Masked Modeling for Self-supervised Representation Learning on Vision and Beyond

Siyuan Li*, Luyuan Zhang*, Zedong Wang, Di Wu, Lirong Wu, Zicheng Liu, Jun Xia, Cheng Tan, Yang Liu, Baigui Sun, Stan Z. Li†, *IEEE Fellow*

Abstract—As the deep learning revolution marches on, self-supervised learning has garnered increasing attention in recent years thanks to its remarkable representation learning ability and the low dependence on labeled data. Among these varied self-supervised techniques, masked modeling has emerged as a distinctive approach that involves predicting parts of the original data that are proportionally masked during training. This paradigm enables deep models to learn robust representations and has demonstrated exceptional performance in the context of computer vision, natural language processing, and other modalities. In this survey, we present a comprehensive review of the masked modeling framework and its methodology. We elaborate on the details of techniques within masked modeling, including diverse masking strategies, recovering targets, network architectures, and more. Then, we systematically investigate its wide-ranging applications across domains. Furthermore, we also explore the commonalities and differences between masked modeling methods in different fields. Toward the end of this paper, we conclude by discussing the limitations of current techniques and point out several potential avenues for advancing masked modeling research. A paper list project with this survey is available at https://github.com/Lupin1998/Awesome-MIM.

Index Terms—Self-supervised Learning, Masked Modeling, Generative Model, Natural Language Processing, Audio and Speech, Graph

1 Introduction

Deep learning has made tremendous progress over the past decade, with an early emphasis on the supervised learning approaches [1], [2], [3], [4] that depend on labeled data. However, self-supervised learning (SSL) and pretraining techniques [5] have burgeoned, captivating the deep learning community with their advanced transferability and reduced dependence on labels. Fundamentally, *SSL is to learn valuable representations from unlabeled data, e.g., intrinsic data structures, with designated pretext tasks*. The development of SSL and pretraining techniques has been rapid, with a proliferation of variants across modalities and fields. To date, their evolutions have followed far different trajectories depending on specific modality and domain. Thus, it is crucial to provide an up-to-date survey of the rapidly growing masked modeling. The development timeline of SSL is schematically illustrated in Figure 1.

Early Attempts. Due to the underwhelming results from discriminative pretext tasks, early-stage SSL methods were dominated by generative objectives. Research at that time focused heavily on generative modeling itself, such as image and text generation tasks, with pretraining treated as a byproduct rather than the major concern. Even today, generative approaches remain at the heart of self-supervision, including Autoencoder-based models [6], [7], GAN-based models [8], and diffusion-based models [9]. In contrast, former discriminative SSL frameworks were hinged on ad-hoc pretext tasks. Methods like [10] and [11] introduced other tasks like colorization and shuffle-reconstruction. [12] pioneered the use of masked inputs for reconstruction, which served as a precursor to today's masked modeling. However, these approaches have not yet hit the mainstream.

Language Domain. In 2018, BERT [13] and GPT [14] introduced Masked Language Modeling (MLM) and Next Sentence Prediction (NSP) for natural language processing (NLP), ushering in more standardized objectives. Because of the remarkable performance of BERT and GPT, generative pretraining methods based on MLM and NSP have become the mainstream approaches for NLP. From 2018 to 2020, the NLP community mainly focused on refining pretraining strategies based on MLM and NSP. After contrastive learning was theoretically formalized, some 2021 works [15] explored contrastive discriminative pretraining for NLP. However, MLM-based research remains in a dominant position.

Vision Domain. In contrast to NLP, self-supervised pretraining in computer vision (CV) has followed a more complex and diverse development. In 2018, theoretical advances in contrastive learning like [16] and [17] established their foundations, enabling significant performance gains in linear evaluation protocols. This catalyzed the rise of discriminative models for SSL in computer vision. From 2019 to 2021, CV research was dominated by contrastive approaches, with influential frameworks like [18], [19], and [20] achieving impressive results. During this period, some generative models like iGPT [21] adopted auto-regressive pretraining with a GPT-2 [14] backbone. However, due to performance limitations, generative self-supervision had minimal impact compared to contrastive learning. This

