*, 2016. 2