Multimodal Pathway: Improve Transformers with Irrelevant Data from Other Modalities

Yiyuan Zhang[1] Xiaohan Ding[2] Kaixiong Gong[1] Yixiao Ge[2] Ying Shan[2] Xiangyu Yue[1*]
[1]MMLab, The Chinese University of Hong Kong [2]Tencent AI Lab
yiyuanzhang.ai@gmail.com, xiaohding@gmail.com, xyyue@ie.cuhk.edu.hk
https://ailab-cvc.github.io/M2PT/

Abstract

We propose to improve transformers of a specific modality with irrelevant data from other modalities, e.g.*, improve an ImageNet model with audio or point cloud datasets. We would like to highlight that the data samples of the target modality are irrelevant to the other modalities, which distinguishes our method from other works utilizing paired (*e.g.*, CLIP) or interleaved data of different modalities. We propose a methodology named Multimodal Pathway - given a target modality and a transformer designed for it, we use an auxiliary transformer trained with data of another modality and construct pathways to connect components of the two models so that data of the target modality can be processed by both models. In this way, we utilize the universal sequence-to-sequence modeling abilities of transformers obtained from two modalities. As a concrete implementation, we use a modality-specific tokenizer and task-specific head as usual but utilize the transformer blocks of the auxiliary model via a proposed method named Cross-Modal Re-parameterization, which exploits the auxiliary weights without any inference costs. On the image, point cloud, video, and audio recognition tasks, we observe significant and consistent performance improvements with irrelevant data from other modalities. The code and models are available at* https://github.com/AILab-CVC/M2PT.

1. Introduction

Transformers [12, 14, 36, 37] are widely adopted in various tasks across modalities, such as text classification [8], object detection [3], point cloud analysis [47], and audio spectrogram recognition [16]. Apart from numerous unimodal tasks, transformers are also effective on multimodal data, *e.g.*, CLIP [32] uses image-text pairs to achieve superior performance in image recognition. Transformers' success in multiple modalities demonstrates their abilities to universally establish sequence-to-sequence modeling, given the input sequences (*i.e.*, tokens) which can be seen as the universal embeddings of data of multiple modalities [3, 12, 16, 46, 47]. For brevity, we refer to such ability as the *universal modeling ability*.

We would like to note that CLIP [32] represents the significant success of a methodology that improves a model's performance on a certain modality (*i.e.*, image) with the help of data from another modality (*i.e.*, text), but the limitation is also apparent - **the data samples from the two modalities must be relevant** (*e.g.*, paired, in this case). This limitation seems so inevitable that it hardly attracts research interest from the literature. Taking another two modalities, image and audio, as an example, we may expect that training with image-audio pairs may help the model recognize images (if we build a dataset with enough image-audio pairs and re-design the model to use the audio labels as the supervision, just like CLIP does with image-text pairs), but **it seems hard to believe that a pure audio dataset would improve a model's performance on ImageNet classification without any relevance between the audio and image samples**.

In this paper, we propose to improve a transformer's performance on a certain modality even with irrelevant data from another modality, as shown in Figure 1. The motivation is that we can see a training process on a certain modality as converting the data of the modality to sequences (*i.e.*, tokens) and establishing sequence-to-sequence mappings with the transformer blocks. For a specific modality, we reckon that the trained model has knowledge encoded in the sequence-to-sequence modeling that can facilitate another modeling process whose input sequences are obtained from another modality. In other words, apart from the obvious modality-specific knowledge acquired through training on a specific modality, we seek the **modality-complementary knowledge of sequence-to-sequence modeling in transformers** and will show that **it does exist**.

*Corresponding Author

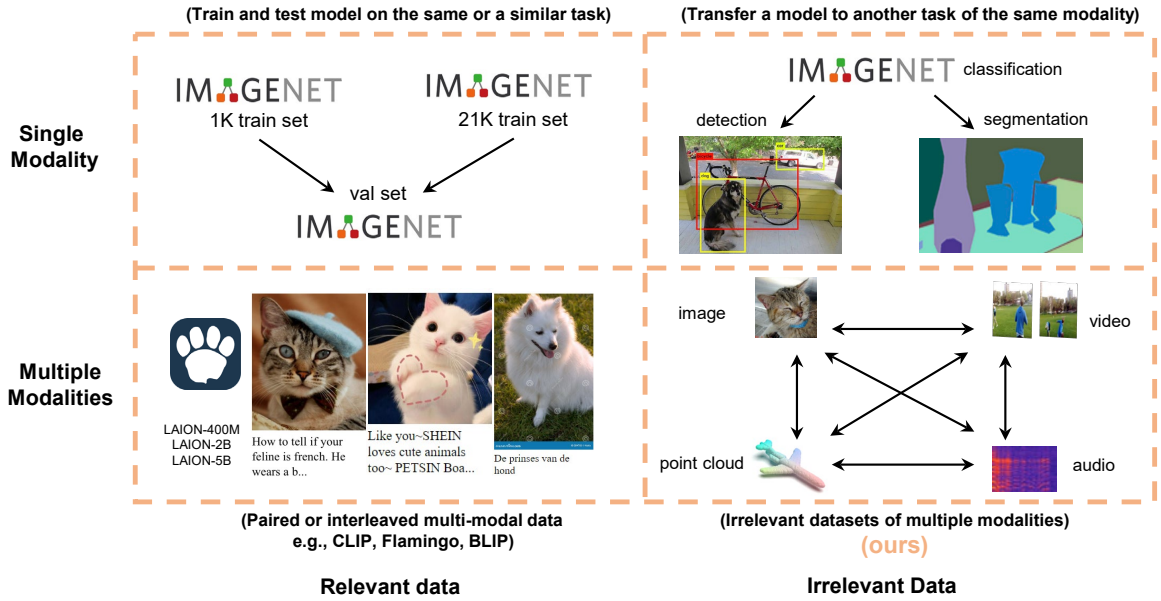

Figure 1. Compared to the known paradigms which use well-aligned multimodal data, we focus on scenarios where the data samples are from multiple modalities but irrelevant, which is an open problem in the literature.

