Efficient Multimodal Fusion via Interactive Prompting

Yaowei Li [1] Ruijie Quan [2] Linchao Zhu [2] Yi Yang [2]

yaowei.li@uts.edu.au, {quanruijie, zhulinchao, yangyics}@zju.edu.cn

[1] ReLER, AAII, University of Technology Sydney
[2] CCAI, Zhejiang University

Abstract

Large-scale pre-training has brought unimodal fields such as computer vision and natural language processing to a new era. Following this trend, the size of multimodal learning models constantly increases, leading to an urgent need to reduce the massive computational cost of finetuning these models for downstream tasks. In this paper, we propose an efficient and flexible multimodal fusion method, namely PMF, tailored for fusing unimodally pretrained transformers. Specifically, we first present a modular multimodal fusion framework that exhibits high flexibility and facilitates mutual interactions among different modalities. In addition, we disentangle vanilla prompts into three types in order to learn different optimizing objectives for multimodal learning. It is also worth noting that we propose to add prompt vectors only on the deep layers of the unimodal transformers, thus significantly reducing the training memory usage. Experiment results show that our proposed method achieves comparable performance to several other multimodal finetuning methods with less than 3% trainable parameters and up to 66% saving of training memory usage.

1. Introduction

Recent years have witnessed the great success of large-scale pretrained language models [8,31,32] and visual models [6,10,23,39], leading to a surge of pretrained multimodal models [13, 14, 43, 47, 48] trying to align different modalities. Many prior methods utilize finetuning to update the entire set of model parameters for every target cross-modal task. Although finetuning can achieve good performance, it requires a large number of computational costs since the gradients and optimizer states for all parameters of multimodal models have to store. Therefore, it encourages researchers to propose more parameter-efficient methods than finetuning for multimodal learning.

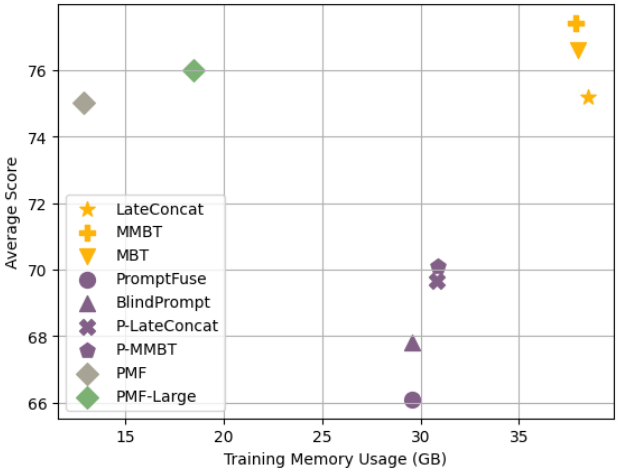

Figure 1. **Comparison over three multimodal classification tasks.** We compare our proposed PMF and PMF-Large with multiple finetuning (yellow) and prompt-based (purple) methods. The y-axis is the average score of three tasks, and the x-axis is the maximum GPU memory usage during training.

More recently, prompting tuning [17, 19, 21, 22, 29] is proposed to address this problem by freezing all parameters of a pretrained model while tuning only the continuous prompts. Specifically, it adds trainable continuous prompts to the original token sequences of input data. During training, only the continuous prompts are updated. For multimodal prompt-based learning, a most recent method [20] proposes to disentangle the functionality of the pretrained model which exhibits high flexibility. Although this method significantly reduces the tuned parameters (*e.g.*, less than 0.1% of the pretrained model), there still exists a large performance gap between it and the finetuning-based methods. In addition, this method adopts a sequential modular structure that the pretrained image transformer model is followed by a language transformer model, which causes two main problems in cross-modal learning: a one-way path learn-

ing and a significant increase in the number of model layers. Specifically, a one-way path learning in the multimodal model usually forces one modality to align with others, but not vice versa. In this way, cross-modal learning based on multiple different modalities is not fully explored due to the missing mutual alignments. Since the prompts are added to the token sequences of input data and are updated in the training, they require extensive gradient calculations in the backward propagation which cost numerous memory usages. As a result, this kind of method does not reduce the memory usage during training by much (up to 20%) though it reduces the number of parameters to update. In other words, this parameter-efficient method still requires massive computational resources which prevents it from being applied to many real-world applications.

To address these issues, we propose a **P**rompt-based **M**ultimodal **F**usion method with a high memory efficiency, namely **PMF**. Firstly, we present a new form of modular multimodal fusion framework which demonstrates high flexibility and facilitates a two-way interaction among different modalities. Specifically, we adopt a two-stream structure where the pretrained language model and image model construct the multimodal model in a parallel way. Therefore, tokens of different modalities can learn mutual interactions through a cross-attention-like operation. Such a parallel modular structure brings two benefits. First, unimodal pretraining can be directly utilized for multimodal learning through a parallel combination, eliminating the need for paired multimodal datasets that can be expensive to construct. Also, the type of image or language model can be changed easily (*e.g.*, replacing BERT with T5 for text generation tasks). Furthermore, incorporating extra modalities is made possible based on the parallel modular structure.

Moreover, we propose to leverage three types of interactive prompts (*i.e.*, query prompts, query context prompts, and fusion context prompts) in order to dynamically learn different objectives for multimodal learning. Intuitively, the query context prompt and query prompt can be seen as a pair of 'questions' and 'answers' with an aim of extracting necessary information for exchange between two modalities. After being translated by a non-linear mapping 'translator', the 'answer' is then delivered to the other modality for better cross-modal understanding. Finally, the fusion context prompts then provide the context to the delivered answer to facilitate the fusion.