- *Siyuan Li and Luyuan Zhang are co-first authors. Stan Z. Li is the corresponding author.*
- *Siyuan Li, Luyuan Zhang, Zedong Wang, Di Wu, Lirong Wu, Zicheng Liu, Jun Xia, Cheng Tan, and Stan. Z. Li are from the AI Lab, Research Center for Industries of the Future, Westlake University, Hangzhou, Zhejiang, China, 310030.*
  E-mail: lisiyuan@westlake.edu.cn; zhangluyuan@smail.nju.edu.cn; wangzedong@westlake.edu.cn; wudi@westlake.edu.cn; wulirong@westlake.edu.cn; liuzicheng@westlake.edu.cn; junxia@westlake.edu.cn; tancheng@westlake.edu.cn; stan.zq.li@westlake.edu.cn.
- *Siyuan Li, Yang Liu, and Baigui Sun are with the DAMO Academy, Hangzhou, Zhejiang, China.*
  Email: ly261666@alibaba-inc.com; baigui.sbg@alibaba-inc.com.

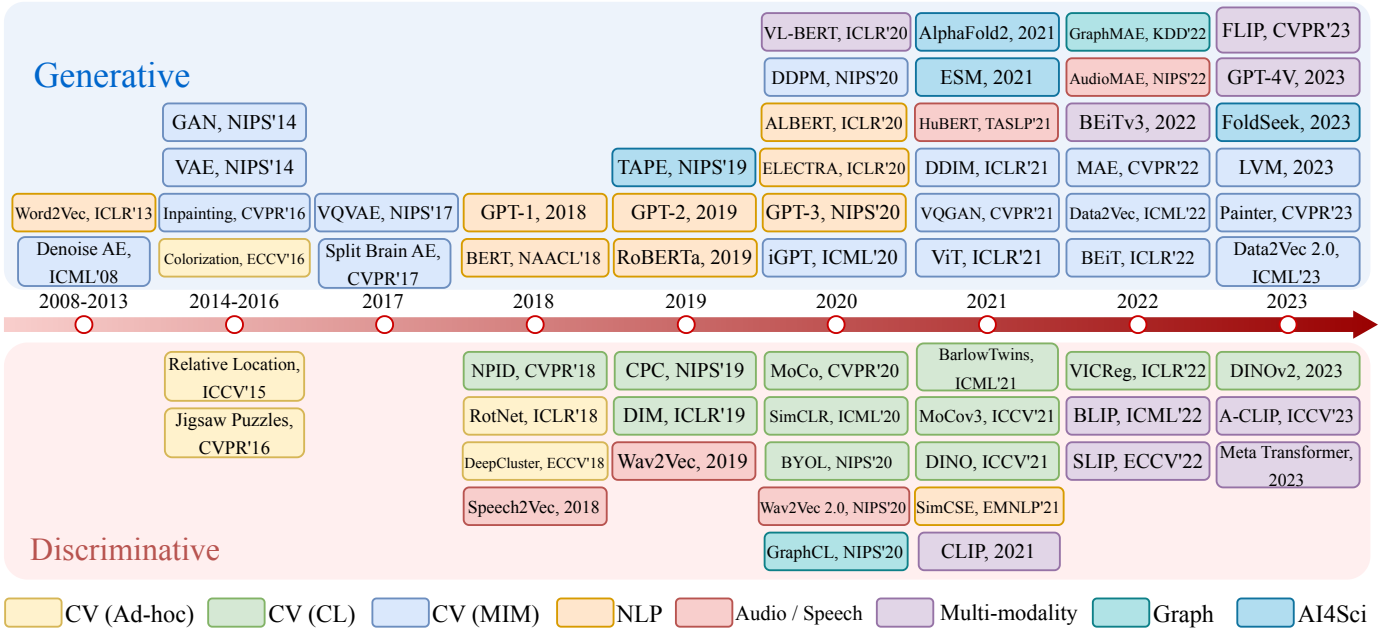


Fig. 1: Research in self-supervised learning (SSL) can be broadly categorized into **Generative** and **Discriminative** paradigms. We reviewed major SSL research since 2008 and found that SSL has followed distinct developmental trajectories and stages across time periods and modalities. Since 2018, SSL in NLP has been dominated by generative masked language modeling, which remains mainstream. In computer vision, discriminative contrastive learning dominated from 2018 to 2021 before masked image modeling gained prominence after 2022.

changed in 2021 when Vision Transformers [22] (ViT) altered the CV self-supervision landscape. Post-ViT [22], CV research began emulating BERT [13] by tokenizing images and then pretraining transformers. MAE [23] formally introduced masked image modeling, achieving strong performance. Since then, CV self-supervision has focused on generative reconstruction and masked modeling.