However, given a target modality, it seems difficult to design the model to utilize some irrelevant data of another modality because the data samples of different modalities (*e.g.*, image and audio) may vary significantly in the semantics, data format, preprocessing, and it seems hardly possible to design a reasonable objective function since there is no relevance between any two samples. In this paper, we solve this problem by not directly mixing training data of two modalities but *seeing a model trained on a specific unimodal dataset as a proxy of the corresponding modality and using the model instead*. Specifically, given a target modality and an auxiliary modality, we propose a framework named *Multimodal Pathway* to improve the performance on the target modality by *using two transformers respectively trained with the unimodal data of the two modalities*. We construct *pathways* across the components of the target and auxiliary models to exploit the modality-complementary knowledge encoded in the latter to help the former. Note pathway is an abstract concept that may refer to any connection between the two models. We name the model as **M**ulti**m**odal **P**athway **T**ransformer (**M2PT**) for brevity.

This paper proposes a simple yet effective implementation of M2PT, where the key is the concrete implementation of pathways that connect the two models. As discussed above, thanks to the universal modeling ability, transformers on different modalities may have different tokenizers, but their main bodies (*i.e.*, transformer blocks) may have the same structure. [1] For a target model and an auxiliary model with the same structure as the main bodies, a layer in the main body of the former should have a counterpart in the latter. For example, the counterpart of the Query layer in the 9th block of the target model, which is the 9th Query layer in the auxiliary model, should exist, and they play a similar role in the two models. Considering this, we build the connections between the two models by augmenting every linear layer in the transformer blocks of the target model with its counterpart in the auxiliary model. In such a conceptual design, we let the two layers take the same inputs and add up their outputs, as shown in Figure 2 (middle).

However, considering the budget on compute and latency, we desire an implementation of the Multimodal Pathway that realizes the pathways and makes good use of the auxiliary model but *brings only marginal training cost and completely no inference cost*, compared to a regular model trained on the target modality. We note that the conceptual structure described above can be equivalently implemented by a re-parameterization method, which equivalently converts the connections between model structures (*i.e.,* linear layers) into connections between the two models' weights. Specifically, we construct a pathway for each target linear layer by adding the corresponding weights of its counterpart in the trained auxiliary model scaled by a learnable multiplier that indicates the strength of the pathway, so that the method is named *Cross-Modal Re-parameterization*. A

[1] Except for transformers, a recent work, UniRepLKNet [11], reveals ConvNets also effectively handle embeddings extracted from different modalities with the same architecture (akin to transformers universally tokenizing and processing data of multiple modalities), achieving state-of-the-art performances in tasks including global weather forecasting and audio recognition.

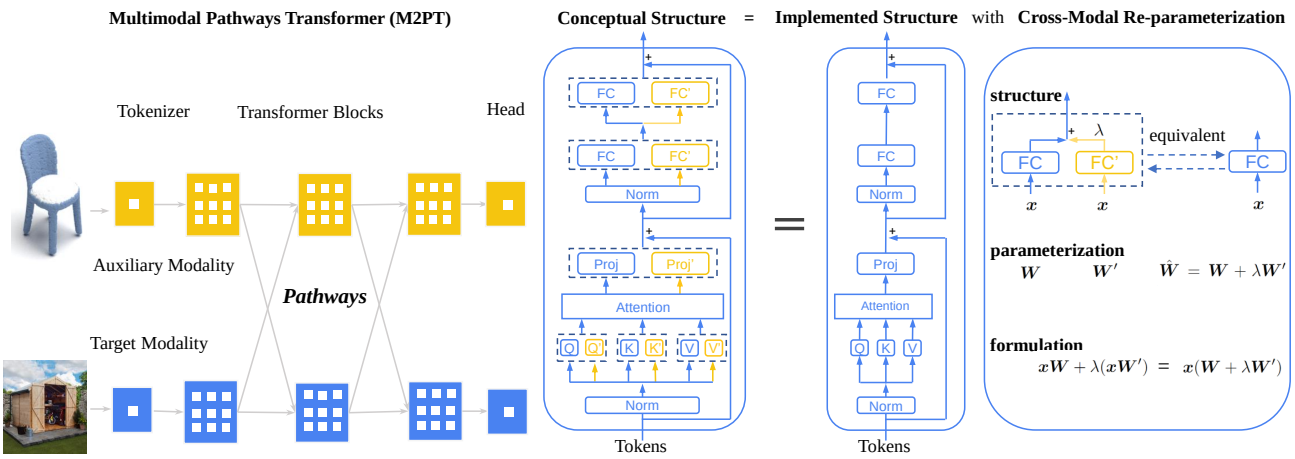

Figure 2. (**Left**) Framework of Multimodal Pathway Transformer (M2PT). We use point cloud and image modalities as an example. Common practices with transformers follow the same pipeline: using 1) tokenizers to convert the input data to sequences, 2) transformer blocks to process the sequences, and 3) heads to decode the sequences. We upgrade the sequence-to-sequence modeling by establishing *pathways* between the components of different modalities so processing the tokens of a specific modality can utilize the transformer blocks trained with another modality. (**Middle**) Conceptual design of M2PT, where the pathways are implemented by letting a linear layer (including the Query/Key/Value/projection layers in the attention block and those in the FFN block) in the target model cooperate with its counterpart in the auxiliary model. (**Right**) Cross-Modal Re-parameterization efficiently realizes M2PT by re-parameterizing the weights of the target model with those of the auxiliary model, introducing marginal training costs and completely no inference costs.