Last but most importantly, PMF is a memory-efficient method that significantly reduces the memory requirements for the large pretrained model. Considering that calculating gradients for prompts for back-propagation is memory-consuming, we propose to add prompts only on the deep layers of the utilized unimodal transformers. Therefore, instead of passing through the entire multimodal model, the backward propagation only needs to pass through the deep few transformer layers to reach all trainable parameters, greatly reducing the training memory usage. We conduct extensive experiments to demonstrate the superior of it in our experiments. As a result, PMF enables large pretrained models to be trained on the GPU with a low memory requirement.

We conduct extensive experiments on three vision-language datasets: UPMC-Food101 [38], MM-IMDB [2], and SNLI-VE [41]. Through comparisons with multiple finetuning and prompt tuning methods (see in Fig. 1), we find that: (1) PMF is the most memory-efficient method for cross-modal learning so far, which reduces the training memory usage by up to 66% compared with finetuning baselines, and by 55% compared with prompt-based methods. (2) PMF can perform comparably compared to prior fine-tuning methods with much fewer trainable parameters (less than 2.5%) and memory usage.

Concretely, our contributions are as follows: (1) we present a new form of modular multimodal fusion framework which enables two-way interactions between different modalities and high flexibility of the entire model; (2) we disentangle vanilla prompts into three types of prompts, in order to dynamically learn different objectives for multimodal learning; (3) our proposed method is quite memory-efficient yet is able to achieve comparable performance with existing finetuning methods for multimodal fusion.

2. Related works

Multimodal Fusion. Multimodal fusion methods aim to simultaneously process the input of different modalities, such as audio-video [27], vision-language [2, 15], and inputs from different types of sensors [42], etc. In this paper, we specifically focus on the fusion of vision-language inputs, though our proposed strategy is compatible with other modality pairs as long as there are unimodally pretrained transformers for these modalities.

Our work is in line with deep learning-based multimodal fusion strategies [16, 18, 26, 28, 33, 34, 44]. In this line of work, [15] proposed a framework whose vision encoder solely serves as a mapping tool to encode the raw images to the token space of the text encoder. Such an architecture is widely used in the later multimodal fusion research [20, 26, 34]. Differently, [27] used a dual-encoder architecture with bottleneck fusion tokens to exchange information between two encoders for video-audio fusion. Our work has a similar architecture as in [27]. But our proposed method completely freezes the unimodal encoders and uses an interactive prompting technique for more efficient fusion.

Another important line of multimodal fusion work is through pretraining with the large-scale datasets, mostly achieved by self-supervised learning algorithms [1, 11, 24, 30, 36, 37, 45]. This line of works can be roughly divided into two kinds by their architectures. The first kind has

Task Type	Pretraining Type	Method
Unimodal	Unimodal	P-Tuning [22], VPT [12]
Unimodal	Multimodal	CoOp [47], MAPLE [14]
Multimodal	Multimodal	PI-VL [13], PTGM [43]
Multimodal	Unimodal	BlindPrompt [20], **PMF**

Table 1. **Prompt-based methods categorized by model pretraining types and downstream task types.** Our proposed PMF falls into the last category which utilize unimodally pretrained transformers for multimodal tasks.

a dual-encoder structure where image and text are treated separately, such as CLIP [30] and ALIGN [11]. The other kind simultaneously processes the vision and language inputs with cross-attention or self-attention over a longer sequence from two modalities. Our proposed method differs from these self-supervised architectures in that the individual components used in our model are unimodally pretrained with much less data. This difference directly results in a huge performance gap because of the lack of multimodal information in the pretraining stage. However, using unimodally pretrained models enables a much more flexible architecture. It has great potential in multimodal tasks where modality-paired large-scale pretraining data is not available, or when more advanced unimodal encoders are proposed in the future.

Prompt Tuning. As shown in Tab. 1, prompt-based methods can be roughly divided into four major categories in terms of the modalities of the pretrained model and the downstream tasks. Prompting techniques originally apply on unimodally pretrained transformers for unimodal natural language processing (NLP) tasks [17, 19, 21, 22, 29]. Pretrained GPT-3 can be simply leveraged with handcrafted prompts, which are some manually chosen words preceding the input text [4]. Then [22] and [17] proposed to change the handcrafted prompt to trainable continuous prompts and only update the prompt vectors during the training. Later on, [19] and [29] proposed to use prompt tuning in every hidden layer in the pretrained transformer instead of the input embeddings only. VPT [12] first applied prompt tuning to the vision transformer.

For methods that prompt multimodally pretrained models for unimodal tasks, many recent works apply prompt tuning to pretrained vision-language models (*i.e.* CLIP [30]) for unimodal vision tasks [3, 14, 25, 46, 47]. Another type of prompt-based method apply to the multimodal pretrained model for multimodal tasks [13, 43]. [13] adds prompts to an encoder-decoder one-for-all multimodal transformer, achieving comparable performance with fine-tuning with improved robustness against adversarial attacks. The method used in [13] is simple but effective, showing that prompting methods works well with powerful and complex multimodally pretraining models.

Our proposed fusion strategy is different from the above methods and falls into the last category where prompts are used to fuse pretrained unimodal models for multimodal tasks. Sharing the same architecture design with [15, 34], [20] uses prompt to align the feature extracted from raw images to the token space of pretrained language model. They achieved comparable performance to several multimodal fusion methods in low-resource settings but underperform fine-tuning baselines by a large margin with full data. Compared with them, our proposed PMF not only is more memory-efficient but can also perform comparably with fine-tuning baselines with full data.