Multimodality. The earliest multimodal pre-trained models emerged in 2020, with VL-BERT [24] fusing modalities using a transformer architecture. In 2021, CLIP [25] combined computer vision and NLP modalities, ushering in an era of contrastive learning for multimodal pretraining that became mainstream in academia. Proposed in 2022, BEiT.v3 [26] introduced masked modeling as a pretraining technique for multimodal models, while MetaTransformer [27] combined multiple approaches. Since then, masked modeling has played a pivotal role in multimodal research.

Other Domains. SSL has been broadly applied across modalities beyond NLP and CV, including Audio, Speech, Biology, Video, and others. Research on SSL pretraining for **Audio and Speech** has closely followed the paradigms in CV and NLP. When contrastive learning gained popularity in 2018, influential speech models like [28] and [29] adopted contrastive learning for pretraining. Notably, [29] combined masked modeling as a data augmentation technique for contrastive learning. In 2021, [30] and then [31] in 2022 drew inspiration from masked image modeling in CV to implement masked spectrum modeling for audio. Since then, Masked Modeling has been a main direction in audio and speech research. As AlphaFold [32] achieved a great breakthrough in accurate protein structure predictions in 2021s, masked modeling has been introduced into **Biology** and **Chemistry** to assist the scientists as the AI-for-Science (**AI4Sci**) research paradigm.

Masked modeling has demonstrated compelling performance across modalities, including vision, language, speech, and beyond. With its widespread adoption, the landscape of masked modeling research has grown increasingly diverse. A multitude of masked modeling methods have emerged, creating a complex ecosystem of models tailored to different data types and tasks. Therefore, it is highly worthwhile to systematically review recent advances and provide structured categorization of the extensive masked modeling literature. In this paper, we conduct an extensive survey of the masked modeling research landscape. We thoroughly investigate the latest innovations in self-supervised representation learning across vision, NLP, speech, and other domains. Our main contribution is a comprehensive taxonomy that organizes the extensive body of masked modeling techniques into coherent groups according to training objectives, model architectures, and applications. This framing elucidates the relationships between existing methods and paves the way for developing new masked modeling techniques. Our review and classification provide a holistic reference to inform and accelerate future masked modeling research across modalities.

To sum up, our contributions include:

1) We provide a timely literature review and a comprehensive framework, taking computer vision as an instance, to holistically conceptualize masked modeling principles that can categorize different applications to date across domains and modalities under a common lens.
2) We meticulously review and discuss the technical details within the masked modeling framework, such as masking strategies, targets, networks, and more, to let researchers get a better grasp of the involved techniques and thus gain a deeper understanding and insights.
3) We systematically survey the downstream applications of masked modeling in vision, presenting the technical challenges and further showcasing their widespread

applicability to other modalities and domains beyond vision, such as audio, speech, graph, biology, and more.
4) Through extensive algorithmic research and detailed evaluations, we provide a collection of comprehensive tables and awesome lists of masked modeling methods on GitHub. In the end, we identified the future directions of masked modeling research and further provided heuristic suggestions and reflections on these directions.

2 PRELIMINARY

2.1 Notations

The notations used in this survey are illustrated in Table 1, and we will present a detailed demonstration of the changes and corresponding relationships between the symbols and variables in the table.