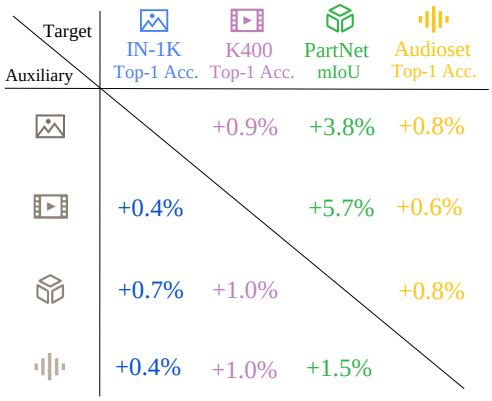

Figure 3. Consistent improvements brought by M2PT across each pair of four modalities - image, video, point cloud, and audio. The metrics are ImageNet-1K accuracy, Kinetics-400 accuracy, PartNet mIoU, and AudioSet accuracy, respectively. The numbers represent the percentage of improvement of M2PT models relative to the performance of baseline models that are pretrained with MAE-style methods [22, 23, 30, 49] on the four modalities, respectively.

significant strength of re-parameterization is that the extra training costs are marginal (*i.e.*, the re-parameterized model will have the same number of linear layers as the original model, and each linear layer merely needs to compute the sum of two weight matrices before projecting the inputs) and we can merge the weights after training so that the structure and number of parameters of the resultant model will be identical to a regular model.

We experimented with the image, video, point cloud, and audio modalities. Figure 3 shows the relative improvements M2PT consistently brings among four modalities. Such results reveal that the modality-complementary knowledge of sequence-to-sequence modeling in transformers does exist. As an early exploration, our empirical studies confirm that such improvements are not solely due to the more parameters, and suggest that such modality-complementary knowledge may be related to the ability to generally process hierarchical representations. Abstraction hierarchy exists in multiple modalities with concepts ranging from low-level to high-level, which may explain the universality of the learned knowledge. In other words, as a transformer is being trained with images, it learns both (ability A) how to understand images and (ability B) how to generally transform the tokens from the lower-level patterns to a higher level without assuming they originally come from images. Meanwhile, as another transformer is being pretrained with audio data, it learns both a different "ability A" for audio and a similar "ability B", so that it can help the aforementioned transformer in image recognition.

In summary, our contributions are as follows:

- We propose Multimodal Pathway, which is a framework to improve transformers via exploiting models trained on other modalities.
- We propose an inference-cost-free implementation of Multimodal Pathway, which is named Cross-Modal Re-parameterization.
- Multimodal Pathway represents an early exploration in this direction, which offers a novel perspective. We re-

alize significant and consistent improvements in four representative modalities, which demonstrates the potential of our method as a promising approach.

2. Related Work

Unimodal pretraining. The evolution of unimodal pretraining paradigms has transitioned from supervised to self-supervised paradigms. For instance, Devlin et al. [8] introduced the mask-reconstruction paradigm and achieved remarkable outcomes. At that time, visual pretraining largely emphasized contrastive learning [4, 6, 21]. Subsequently, leveraging the vast amounts of unlabeled data, the BERT paradigm gained traction and pioneers like MAE [22] successfully applied it to visual pretraining, while others [16, 30, 35, 46] extended this paradigm to areas like point cloud, audio, and video perception.

We use MAE-style unimodal pretraining methods to obtain the weights on each modality for simplicity. We do not use supervised pretraining because we would like to ensure that two unimodal datasets are completely irrelevant by avoiding using labels, considering that the labels of two datasets may somehow overlap.

Multimodal pretraining. Existing multimodal learning methods require paired [19, 39, 40, 50] or interleaved data [1]. In either case, the data samples of different modalities are well-aligned (*i.e.*, strongly related). A recent study highlighted a main trend in the literature - *existing multimodal pretraining methods are overly dependent on the well-aligned multimodal sample pairs/tuples* [43]. For instance, VideoBERT [34] and CBT [33] utilize well-aligned video and speech data;

Nowadays, using the weakly-aligned or unpaired/unaligned multimodal data as the pretraining corpora remains understudied [43]. This work represents an early exploration in this direction, which serves to fill this gap in the field and contributes to multimodal calibration [38].

Structural Re-parameterization is a methodology that constructs extra structures (*e.g.*, convolutional layers) during training and converts the trained structures via transforming the parameters [9–11]. A primary drawback of Structural Re-parameterization is that the constructed layers must participate in the computations with the inputs, resulting in significant extra training costs.

In contrast, Cross-Modal Re-parameterization is a simple re-parameterization method that is more efficient than Structural Re-parameterization. Specifically, the extra computation of each re-parameterized layer in the forward computation adds up two weight matrices,

3. Method

3.1. Architectural Design

We design a transformer for a specific modality as three modules - the modality-specific tokenizer, the modality-agnostic transformer blocks, and the modality-specific head. We assume the dimension of tokens is D, which is a pre-defined architectural hyper-parameter, and describe how to tokenize the input data of multiple modalities into D-dimensional tokens.

Image tokenizer. We represent an image by $\boldsymbol{x}_I \in \mathbb{R}^{H\times W\times C}$, where (H, W) specifies the image's resolution, and C is the number of channels. With an image patch of (S, S), we obtain:

$$\boldsymbol{x}_I \in \mathbb{R}^{H\times W\times C} \rightarrow \boldsymbol{x}'_I \in \mathbb{R}^{\frac{HW}{S^2}\times D}. \tag{1}$$

Video tokenizer. Analogous to 2D images, we use video patches as the basic units for learning video representations. Given an N-frame video $\boldsymbol{x} \in \mathbb{R}^{N\times H\times W\times C}$, similar to images, we use an $S\times S$ embedding layer so that

$$\boldsymbol{x}_V \in \mathbb{R}^{N\times H\times W\times C} \rightarrow \boldsymbol{x}'_V \in \mathbb{R}^{\frac{NHW}{S^2}\times D}. \tag{2}$$

Following ViT [12], we use $S = 16$ by default.