3. Prompt-based Multimodal Fusion

In this section, we describe our proposed **P**rompt-based **M**ultimodal **F**usion strategy (**PMF**). We begin by summarising unimodal transformers developed for vision and language tasks in Sec. 3.1. Then we describe the base feature extraction process in Sec. 3.2. Lastly, we give a detailed description of how PMF integrates two unimodal transformer layers into a multimodal one via interactive prompting in Sec. 3.3.

3.1. Unimodal Transformers

Vision Transformer (ViT) [9] adapts the Transformer [35] architecture with minimum modifications. The RGB image input $\mathbf{x}_{img} \in \mathbb{R}^{h,w,c}$ is first cut into N_{img} non-overlapping patches and then linearly projected into a sequence of embeddings $\mathbf{z}$ with each $z_i \in \mathbb{R}^d$. Differently, the language Transformer first tokenizes raw text to N_{txt} one-hot word embeddings and then converted these discrete vectors into a sequence of N_{txt} continuous embeddings. The resulting continuous embedding for both Language Transformer and Vision Transformer share the same structure as follows:

$$\mathbf{z} = [\text{CLS}, z_1, z_2, ..., z_N] \tag{1}$$

where CLS is a special token prepended to the sequence so that its representation at the final layer can be used as the representation of the whole sequence for classification. Please note that the two unimodal transformers have different CLS tokens. The continuous embedding $\mathbf{z}$ is then fed into a transformer encoder which consists of L transformer layers. For each transformer layer, the input passes through modules including multi-head self-attention, layer normalization, multilayer perceptron, and finally added to the original input with a residual connection.

3.2. Unimodal Base feature Extraction

As shown in Fig. 2, the image and text inputs are first processed and fed into the unimodal transformer layers to

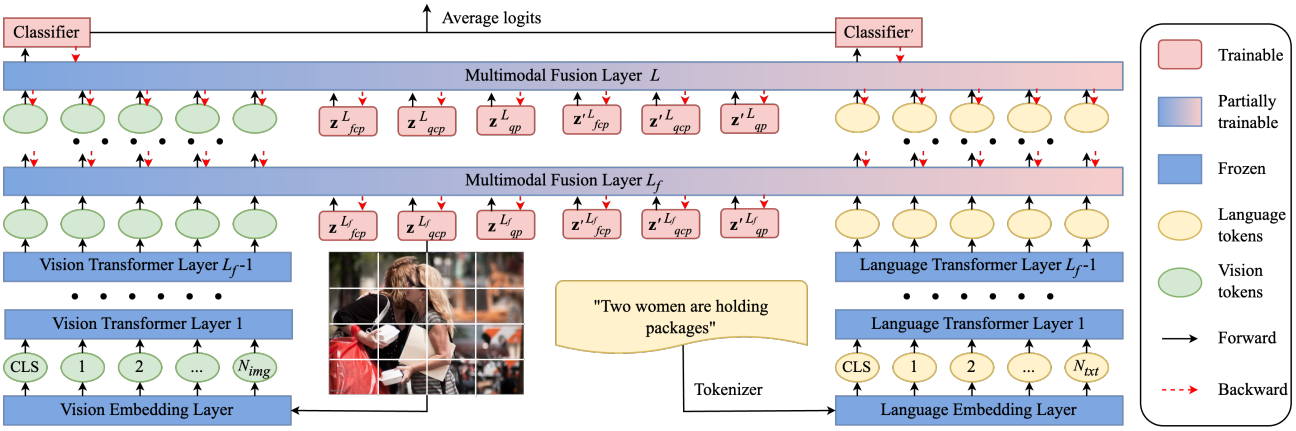

Figure 2. **Prompt-based multimodal fusion strategy (PMF) applied to vision-language inputs.** In the forward propagation, image and text inputs are first embedded into continuous token sequences and fed to the unimodal transformer layers for base feature extraction. The base features from both modalities then pass through multiple prompt-based multimodal fusion layers to get the feature of two CLS tokens for final classification. In the backward propagation, only multimodal fusion layers take part in the calculation of gradients, greatly saving memory usage during training. All pretrained parameters in both transformers are frozen during the training.

extract base features, respectively. At this stage, each encoder works exactly the same as they did in unimodal tasks. Here we denote the starting fusion layer as L_f. And the base feature extraction of each encoder can be denoted as:

$$\mathbf{z}^{l+1} = \text{TransLayer}^{l}(\mathbf{z}^{l}; \theta) \quad \text{if } l < L_f \tag{2}$$

where θ stands for the pretrained parameters. A smaller L_f leads to an earlier fusion and a larger L_f leads to a later fusion. A detailed discussion of the impact brought by different L_f can be found in Sec. 4.5.

3.3. Multimodal Fusion Layer

The extracted unimodal base features are then passed through multiple multimodal fusion layers, each consisting of a 'querying stage' and a 'fusion stage', as shown in Fig. 3 The querying stage focus on the extraction of what is necessary to pass, and the fusion stage focus on fusing the extracted information delivered from the other modality.