In this paper, x denotes a data sequence which can be a sentence in NLP, a patch sequence in CV, and a data sequence in another modality. In CV, $\boldsymbol{x}$ denotes a patch sequence, that $\boldsymbol{x} \in \mathbb{R}^{N\times(P^2C)}$ and N denotes the number of the patch, $P^2 \times C$ denotes the dimension of a patch vector. That means $\boldsymbol{x} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n]$, and $\boldsymbol{x}_i$ denote a patch, and $\boldsymbol{x}_i = [x_i^1, x_i^2, \cdots, x_i^{P^2\times C}]$. In this paper, we use $\boldsymbol{x}^k, \boldsymbol{x}_i^k$ to denote the different sequences and patches, and we use $\boldsymbol{x}^{v_i}$ to denote the different views of the patch. In NLP, $\boldsymbol{x} = [\boldsymbol{x_1}, \boldsymbol{x_2}, \cdots, \boldsymbol{x_L}]$ denotes the original sentence. we use $\boldsymbol{e} = [\boldsymbol{e_1}, \boldsymbol{e_2}, \cdots, \boldsymbol{e_L}]$ to denote the embedded sequence. Encoder and decoder are denoted by $f_\theta(\cdot)$ and $g_\phi(\cdot)$, where θ and ϕ are learnable parameters. In masked modeling, some tokens or patches of $\boldsymbol{x}$ are selected to mask, and we use $\mathcal{M} = \{0,1\}^N$ to denote the mask set, which means a masked sequence can be denoted as $\boldsymbol{x} \odot \mathcal{M} = [\boldsymbol{x_1}, \cdots, \boldsymbol{x_{i-1}}, 0, \boldsymbol{x_{i+1}}, \cdots, \boldsymbol{x_n}]$. The left visible patch(token) can be denoted as $\tilde{x} = x^N_{i=1,\mathbb{I}_{\mathcal{M}=1}}$ or $\tilde{e} = e^N_{i=1,\mathbb{I}_{\mathcal{M}=1}}$.

2.2 Self-Supervised Learning

In this subsection, we will give a brief introduction to the methods of SSL. Typically, SSL methods are universally divided into two categories [33], *i.e.*, Generative and Discriminative, as shown in Table 3.

Generative model usually encodes the input x into a latent variable z and decodes the latent variable z to reconstruct the input x with an encoder-decoder architecture. Autoregressive models typically model a series of regressions one by one for one input.

Auto-Regressive models typically model a series of regressions one by one for one input, where the current output depends on the previous inputs or outputs in the sequence. **GPT** [14] and **Transformer** [34] are both AR models. The learning object of the AR model can be formulated as:

$$\max_\theta p_\theta(\boldsymbol{x}) = \sum_{t=1}^{T} \log p_\theta(\boldsymbol{x}_t | \boldsymbol{x}_{1:t-1}), \tag{1}$$

where each variable is dependent on previous variables [33].

Auto-Encoder typically reconstructs the input from the corrupted input. The learning object of the AE model can be formulated as:

$$\min \mathcal{L}(\boldsymbol{x}, g_{\text{dec}}(f_{\text{enc}}(\boldsymbol{x}))). \tag{2}$$

Notations	Descriptions
$\mathbb{R}^{m\times n}$	Two-dimensional tensor space
$\mathbb{R}^{m\times n\times p}$	Three-dimensional tensor space
$\mathcal{N}$	Natural number set from 1 to N
$\boldsymbol{x}, \boldsymbol{x}_i$	A data sequence and its i-th element
$\boldsymbol{x}_{m:n}$	The subsequence from m to n in $\boldsymbol{x}$
$\boldsymbol{m}$	Encoded representations of masked patch/token
$\boldsymbol{z}$	Latent variables
$\mathcal{M} = \{0,1\}^N$	A set of masks for N elements
$\mathcal{M}_i$	The i-th element in set $\mathcal{M}$
$\tilde{\boldsymbol{x}}$	The set of visible tokens in masked sequence
$\theta, \omega, \gamma, \cdots$	Parameters of the deep neural networks
τ	Temperature parameter in contrastive learning
λ	Weights of loss functions
Natural Language Processing (NLP)	
$\boldsymbol{e}$	Embedded word tokens.
$\mathcal{V}, v_i$	Vocabulary set (or codebook) and its i-th elements
Computer Vision (CV)	
$\mathbf{X}$	Images X
$\mathbf{X}^v$	Different views of the image $\mathbf{X}$
$\boldsymbol{x}^{v_i}$	A patch sequence with different views.
$\boldsymbol{q}_\phi(\cdot\|\cdot)$	The quantization tokenizer
$\boldsymbol{p}_\psi(\cdot\|\cdot)$	The decoder to train the tokenizer
$f_\theta(\cdot)$	Encoder with parameter θ
$f_{\theta'}(\cdot)$	The teacher model with parameter θ'
$g_\theta(\cdot)$	Decoder of with parameter θ
$\nabla(\cdot)$	Gradient function
$\mathcal{T}(\cdot)$	The transformation function
$\mathbb{I}_{(\cdot)}$	An indicator function
$\mathcal{G}(\cdot)$	Generator in adversarial learning
$\mathcal{D}(\cdot)$	Discriminator in adversarial learning
$\mathcal{F}(\cdot)$	Fourier transform function
$p(\cdot)$	Probability density function
$p(\cdot\|\cdot)$	Conditional probability distribution
$\text{sg}(\cdot)$	Stop-gradient operation
$\langle\cdot,\cdot\rangle$	Inner product function
$\|\cdot\|$	Cardinality of the set
$\|\cdot\|$	Norm of the vector
$\mathcal{S}$	Similarity measurement function
$(\cdot)^T$	Transpose function
$\odot$	Element-wise multiplication operation