Point cloud tokenizer. Given a point cloud $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^{P}$ comprising P points, each point $\boldsymbol{x}_i$ is defined as $\boldsymbol{x}_i = (\boldsymbol{p}_i, \boldsymbol{f}_i)$, where $\boldsymbol{p}_i \in \mathbb{R}^3$ denotes the 3D coordinates and $\boldsymbol{f}_i \in \mathbb{R}^c$ encodes the attributes, *e.g.*, color, viewpoint, normal, *etc*. We use the Farthest Point Sampling to sample a representative skeleton from the original points at a fixed sampling ratio of 1/4, then K-Nearest Neighbor method to group proximate points. Then we model the geometric relevance by constructing an adjacency matrix $\mathbb{R}^{\frac{P}{4}\times\frac{P}{4}}$ between each pair of groups, which is then projected into D-dimensional tokens. That is

$$\boldsymbol{x}_P \in \mathbb{R}^{P\times(3+c)} \rightarrow \boldsymbol{x}'_P \in \mathbb{R}^{\frac{P}{4}\times\frac{P}{4}} \rightarrow \boldsymbol{x}''_P \in \mathbb{R}^{\frac{P}{4}\times D}. \tag{3}$$

Audio spectrogram tokenizer. Let T and F be the numbers of time frames and frequency bins, we use $\boldsymbol{x}_A \in \mathbb{R}^{T\times F}$ to represent a sample. Analogous to 2D images, we see an audio sample as a single-channel image and use a similar embedding layer so that

$$\boldsymbol{x}_A \in \mathbb{R}^{T\times F} \rightarrow \boldsymbol{x}'_A \in \mathbb{R}^{\frac{TF}{S^2}\times D}. \tag{4}$$

In our AudioSet experiments, we have T=F=128, S=16.

Transformer blocks. We adopt the structural design of the transformer blocks in Vision Transformer (ViT) [12], where each transformer block comprises a self-attention block and a Feed-Forward Network (FFN) block. The linear layers include the Query/Key/Value/projection layers in the attention block and two layers in the FFN block. For fairness and reproducibility, we use the same architectural hyper-parameters (*e.g.*, dimension of tokens, number of blocks, and number of heads) as ViT-Base for every M2PT model on every modality.

3.2. Cross-Modal Re-parameterization

For an M2PT model on a specific modality, we use Cross-Modal Re-parameterization in the transformer blocks to utilize another model's weights trained on another modality. Specifically, let θ be an arbitrary trainable parameter of a layer in the transformer, x be the input, and y be the output, we use f to denote the operation so that $y = f(x;\theta)$. With Cross-Modal Re-parameterization, we simply re-parameterize the layer with parameters of its counterpart in another modal that is trained on another modality. Let θ' be the parameter of the counterpart, the operation becomes

$$y = f(x;\theta + \lambda\theta')\,. \tag{5}$$

We refer to λ as the *Cross-Modal Scale* and θ' as the *Cross-Modal Parameter*. After training, we merge the model by computing and saving $\hat{\theta} = \theta + \lambda\theta'$ so that the model will no longer have extra parameters and the inference costs and model size will be identical to a regular model.

With Cross-Modal Re-parameterization, we equivalently realize the proposed M2PT transformer block with marginal training costs and completely no inference costs. For a linear layer whose parameters form a matrix $\boldsymbol{W} \in \mathbb{R}^{D_{in}\times D_{out}}$ and the inputs and outputs are matrices $\boldsymbol{x} \in \mathbb{R}^{B\times D_{in}}$ and $\boldsymbol{y} \in \mathbb{R}^{B\times D_{out}}$. We omit the bias term for brevity and the original operation is formulated by

$$\boldsymbol{y} = \boldsymbol{x}\boldsymbol{W}\,. \tag{6}$$

As described in the conceptual structure depicted in Figure 2, the linear layer and its counterpart take the same input. The output will be

$$\boldsymbol{y} = \boldsymbol{x}\boldsymbol{W} + \lambda(\boldsymbol{x}\boldsymbol{W}')\,. \tag{7}$$

Note

$$\boldsymbol{x}\boldsymbol{W} + \lambda(\boldsymbol{x}\boldsymbol{W}') = \boldsymbol{x}(\boldsymbol{W} + \lambda\boldsymbol{W}')\,, \tag{8}$$

so that the two layers can be equivalently implemented by a single layer that has a trainable scalar λ and an additional trainable matrix which is initialized with the counterpart in the auxiliary model. Both the original weight matrix and the additional one are trainable. At each forward computation, the layer computes the equivalent weight matrix and then uses it to project the input, which is

$$\boldsymbol{y} = \boldsymbol{x}(\boldsymbol{W} + \lambda\boldsymbol{W}')\,. \tag{9}$$

After training, we merge the parameters by computing $\hat{\boldsymbol{W}} = \boldsymbol{W} + \lambda\boldsymbol{W}'$ and save it only. For inference, we simply construct a regular linear layer and load $\hat{\boldsymbol{W}}$.

In summary, to construct and use an M2PT with Cross-Modal Re-parameterization, we

- Construct the tokenizer and head according to the target modality.
- Construct the transformer blocks with Cross-Modal Re-parameterization. For each linear layer, except for the original weight matrix, we add an extra trainable weight matrix and initialize it with the corresponding one from a transformer trained on the auxiliary modality, and add a trainable scalar parameter initialized with 0.
- Train the re-parameterized cross-modal model just like we train a regular model.
- After training, convert the trained model and save the converted one for inference.