This two-stage setting makes the vanilla prompt training entangled with different learning objectives. Therefore, we propose to decouple the vanilla prompts into three kinds: 'query prompt' (QP, denoted $\mathbf{z}_{qp}$), 'query context prompt' (QCP, denoted $\mathbf{z}_{qcp}$), and 'fusion context prompt' (FCP, denoted $\mathbf{z}_{fcp}$) to dynamically learn different objectives for multimodal learning. According to the modality where prompts are used, each kind of prompt can be further specified as $\mathbf{z}_*$ and $\mathbf{z}'_*$ to distinguish from each other (*e.g.* $\mathbf{z}_{fcp}$ and $\mathbf{z}'_{fcp}$).

As shown in Fig. 3, QP and QCP are used in the querying stage and the FCP is used in the fusion stage. As suggested by their names, QP is to query information from the unimodal input sequence, QCP is to help this process by providing extra context to the query. QP and QCP like a pair of 'questions' and 'answers', translated by the non-linear mapping. As for FCP, it is responsible for providing the context to the fusion in the fusion stage. We now introduce how these three kinds of prompts interact with each other in the two stages of each multimodal fusion layer.

Querying Stage. We first concatenate corresponding QP and QCP to the input sequence $\mathbf{z}$. The resulting input sequence after the concatenation is:

$$[\mathbf{z}^{l}||\mathbf{z}^{l}_{qcp}||\mathbf{z}^{l}_{qp}] \tag{3}$$

where '||' denotes the concatenation operation. Then we feed the concatenated sequence to the unimodal transformer layer, which can be denoted as:

$$[\hat{\mathbf{z}}^{l}||\hat{\mathbf{z}}^{l}_{qcp}||\hat{\mathbf{z}}^{l}_{qp}] = \text{TransLayer}^{l}([\mathbf{z}^{l}||\mathbf{z}^{l}_{qcp}||\mathbf{z}^{l}_{qp}]; \theta) \tag{4}$$

After the forward propagation, the output of QP $\hat{\mathbf{z}}^{l}_{qp}$ is extracted as the queried information to be used in the following fusion operations. It should be noted that though $\hat{\mathbf{z}}^{l}_{qcp}$ will not be used in the following fusion operations, it has played an important role by providing the context of query in the querying stage.

The queried fusion intermediate $\hat{\mathbf{z}}^{l}_{qp}$ is then mapped to the representation space of the other modality through a non-linear mapping function:

$$\mathbf{y}^{l}_{qp} = f^{l}(\hat{\mathbf{z}}^{l}_{qp}) \tag{5}$$

where f is a non-linear mapping function. Specifically, a mapping function consists of two linear layers with a bottleneck structure to reduce dimension and only the first linear layer has a ReLu function. Each fusion layer l has two

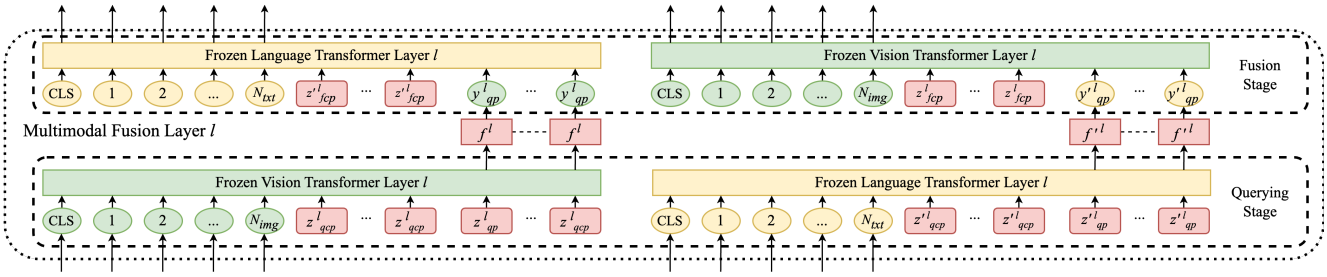

Figure 3. **Prompt-based multimodal fusion layer.** We propose to use three kinds of interactive prompts to achieve the fusion/exchange of information from two modalities. We use 'query prompt' (z_{qp}, z'_{qp}) and 'query context prompt' (z_{qcp}, z'_{qcp}) in the querying stage to extract what is necessary, after a non-linear mapping, the extracted information is then fused to the other modality with the help of 'fusion context prompt'(z_{fcp}, z'_{fcp}) in the fusion stage. Yellow and green elements stand for language and vision modalities, respectively. Red boxes indicate trainable modules.

mapping functions f^l and f'^l, building a two-way interaction among different modalities. Note that the non-linear mapping functions contain more than 95% of the trainable parameters in PMF.

Fusion Stage. We first concatenate the mapped fusion intermediates $\mathbf{y}^l_{qp}$ to the original input sequence $\mathbf{z}'^l$ and FCP $\mathbf{z}'^l_{fcp}$ of the other modality. Then we feed the concatenated sequence to the unimodal transformer layer of the other modality to complete a one-way fusion. These two processes can be together denoted as:

$$[\mathbf{z}'^{l+1}||\hat{\mathbf{z}}'^l_{fcp}||\hat{\mathbf{y}}^l_{qp}] = \text{TransLayer}^l([\mathbf{z}'^l||\mathbf{z}'^l_{fcp}||\mathbf{y}^l_{qp}]; \theta') \tag{6}$$

where $[*]'$ means $[*]$ from the other modality.