TABLE 1: Mathmetical notations.

We further divide the AE model into **denoising AE** and **masked AE**. The **denoising AE** model is trained to reconstruct clean data from noisy or corrupted input. By removing noise or corruption, the model learns robust representations. And a **masked auto-encoder** is trained to predict missing or masked portions of the input data. By reconstructing the missing parts, the model learns contextual representations.

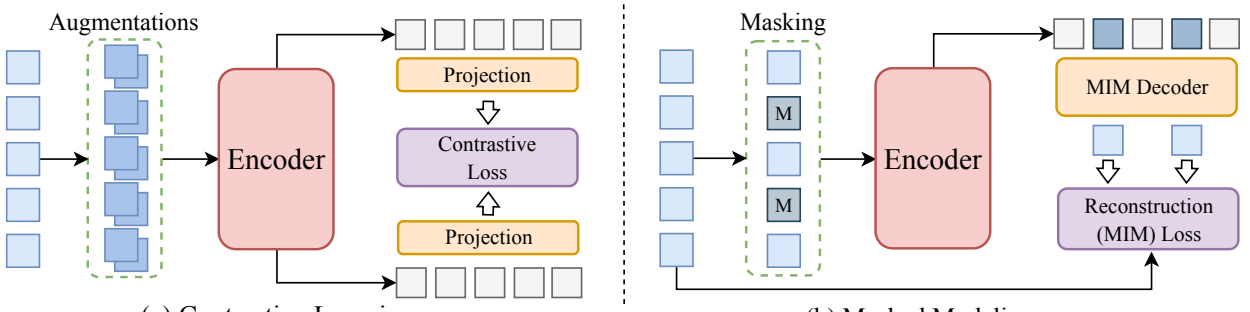
Flow Based model aims to learn densities $p(x)$ from data. Suppose a latent variable z follows a known distribution $p_Z(x)$ and define $z = f_\theta(x)$. The learning objective is to maximize the likelihood [33]:

$$\begin{aligned} &\max_\theta \sum_i \log p_\theta(x^{(i)}) \\ =&\max_\theta \sum_i \log p_Z(f_\theta(x^{(i)})) + \log \left| \frac{\partial f_\theta}{\partial x}(x^{(i)}) \right|. \end{aligned} \tag{3}$$

GAN-Based model so-called adversarial learning involves training two models in competition with each other, typically a generator $\mathcal{G}$ and discriminator $\mathcal{D}$, which can be formulated as:

$$\begin{aligned} &\min_{\mathcal{G}} \max_{\mathcal{D}} V(\mathcal{D}, \mathcal{G}) = \\ &\mathbb{E}_{x\sim p_{\text{data}}(x)}\big[\log \mathcal{D}(x)\big] + \mathbb{E}_{z\sim p_z(z)}\big[\log(1-\mathcal{D}(\mathcal{G}(z)))\big]. \end{aligned} \tag{4}$$

Diffusion-based model initially processes images through a series of Gaussian noise treatments, followed by