4. Experiments

4.1. Setup

Datasets. For image recognition, we evaluate the models' performance on three representative image datasets. 1) ImageNet-1K [7] contains nearly 1.3 million images of 1000 categories. 2) MSCOCO 2017 [27] is a common benchmark for object detection. M2PT is trained on the `train` set and evaluated on the `val` set with Mask RCNN [20]. 3) ADE-20K [48] is used for semantic segmentation experiments with UperNet [41] and we adopt the single-scale evaluation setting. For point cloud, we evaluate the performance of M2PT on ShapeNetPart [44], which contains 16,880 models and 16 categories. For audio recognition, following AudioMAE [23], we utilize the AudioSet-2k [15] dataset. For video, we experiment on the action recognition dataset, Kinetics-400 [24], which contains 240k training videos and 20k validation videos from 400 classes.
Experimental details. For a pair of target modality and auxiliary modality, we obtain the auxiliary model by self-supervised training on a dataset of the auxiliary modality. Specifically, the auxiliary image model is pretrained with MAE [22] on ImageNet-1K [7], the auxiliary point cloud model is pretrained with Point-MAE [30] on ShapeNet [5], the auxiliary audio model is pretrained with AudioMAE [23] on AudioSet-2M [15], the auxiliary video model is pretrained with VideoMAE [35] on Kinetics-700 [24]. We do not use supervised pretraining because we would like to eliminate the effects of labels in the pretraining datasets so that we can ensure the irrelevance of the data samples, considering that the labels of two datasets may somehow overlap. In terms of the initialization of the target model, we adopt two settings. 1) The target model (*i.e.*, the parameters denoted by $\boldsymbol{W}$ in Eq. 9) is initialized with the aforementioned weights pretrained with the self-supervised methods on the target modality. We finetune the M2PT model with the default finetuning configurations described by the corresponding pretraining methods. The baseline model is also initialized with the pretrained weights and fine-tuned with identical configurations so that this setting is referred to as the *pretrained setting* for brevity. 2) The

Table 1. **Experimental results on image recognition tasks.** On ImageNet, we report the results with the linear layers in transformer blocks finetuned (tune acc) or fixed (fix acc). The architecture of every model is ViT-B. The relative improvements over the baselines are shown in green. * The standard error of M2PT on image recognition tasks is 0.04.

Method	ImageNet		MS COCO		ADE20K
	tune acc(%)	fix acc(%)	AP_{box}(%)	AP_{mask}(%)	mIOU(%)
Pretrained setting					
SemMAE[25]	83.4	65.0	-	-	46.3
MFF [28]	83.6	67.0	48.1	43.1	47.9
MAE*[22]	83.3	65.6	47.3	42.4	46.1
M2PT-Video (Ours)	**83.6** ↑ 0.4%	**67.1** ↑ 2.3%	-	-	-
M2PT-Audio (Ours)	**83.7** ↑ 0.4%	**67.3** ↑ 2.6%	-	-	-
M2PT-Point (Ours)	**83.9** ↑ 0.7%	**67.8** ↑ 3.4%	**50.0** ↑ 5.7%	**44.0** ↑ 3.8%	47.9 ↑ 3.9%
From-scratch setting					
ViT [12]	76.5	14.5	46.2	40.5	39.7
M2PT-Point (Ours)	**81.9** ↑ 7.1%	**19.5** ↑ 34.5%	**48.9** ↑ 5.8%	**42.2** ↑ 4.2%	**42.5** ↑ 7.1%

target model is initialized as usual, and we use the widely adopted training configurations to train the M2PT model. The baseline model is trained from scratch with identical configurations for fair comparisons so that the setting is referred to as the *from-scratch setting* for brevity.

Metrics. We report the performance of M2PT models on various datasets, including top-1 accuracy for ImageNet-1K, AudioSet, Kinetics-400, mIoU for ADE20K, ShapeNetPart and PartNet, and box/mask AP for MS COCO. To fairly assess the performance improvements over the baselines in multiple metrics, we also report the relative percentage of improvement in Table 1, 2, 3, and 4.

4.2. Main Results

Image recognition. We first conduct a group of experiments under the pretrained setting, where the target weights are initialized with a ViT pretrained with MAE on ImageNet, and the auxiliary weights are from the models pretrained on video, audio, and point datasets, respectively. Such three models, which are labeled as M2PT-Video, M2PT-Audio, and M2PT-Point, respectively, and the baseline (the original MAE-pretrained ViT) are trained on ImageNet with the finetuning configurations originally adopted by MAE [22], and the resultant accuracies are reported in the "tune acc" column in Table 1. Then we transfer the best-performing model, which is M2PT-Point, to COCO object detection and ADE20K semantic segmentation tasks. The improvements are significant: the ImageNet accuracy improves from 83.3 to 83.9, the COCO box AP improves from 47.3 to 50.0, and the ADE20K mIoU improves from 46.1 to 47.9, so the relative improvements are 0.7%, 5.7%, and 3.9%, respectively.

Apart from finetuning the target and auxiliary weights, we test another setting where the parameters of linear weights in transformer blocks are fixed, and only the Cross-Modal Scales together with the classifier are trainable. The accuracies are reported in the "fix acc" column. Naturally, under this setting, the baseline should be the MAE-pretrained ViT where only the classifier is trainable. Impressively, the relative improvement becomes more significant (65.6→67.8 so that the relative improvement is 3.4%), demonstrating that the weights obtained from the auxiliary modality work on another modality, even if the weights are fixed. We would like to note MAE is a powerful pretraining method, and it is challenging to gain further improvements on top of MAE. Some insightful recent methods [25, 28] improved MAE but our results are more significant.