Finally, the output of two unimodal transformer layers $\mathbf{z}^{l+1}$ and $\mathbf{z}'^{l+1}$ are together as the output of the multimodal fusion layer and fed to the higher layers. The entire multimodal fusion process can be concluded as:

$$[\mathbf{z}^{l+1}||\mathbf{z}'^{l+1}] = \text{FusionLayer}^l([\mathbf{z}^l||\mathbf{z}'^l]; \theta, \theta') \quad \text{if } L_f \leq l \tag{7}$$

After the multimodal fusion is complete, we take the output representation of CLS token z^L_{CLS} and z'^L_{CLS} to two different linear classifiers and average the pre-softmax logits for classification. Mathematically, such a classifier setting is equivalent to feeding the classifier with the concatenated features when a linear classifier is used, except for a different scale of gradient caused by the averaging operation.

4. Experiments

In this section, we analyze the performance of our proposed PMF on three multimodal datasets and aim to answer the following questions: (1) How efficient is the PMF and does PMF perform well (Sec. 4.4)? (2) What factors affect the effectiveness of PMF (Sec. 4.5)? In addition, we explore the PMF equipped with larger transformers and its impact brought to the performance and memory efficiency in Sec. 4.6. Lastly, we introduce a Neural Architecture Search (NAS) method to automatically search for the preferable fusion structure for PMF in Sec. 4.7.

4.1. Datasets and Metrics

UPMC Food-101 [38] is a multimodal classification dataset, which contains food images with textual recipe descriptions for 101 kinds of food. UPMC Food-101 contains a total of 90,840 image-text pairs with a size range between 790 and 956 pairs for different classes. As the dataset only has training and testing sets, we follow [15] and create a validation split of 5000 samples from the training set.

MM-IMDB [2] is a multimodal multi-label classification dataset, which contains movie plot outlines and movie posters. The task is to predict the genre of movies. This dataset contains 25,956 image-text pairs for 23 classes with a long-tail distribution.

SNLI-VE [41] is a multimodal classification dataset for the visual entailment tasks, which is to reason about the relationship between an image premise and a text hypothesis into entailment, contradiction, or neutrality. SNLI-VE contains a total of 565,286 image-text pairs. Please note that we only use image premise and text hypothesis in the input, which is different from the settings in some other papers where text premises are also used in the inputs [43].

We report accuracy for UPMC Food-101 and SNLI-VE, and Macro/Micro-F1 scores for MM-IMDB as metrics.

4.2. Existing Methods and Baselines

We report the performance of several baselines and existing methods. First, we report the performance of finetuning unimodal models (*i.e.* **BERT** [8], **ViT** [9]) to verify the effectiveness of multimodal fusion. Specifically, we take the output representation of CLS token of the last layer in ViT and BERT, and feed it into a linear classifier. We also report the performance of **VPT** [12] and a prompt-based BERT (denoted **P-BERT**) for a better comparison. For VPT

Method	Updated Param. (Million)	Memory Usage (GB) Train/Inference	SNLI-VE	Food-101	MM-IMDB	Avg.
Linear	-	3.76 / 3.23	50.05	78.96	49.76 / 56.83	60.77
ViT	86.5	9.36 / 1.99	33.33	74.69	38.39 / 49.88	50.72
BERT	109.0	30.82 / 2.79	69.82	87.44	58.91 / 64.31	72.96
LateConcat	196.0	38.54 / 3.36	70.01	93.29	59.56 / 64.92	75.18
MMBT*	196.5	37.87 / 3.48	74.69	94.10	60.80 / 66.10	77.41
MBT*	196.0	38.00 / 4.06	74.02	93.56	59.60 / 64.81	76.60
VPT	-	6.12 / 2.01	33.33	72.55	35.22 / 44.49	48.58
P-BERT	-	28.13 / 2.99	63.28	81.07	48.67 / 54.58	65.33
PromptFuse	-	29.57 / 3.55	64.53	82.21	48.59 / 54.49	66.09
BlindPrompt	-	29.57 / 3.65	65.54	84.56	50.18 / 56.46	67.81
P-LateConcat	0.3	30.82 / 3.43	63.05	89.03	53.91 / 59.93	69.67
P-MMBT	0.9	30.90 / 3.48	67.58	86.58	52.95 / 59.30	70.10
PMF (M=4, L_f=10)	2.5	12.84 / 4.08	71.92	91.51	58.77 / 64.51	75.02
PMF-large (M=4, L_f=22)	4.5	18.44 / 6.42	72.10	91.68	61.66 / 66.72	75.99

Table 2. **Multimodal classification performance.** PMF achieve comparable performance to the finetuning baselines with less than 3% of trainable parameters and up to 66% of training memory usage. MM-IMDB is F1-Macro / F1-Micro, others are accuracy. We report the maximum memory usage in training and evaluating UPMC Food-101 for each method. We report mean performance over 3 runs with different random seeds. '-' means trainable parameter less than 0.1 M. PMF-Large uses bert-large and vit-large models (24 hidden layers) while others use bert-base and vit-base models (12 hidden layers). M is the prompt length and L_f is the starting fusion layer.

and P-BERT, the input sequence to each transformer layer is concatenated with a prompt vector, whose length is set to 10. And the concatenated prompt vectors and the final linear classifier are the only updated modules in training.

In addition, we compare against a strong baseline method which concatenates the output features of CLS tokens of ViT and BERT, and feed the concatenated feature to a linear classifier, denoted as **LateConcat**. In this case, the input to the classifier is (768 + 768)-dimensional. Besides, we also introduce **Linear**, which shares the same architecture with LateConcat with the only difference in the updated modules. Linear only updates the linear classifier while LateConcat updates all parameters during training.

We reimplement MMBT (denoted **MMBT***) [15] and MBT (denoted **MBT***) [27] with a `vit-base` model as the vision encoder and a `bert-base` model as the text encoder for fair and controlled comparison. We set the fusion layer $L_f = 8$ and use 4 fusion tokens in MBT as recommended in the original paper.

We also propose a prompt-based MMBT and a prompt-based LateConcat, denoted as **P-MMBT** and **P-LateConcat**, respectively. In P-MMBT and P-LateConcat, we apply deep prompt tuning on both vision and language encoders, which are pretrained backbones with frozen parameters during training. We set the prompt length in each layer of two encoders to 10, totalling 240 prompt vectors. Similar to VPT and P-BERT, P-LateConcat only updates the final linear classifier and prompt vectors during training. Compared with P-LateConcat, P-MMBT have an extra linear projection layer and a smaller linear classifier to train.

Lastly, we report the performance of **PromptFuse** and **BlindPrompt** [20], both proposed in the only existing paper which leverages unimodally pretrained models for multimodal fusion through prompting. We set the prompt length to 20 as recommended in the original paper.

4.3. Implementation Details

Pretrained Backbone and Initialization. Unless otherwise noted, we use an ImageNet-21k [7] pretrained `vit-base` model for the vision encoder and a `bert-base-uncased` model for the language encoder in all experiments. All pretrained checkpoints are from huggingface [40]. All prompt vectors are initialized through a Gaussian distribution (mean=0, std=0.02).

Network Training. We use SGD optimizer in all experiments with momentum set to 0.9 and weight decay set to $1e^{-4}$. The batch size is set to 64 for SNLI-VE, and 32 for UPMC Food-101 and MM-IMDB. Cross entropy loss is applied in all experiments and the class labels are weighted by their inverse frequency for UPMC Food-101 and MM-IMDB. More details are in the supplementary material.

4.4. Main Results

PMF is most memory-efficient. As shown in Tab. 2, PMF is the most memory-efficient multimodal fusion model of all existing prompt-based methods and baselines. PMF can save up to 66% of training memory usage compared with finetuning baselines. Even compared with the existing most

QP	Mapping f	QCP	FCP	MM-IMDB
				48.80/56.38
			✓	45.38/53.34
✓				43.56/51.43
✓	✓			54.92/62.25
✓✓	✓			57.98/63.69
✓	✓	✓		58.30/64.07
✓	✓		✓	58.34/64.15
✓	✓	✓	✓	58.63/64.23

Table 3. **PMF Component Ablation.** There are four types of trainable modules in our proposed PMF. We set the fusion layer $L_f = 10$ and add different components one at a time to see their individual impact. All prompts with ✓have a length of 4, and prompts with ✓✓have a length of 8.

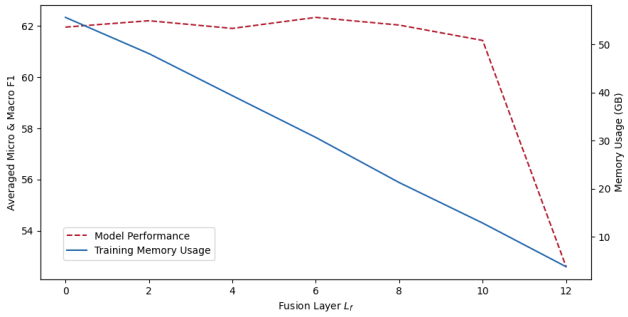

Figure 4. **Model Performance and Training Memory Usage under different fusion layers L_f.** We set the prompt length $M = 4$ for every prompt vector with different fusion layer $L_f = 0, 2, 4, 6, 8, 10, 12$.

memory-efficient prompt-based multimodal method, PMF still saves an extra of more than 50% of training memory.

PMF outperforms all existing prompt-based methods. With the same pretrained unimodal transformers, prompt-based methods underperform the full finetuning methods by a large margin. Some prompt-based methods even underperform unimodal finetuning baselines. Our proposed PMF achieves the best performance among all prompt-based methods. Especially, PMF outperforms all unimodal baselines in all experiments, showing that the two modalities are successfully fused.

PMF is competitive with finetuning baselines. Tab. 2 shows that PMF achieves comparable performance with full finetuning baselines with less than 3% trainable parameters while saving 66% of memory cost, significantly narrowing the gap between finetuning and prompt-based methods. Furthermore, PMF even outperforms the finetuning LateConcat when equipped with larger transformers (*i.e.* `bert-large` and `vit-large`).

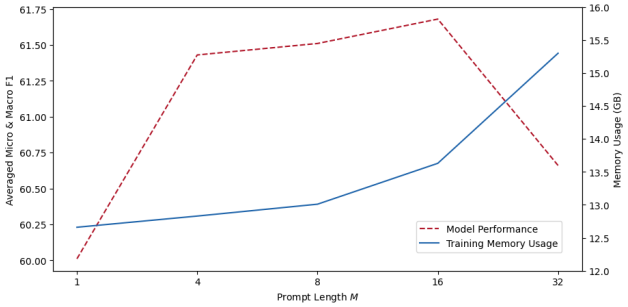

Figure 5. **Model Performance and Training Memory Usage under different prompt length M.** We fix the fusion layer $L_f = 10$ and study 1, 4, 8, 16, and 32 tokens for each prompt vector.

4.5. Ablation Study

In this section, we investigate the impact of different factors in our proposed fusion strategy. All experiments in this section are conducted on the MM-IMDB dataset.