On the other hand, under the from-scratch setting, the baseline is a ViT trained from scratch, and the target weights of M2PT are also randomly initialized. The accuracy is drastically improved from 81.9 to 76.5 so the relative improvement is 7.1%, suggesting the auxiliary weights significantly facilitate the training process. Intuitively, the Cross-Modal Scales are initialized with 0 but will soon become non-zero as the training proceeds so the model will be gradually influenced by the auxiliary weights and benefit from the modality-complementary knowledge. When we transfer such two models to COCO and ADE20K, we observe consistent improvements in the box AP and mIoU.

3D point cloud understanding. Table 2 presents the experimental results on ShapeNetPart and PartNet datasets, where we compare M2PT with existing point cloud pretraining methods such as Point-BERT [30] and Point-MAE [45]. M2PT consistently improves the class mIoU from 84.2 to 85.6 and instance mIoU from 86.1 to 87.5 on ShapeNetPart and raises the mIoU from 47.4 to 50.1 on PartNet. Under the from-scratch setting, we also observe consistent improvements.

Audio recognition. For the pretrained setting, the tar-

Table 2. **Experimental results on point cloud datasets**. We report the class mIoU (mIoU$_C$) and instance mIoU$_I$ on ShapeNetPart and mIoU on PartNet. The relative improvements over the baselines are shown in green.

Method	ShapeNetPart		PartNet
	mIoU$_C$ (%)	mIoU$_I$ (%)	mIoU (%)
Pretrained setting			
PointNet++ [31]	81.9	85.1	42.5
Point-BERT [45]	84.1	85.6	-
Point-MLP [29].	84.6	86.1	48.1
Point-MAE [45]	84.2	86.1	47.4
M2PT-Video	**85.6** ↑ 1.7%	**87.5** ↑ 1.6%	**50.1** ↑ 5.7%
M2PT-Image	**85.6** ↑ 1.7%	**87.5** ↑ 1.6%	**49.2** ↑ 3.8%
M2PT-Audio	**85.6** ↑ 1.7%	**87.5** ↑ 1.6%	**48.1** ↑ 1.5%
From-scratch setting			
N/A	50.2	68.4	-
M2PT-Video	**50.8** ↑ 1.2%	**68.8** ↑ 0.6%	-

Table 3. **Experimental results on AudioSet-2k**. The relative improvements over the baselines are shown in green.

Method	Model	Top-1 Acc. (%)
Pretrained setting		
PSLA [17]	CNN+Trans	31.9
AST [16]	ViT-B	34.7
SSAST [18]	ViT-B	31.0
AudioMAE [23]	ViT-B	35.3
M2PT-Point	ViT-B	**35.6** ↑ 0.8%
M2PT-Video	ViT-B	**35.5** ↑ 0.6%
M2PT-Image	ViT-B	**35.6** ↑ 0.8%
From-scratch setting		
N/A	ViT-B	11.0
M2PT-Point	ViT-B	**11.4** ↑ 3.6%

Table 4. **Experimental results on Kinetics-400**. The relative improvements over the baselines are shown in green

Method	Model	Top-1 Acc. (%)
SlowFast-101 [13]	ResNet-101	79.8
MViTv2-B [26]	ViT-B	81.2
TimeSFormer [2]	ViT-B	80.7
VideoMAE [35]	ViT-B	81.5
M2PT-Point	ViT-B	**82.3** ↑ 1.0%
M2PT-Image	ViT-B	**82.2** ↑ 0.9%
M2PT-Audio	ViT-B	**82.3** ↑ 1.0%

get weights are initialized with an AudioMAE-pretrained model. As shown in Table 3, we compare M2PT with existing competitive methods including SSAST [18], AST [16], and AudioMAE [23]. M2PT improves the top-1 accuracy by 0.8% relatively on the Audioset balanced split, demonstrating that M2PT is also effective in audio recognition. Under the from-scratch setting, M2PT brings out a relative improvement of 3.6%.

Video understanding. For the experiments on Kinetics-400, we adopt only the pretrained setting because it is not a common practice to train a model from scratch on a video dataset, which would deliver inferior performance. We use the Video-MAE-pretrained ViT to initialize the target weights. Naturally, the baseline should be the VideoMAE-pretrained model directly finetuned on Kinetics-400. Table 4 shows that compared with previous works including SlowFast [13], MViTv2 [26], TimeSFormer [2], and VideoMAE [35], M2PT outperforms by at least +0.8 top-1 accuracy (82.3 vs. 81.5), which reveals that the temporal awareness for video understanding can also be enhanced with irrelevant data from other modalities.

4.3. Ablation Studies

As shown in Table 5, we evaluate the design choices of M2PT separately through a group of ablation studies under the pretrained setting on ImageNet and the auxiliary modality is the point cloud. We make the following observations.

1) Applying Cross-Modal Re-parameterization to every linear layer delivers the best performance. In each transformer block, we may choose to apply our method to any of the Query/Key/Value/projection layers in the attention block and the two linear layers in the FFN. Table 5 shows changing any one of the layers brings improvements, and the best result is achieved by changing them all.

2) Cross-Modal Scale should be initialized with 0. By default, we initialize the Cross-Modal Scale λ with 0 for every layer. We observe that initializing it to a higher value degrades the performance, suggesting that the initial state of the M2PT should be identical to the target weights (*i.e.*, the weights pretrained with MAE, in this case).

3) Cross-Modal Scale should be learnable. Fixing the Cross-Modal Scale degrades the performance, suggesting it is important to let the model learn how to combine the target weights and the corresponding auxiliary weights.

4.4. Empirical Discussions

4.4.1 On the Modality-Complementary Knowledge

The observed improvements on multiple modalities have shown that the auxiliary transformer has learned some knowledge that can be transferred to the target modality. We continue to investigate the properties of such modality-complementary knowledge through two groups of experiments (Table 6).