Components Ablation. We verify the effectiveness of three kinds of prompts and the non-linear mapping function in this section. The results are shown in Tab. 3. The first row without any components in PMF is equivalent to the Linear model introduced in Sec. 4.2. The comparison between the first three rows shows that merely prompting the top layers of the two transformers not only cannot achieve multimodal fusion. Oppositely, it will disturb the feature space of two transformers, which finally hurts the performance. Though the mapping functions give the biggest boost to the performance, it should be noted that the mapping functions f cannot work without QP querying the fusion intermediates.

In addition, the comparison between the last four rows shows that decoupling the prompts into three individual modules with different learning objectives brings a performance gain. More specifically, the comparison between the fifth and sixth row in Tab. 3 shows that an extended QP cannot replace QCP. Since only the output of QP tokens are fused to the other modality while that of QCP discarded, replacing QCP with a longer QP not only increases the computation as the sequences in the fusion stage are longer but also results in a damaged performance As a result, every module introduced in PMF contributes to the quality of multimodal fusion. The missing of any of the four modules will bring a performance drop at different scales.

Fusion Layer. We now investigate impacts brought by different fusion layers L_f to the fusion performance and memory efficiency. The results are summarized in Fig. 4. As can be seen in the figure, the training memory usage keeps decreasing as the fusion starts later. However, the performance of the fusion model is relatively consistent with $L_f \leq 10$. Therefore, adding prompts only on the deep layers ($10 < l < L$) is empirically better for the trade-off between performance and memory efficiency.

Text Encoder	Image Encoder	Memory Usage Train/Inference	MM-IMDB
bert-base	vit-base	12.84 / 4.08	58.77 / 64.51
bert-base	vit-large	14.16 / 4.89	59.70 / 65.20
bert-large	vit-base	17.17 / 5.53	60.08 / 65.41
bert-large	vit-large	18.44 / 6.42	61.66 / 66.72

Table 4. **Comparison of PMF applying to different unimodal transformers.** We set the fusion layer $L_f = L - 2$ and prompt length $M = 4$ in all experiments. MM-IMDB is reported with F1-Macro / F1-Micro.

Training Memory	SNLI-VE	Food-101	MM-IMDB	Avg.
33.36 GB	72.27	92.1	59.67 / 65.57	75.66

Table 5. **Performance of PMF applied with NAS.** MM-IMDB is F1-Macro / F1-Micro, others are accuracy. We only report the training memory usage.

Prompt Length. The ablation study on prompt length is carried out with three kinds of prompts set to have the same length (*i.e.* $M_{qp} = M_{qcp} = M_{fcp}$). Fig. 5 summarizes the results. The performance increases as the prompt length grow longer when $M \leq 16$, and drops when the prompts are too long ($M = 32$). It should be stressed that the training memory usage only increases around 1 GB as the prompt length grows from 1 to 16, which means the fusion layer L_f is the major factor of the training memory usage instead of prompt length.

4.6. Modularity and Flexibility

PMF is highly modular, which means it is trivial to replace the unimodal transformers when there are better ones. In this section, we first describe how to replace the unimodal transformer and then show the benefits of such flexibility through experiments with larger pretrained models.

Since the total transformer layers, L_{img} and L_{txt} of each unimodal transformer are now different, the unimodal base features of two modalities now take different layers to extract, and the number of remaining layers for fusion stays the same. Simply, the fusion layers L_f of two transformers can be further specified as $L_{f-img} = L_{img} - 2$ and $L_{f-txt} = L_{txt} - 2$. In addition, the difference between different hidden dimensions d is automatically handled by the non-linear mapping functions f.

The results shown in Tab. 4 clearly demonstrate that PMF can be empowered by larger unimodal transformers with a very limited increase of training memory usage.

4.7. PMF with NAS

The hyper-parameters introduced in the proposed PMF are fusion layer L_f and prompt length M. Although PMF works well without exhausting hyper-parameter tuning, it is still preferable to have specific settings for every different task and data distribution. In this section, we experiment with automatic fusion structure search via AutoFormer [5]. A detailed description of the search space and evolution search can be found in the supplementary material.

Tab. 5 shows the performance of NAS-applied PMF on three datasets. With an increase in training memory usage, PMF-NAS achieves better results than regular PMF with the same vision and language encoders, greatly reducing the workload of finding the preferable fusion structure.

5. Limitations and Future Works

The first limitation is that PMF's performance on three datasets is still behind finetuning baselines with the same pretrained backbones, indicating more work developing prompt-based methods to fully leverage the knowledge inside the pretrained models in the future, finally achieving equivalent or surpassing results through prompting.

The second limitation is about the hyper-parameters tuning: It is preferable to decouple prompts into three kinds by their roles in multimodal fusion. However, it also brings more work to hyper-parameter tuning if someone is expecting the best results via an optimal fusion structure.

Our future research endeavours will involve further investigation of the PMF in diverse multimodal understanding tasks such as Visual Question Answering, utilizing various model architectures.

6. Conclusion

We propose a new form of modular multimodal fusion framework which demonstrates high flexibility and facilitates a two-way interaction among different modalities, namely PMF. PMF leverages three types of interactive prompts in order to dynamically learn different objectives for multimodal learning. By adding the prompts only on the deep layers of utilized unimodal transformers, PMF can significantly reduce the memory usage of the gradient calculation in the backward propagation. Through extensive experiments, we demonstrate that PMF is quite memory-efficient and yet able to perform comparably with existing finetuning baselines.

7. Acknowledgement

This work was supported in part by the Australian Research Council (ARC) under Grant DP200100938.