1) Modality-complementary knowledge & Abstraction Hierarchy. Vision Transformers excel in general hierarchical representations by stacking blocks [12]. For example, in the image and point cloud modalities, this hierarchy may include textures (in images) or individual points (in point clouds), object parts, and whole objects. In Table 6, we construct the multimodal pathway by connecting transformer blocks of different depths. Specifically, the counterpart of the first target block should be the first

Table 5. **Ablation studies** on design choices of M2PT including the layers to re-parameterize and configurations of Cross-Modal Scale λ. We use the point cloud and video as auxiliary modalities for image and 3D evaluation. The first row reports the results of direct tuning.

Multimodal Pathway Components				Cross-Modal Scale		ImageNet	ShapeNetPart	PartNet
Attn QKV	Attn Proj	FFN 1st	FFN 2nd	Init.	Trainable	(%)	(%)	(%)
				-	-	83.3	84.2/86.1	47.4
✔				0	✔	83.4	84.6/86.5	48.3
	✔			0	✔	83.6	84.8/87.1	48.2
		✔		0	✔	83.6	84.9/87.0	48.4
			✔	0	✔	83.7	85.2/87.2	48.3
✔	✔	✔	✔	0	✔	**83.9**	**85.6/87.5**	**50.1**
✔	✔	✔	✔	10^{-2}	✘	83.5	84.6/86.3	48.2
✔	✔	✔	✔	10^{-2}	✔	83.6	84.3/86.2	48.0
✔	✔	✔	✔	10^{-4}	✔	83.6	84.7/86.2	48.1
✔	✔	✔	✔	10^{-6}	✔	83.7	84.7/86.4	48.2

Table 6. ImageNet accuracy with changed order of auxiliary weights or fewer pretraining epochs.

Order of aux weights	Epochs pretrained	Top-1 acc
Normal	20	83.55
Normal	220	83.69
Normal	300	83.93
Reversed	300	83.61

Table 7. Training efficiency of Multimodal Pathway.

Model	Train Time	Train Param.	Inference Time	Inference Param.
MAE	16.95 Hours	86.3M	11.64 ms	86.3M
M2PT	22.84 Hours	172.6M	11.64ms	86.3M

auxiliary block. Under the reverse-order setting, we observe that doing so decreases the accuracy to 83.61%, which is 0.32% lower than the normal M2PT. We observe that modality-complementary knowledge in the auxiliary transformer can transfer to another modality but can be harmed if the low-to-high correspondence is interrupted, suggesting that modality-complementary knowledge reinforces hierarchical representations of the transformer architecture.

2) More trainable parameters? Just better initialization? For this part, we use insufficiently pretrained auxiliary weights. Specifically, the default auxiliary weights are pretrained for 300 epochs with mask modeling on point cloud data, but we alternatively use the checkpoints saved at the 20th and 220th epoch, respectively, as the auxiliary weights. Not surprisingly, we observe that the performance degrades to 83.55% and 83.69%, respectively, which is still higher than the baseline. This phenomenon suggests that the improvements brought by the auxiliary weights cannot be explained as better initialization, because after pretraining the auxiliary model from 20 to 300 epochs, the accuracy increases from 83.5 to 83.9. If improvements were due to initialization, the results of pretraining 20 epochs should be close to random initialization (83.5 *v.s.* 81.9).

4.4.2 Discussion on the Data Scale

1) From small-scale data to large-scale data. Previous works such as Image2Point [42] introduces image-pretrained models to data-insufficient 3D perception tasks. Differently, M2PT sets up a brand new methodology and breaks the former consensus - we discover that *even though the data scale of point clouds is limited, such data still brings out impressive improvements to the image, video, and audio perception tasks*. Impressively, the pretraining data of the latter modalities is larger in magnitude than that of the point cloud, but the point cloud data makes a difference. **2) From large-scale data to small-scale data.** On the other hand, the effectiveness of M2PT highlights that for 3D vision research and other areas that lack large-scale data for pretraining, M2PT introduces a promising direction to leverage irrelevant large-scale data from other modalities.

5. Conclusion and Limitation

This paper explores the feasibility and advantages of improving a transformer's performance on a specific modality with irrelevant data from other modalities. We propose the Multimodal Pathway and a concrete implementation of no additional inference cost named Cross-Modal Re-parameterization. It represents an early exploration in this direction, which offers a novel perspective. We realize significant and consistent improvements on four representative modalities, demonstrating the potential of our method as a promising approach. In the future, we will explore to construct multimodal pathways among CNNs and cross-architecture. The primary limitation is that the theory behind the improvements remains to be revealed. Apart from empirical explanations, we believe further investigations (*e.g.*, a mathematically provable bound) will be useful.

Acknowledgements. This work is partially supported by the National Natural Science Foundation of China (Grant No. 8326014).

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*, 2022. 4

[2] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, page 4, 2021. 7

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 213–229. Springer, 2020. 1

[4] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 4

[5] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv:1512.03012*, 2015. 5

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020. 4

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255. Ieee, 2009. 5

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2019. 1, 4

[9] Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun. Repvgg: Making vgg-style convnets great again. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13733–13742, 2021. 4

[10] Xiaohan Ding, Xiangyu Zhang, Jungong Han, and Guiguang Ding. Scaling up your kernels to 31x31: Revisiting large kernel design in cnns. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11963–11975, 2022.