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. 2

[2] John Arevalo, Thamar Solorio, Manuel Montes-y Gómez, and Fabio A González. Gated multimodal units for information fusion. *arXiv preprint arXiv:1702.01992*, 2017. 2, 5

[3] Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. Visual prompting: Modifying pixel space to adapt pre-trained models. *arXiv preprint arXiv:2203.17274*, 2022. 3

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 3

[5] Minghao Chen, Houwen Peng, Jianlong Fu, and Haibin Ling. Autoformer: Searching transformers for visual recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12270–12280, 2021. 8

[6] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15750–15758, 2021. 1

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 6

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 1, 5

[9] 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. *arXiv preprint arXiv:2010.11929*, 2020. 3, 5

[10] 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. 1

[11] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pages 4904–4916. PMLR, 2021. 2, 3

[12] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. *arXiv preprint arXiv:2203.12119*, 2022. 3, 5

[13] Chen Ju, Tengda Han, Kunhao Zheng, Ya Zhang, and Weidi Xie. Prompting visual-language models for efficient video understanding. *arXiv preprint arXiv:2112.04478*, 2021. 1, 3

[14] Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. *arXiv preprint arXiv:2210.03117*, 2022. 1, 3

[15] Douwe Kiela, Suvrat Bhooshan, Hamed Firooz, Ethan Perez, and Davide Testuggine. Supervised multimodal bitransformers for classifying images and text. *arXiv preprint arXiv:1909.02950*, 2019. 2, 3, 5, 6

[16] Ryan Kiros, Ruslan Salakhutdinov, and Rich Zemel. Multimodal neural language models. In *International conference on machine learning*, pages 595–603. PMLR, 2014. 2

[17] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021. 1, 3

[18] Guangrui Li, Guoliang Kang, Yi Zhu, Yunchao Wei, and Yi Yang. Domain consensus clustering for universal domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[19] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021. 1, 3

[20] Sheng Liang, Mengjie Zhao, and Hinrich Schütze. Modular and parameter-efficient multimodal fusion with prompting. *arXiv preprint arXiv:2203.08055*, 2022. 1, 2, 3, 6

[21] Xiao Liu, Kaixuan Ji, Yicheng Fu, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. *arXiv preprint arXiv:2110.07602*, 2021. 1, 3

[22] Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. Gpt understands, too. *arXiv preprint arXiv:2103.10385*, 2021. 1, 3

[23] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 1

[24] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32, 2019. 2

[25] Yuning Lu, Jianzhuang Liu, Yonggang Zhang, Yajing Liu, and Xinmei Tian. Prompt distribution learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5206–5215, 2022. 3

[26] Oscar Mañas, Pau Rodriguez, Saba Ahmadi, Aida Nematzadeh, Yash Goyal, and Aishwarya Agrawal. Mapl: Parameter-efficient adaptation of unimodal pre-trained models for vision-language few-shot prompting. *arXiv preprint arXiv:2210.07179*, 2022. 2

[27] Arsha Nagrani, Shan Yang, Anurag Arnab, Aren Jansen, Cordelia Schmid, and Chen Sun. Attention bottlenecks for multimodal fusion. *Advances in Neural Information Processing Systems*, 34:14200–14213, 2021. 2, 6

[28] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y Ng. Multimodal deep learning. In *ICML*, 2011. 2

[29] Guanghui Qin and Jason Eisner. Learning how to ask: Querying lms with mixtures of soft prompts. *arXiv preprint arXiv:2104.06599*, 2021. 1, 3

[30] 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. 2, 3

[31] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018. 1

[32] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67, 2020. 1

[33] Nitish Srivastava and Russ R Salakhutdinov. Multimodal learning with deep boltzmann machines. *Advances in neural information processing systems*, 25, 2012. 2

[34] Maria Tsimpoukelli, Jacob L Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. *Advances in Neural Information Processing Systems*, 34:200–212, 2021. 2, 3

[35] 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. 3

[36] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International Conference on Machine Learning*, pages 23318–23340. PMLR, 2022. 2

[37] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14733–14743, 2022. 2

[38] Xin Wang, Devinder Kumar, Nicolas Thome, Matthieu Cord, and Frederic Precioso. Recipe recognition with large multimodal food dataset. In *2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pages 1–6. IEEE, 2015. 2, 5

[39] Zejia Weng, Xitong Yang, Ang Li, Zuxuan Wu, and Yu-Gang Jiang. Semi-supervised vision transformers. In *ECCV*, 2022. 1

[40] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45, 2020. 6

[41] Ning Xie, Farley Lai, Derek Doran, and Asim Kadav. Visual entailment: A novel task for fine-grained image understanding. *arXiv preprint arXiv:1901.06706*, 2019. 2, 5

[42] Ning Xiong and Per Svensson. Multi-sensor management for information fusion: issues and approaches. *Information fusion*, 3(2):163–186, 2002. 2

[43] Hao Yang, Junyang Lin, An Yang, Peng Wang, Chang Zhou, and Hongxia Yang. Prompt tuning for generative multimodal pretrained models. *arXiv preprint arXiv:2208.02532*, 2022. 1, 3, 5

[44] Yi Yang, Yueting Zhuang, and Yunhe Pan. Multiple knowledge representation for big data artificial intelligence: framework, applications, and case studies. *Frontiers of Information Technology & Electronic Engineering*, 22(12):1551–1558, 2021. 2

[45] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022. 2

[46] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16816–16825, 2022. 3

[47] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022. 1, 3

[48] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8746–8755, 2020. 1