[11] Xiaohan Ding, Yiyuan Zhang, Yixiao Ge, Sijie Zhao, Lin Song, Xiangyu Yue, and Ying Shan. Unireplknet: A universal perception large-kernel convnet for audio, video, point cloud, time-series and image recognition. *arXiv preprint arXiv:2311.15599*, 2023. 2, 4

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 1, 4, 6, 7

[13] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6202–6211, 2019. 7

[14] Chongjian Ge, Xiaohan Ding, Zhan Tong, Li Yuan, Jiangliu Wang, Yibing Song, and Ping Luo. Advancing vision transformers with group-mix attention. *arXiv preprint arXiv:2311.15157*, 2023. 1

[15] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 776–780. IEEE, 2017. 5

[16] Yuan Gong, Yu-An Chung, and James Glass. Ast: Audio spectrogram transformer. *arXiv preprint arXiv:2104.01778*, 2021. 1, 4, 7

[17] Yuan Gong, Yu-An Chung, and James Glass. Psla: Improving audio tagging with pretraining, sampling, labeling, and aggregation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3292–3306, 2021. 7

[18] Yuan Gong, Cheng-I Lai, Yu-An Chung, and James Glass. Ssast: Self-supervised audio spectrogram transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10699–10709, 2022. 7

[19] Jiaming Han, Kaixiong Gong, Yiyuan Zhang, Jiaqi Wang, Kaipeng Zhang, Dahua Lin, Yu Qiao, Peng Gao, and Xiangyu Yue. Onellm: One framework to align all modalities with language. *arXiv preprint arXiv:2312.03700*, 2023. 4

[20] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, pages 2961–2969, 2017. 5

[21] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020. 4

[22] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 3, 4, 5, 6

[23] Po-Yao Huang, Hu Xu, Juncheng Li, Alexei Baevski, Michael Auli, Wojciech Galuba, Florian Metze, and Christoph Feichtenhofer. Masked autoencoders that listen. *arXiv preprint arXiv:2207.06405*, 2022. 3, 5, 7

[24] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 5

[25] Gang Li, Heliang Zheng, Daqing Liu, Chaoyue Wang, Bing Su, and Changwen Zheng. Semmae: Semantic-guided masking for learning masked autoencoders. *Advances in Neural Information Processing Systems*, 35:14290–14302, 2022. 6

[26] Yanghao Li, Chao-Yuan Wu, Haoqi Fan, Karttikeya Mangalam, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Mvitv2: Improved multiscale vision transformers for

classification and detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4804–4814, 2022. 7

[27] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 5

[28] Yuan Liu, Songyang Zhang, Jiacheng Chen, Zhaohui Yu, Kai Chen, and Dahua Lin. Improving pixel-based mim by reducing wasted modeling capability. *arXiv preprint arXiv:2308.00261*, 2023. 6

[29] Xu Ma, Can Qin, Haoxuan You, Haoxi Ran, and Yun Fu. Rethinking network design and local geometry in point cloud: A simple residual mlp framework. *ICLR*, 2022. 7

[30] Yatian Pang, Wenxiao Wang, Francis EH Tay, Wei Liu, Yonghong Tian, and Li Yuan. Masked autoencoders for point cloud self-supervised learning. *arXiv preprint arXiv:2203.06604*, 2022. 3, 4, 5, 6

[31] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*, 2017. 7

[32] 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 *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 1

[33] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Learning video representations using contrastive bidirectional transformer. *arXiv preprint arXiv:1906.05743*, 2019. 4

[34] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7464–7473, 2019. 4

[35] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *arXiv preprint arXiv:2203.12602*, 2022. 4, 5, 7

[36] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. pages 10347–10357. PMLR, 2021. 1

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 1

[38] Weiyao Wang, Du Tran, and Matt Feiszli. What makes training multi-modal classification networks hard? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12695–12705, 2020. 4

[39] Wenhui Wang, Hangbo Bao, Li Dong, and Furu Wei. Vlmo: Unified vision-language pre-training with mixture-of-modality-experts. *arXiv preprint arXiv:2111.02358*, 2021. 4

[40] Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. Simvlm: Simple visual language model pretraining with weak supervision. *arXiv preprint arXiv:2108.10904*, 2021. 4

[41] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *ECCV*, pages 418–434, 2018. 5

[42] Chenfeng Xu, Shijia Yang, Tomer Galanti, Bichen Wu, Xiangyu Yue, Bohan Zhai, Wei Zhan, Peter Vajda, Kurt Keutzer, and Masayoshi Tomizuka. Image2point: 3d point-cloud understanding with 2d image pretrained models. In *European Conference on Computer Vision*, pages 638–656. Springer, 2022. 8

[43] Peng Xu, Xiatian Zhu, and David A Clifton. Multimodal learning with transformers: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 4

[44] Li Yi, Vladimir G Kim, Duygu Ceylan, I Shen, Mengyan Yan, Hao Su, ARCewu Lu, Qixing Huang, Alla Sheffer, Leonidas Guibas, et al. A scalable active framework for region annotation in 3d shape collections. *ACM TOG*, 35(6): 210, 2016. 5

[45] Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *CVPR*, 2022. 6, 7

[46] Yiyuan Zhang, Kaixiong Gong, Kaipeng Zhang, Hongsheng Li, Yu Qiao, Wanli Ouyang, and Xiangyu Yue. Meta-transformer: A unified framework for multimodal learning. *arXiv preprint arXiv:2307.10802*, 2023. 1, 4

[47] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *ICCV*, pages 16259–16268, 2021. 1

[48] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017. 5

[49] Jinxing Zhou, Jianyuan Wang, Jiayi Zhang, Weixuan Sun, Jing Zhang, Stan Birchfield, Dan Guo, Lingpeng Kong, Meng Wang, and Yiran Zhong. Audio–visual segmentation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVII*, pages 386–403. Springer, 2022. 3

[50] Jinguo Zhu, Xiaohan Ding, Yixiao Ge, Yuying Ge, Sijie Zhao, Hengshuang Zhao, Xiaohua Wang, and Ying Shan. Vl-gpt: A generative pre-trained transformer for vision and language understanding and generation. *arXiv preprint arXiv:2312.09251*, 2023. 4