(a) Contrastive Learning (b) Masked Modeling

Fig. 2: Illustration of two popular self-supervised learning frameworks. For simplicity, the input data can be serialized and transformed into a sequence of embedded tokens. (a) Contrastive learning learns discriminative representation from two augmented views by aligning two projected tokens. (b) Masked modeling learns contextual information by the generative paradigm that reconstructs the masked tokens, which can be

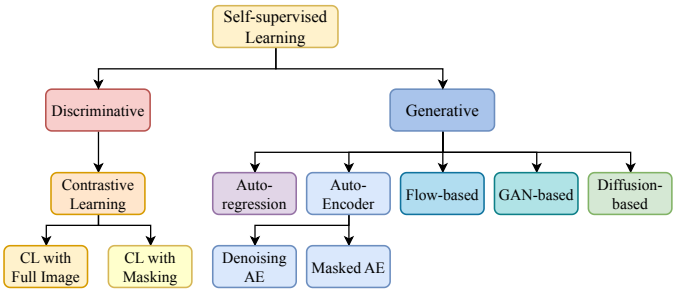


Fig. 3: Self-supervised learning is universally divided into generative and discriminative [33], and the generative model can be further divided into AR, AE, Flow-based, and GAN-based models, where AE model can be divided into Denoised AE and Masked AE.

restoration of the image through the model. The diffusion-based model process is divided into forward and reverse processes. The forward process treats the image with cumulative Gaussian noise, which can be modeled as follows:

$$\begin{aligned} q(x_t|x_{t-1}) &= \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t\mathbf{I}), q(x_{1:T}|x_0) \\ &= \prod_{t=1}^{T} q(x_t|x_{t-1}), \end{aligned} \tag{5}$$

in which β_t is mean coefficient. The reverse process of the diffusion-based model, which involves denoising and inference, has a learning objective that can be modeled as follows:

$$p_\theta(X_{0:T}) = p(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t); \tag{6}$$

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t,t), \Sigma_\theta(x_t,t)). \tag{7}$$

Discriminative model are typically formulated using contrastive learning objectives. The core idea in contrastive learning is to train encoders to produce similar representations for semantically related instances while distinguishing unrelated samples [33]. Contrasting at the **context-instance** level involves comparing the local feature, which is encoded, with the global representation from the identical sample. In contrast, the **instance-instance** contrast method is more focused on the representation at the instance level, examining the commonalities across multiple samples [33]. **InfoNCE** [6] is one of the basic loss functions in contrastive learning. It can be formulated as:

$$\mathcal{L}_{\text{infoNCE}} = -\mathbb{E}_{(\boldsymbol{x}^i,\boldsymbol{x}^j)\sim p(\boldsymbol{x})}\left[\frac{\exp(f(\boldsymbol{x}^i)^T f(\boldsymbol{x}^j)/\tau)}{\sum_{k=1}^{K}\exp(f(\boldsymbol{x}^i)^T f(\boldsymbol{x}^k)/\tau)}\right]. \tag{8}$$

2.3 Masked Modeling

Masked Language Modeling. Masked Language Modeling was first introduced in BERT. The central idea of MLM is to randomly mask tokens within a sentence and replace them with a Mask vector. The encoder then predicts the masked vector. We formally define the problem of MLM as follows: A sentence $\boldsymbol{x} = [\boldsymbol{x_1}, \boldsymbol{x_2}, \cdots, \boldsymbol{x_L}]$ is first tokenized as $\boldsymbol{e} = [\boldsymbol{e_1}, \boldsymbol{e_2}, \cdots \boldsymbol{e_L}]$ through a tokenizer $\boldsymbol{q}_\phi(\cdot|\cdot)$, in which L denotes the number of the tokens in this sentence. The masked sequence of the embedded sentence $\boldsymbol{e} \odot \mathcal{M}$ is fed into a Transformers encoder $f_\theta(\cdot)$. $m_i = f_\theta(\tilde{e})$ is the hidden state of the last layer at the masked position and can be regarded as a fusion of contextualized representations of surrounding tokens. And the MLM task [35] can be formulated mathematically as:

$$\mathcal{L}_{\text{MLM}}(x) = -\frac{1}{\|\mathcal{M}\|}\sum_{i\in\mathcal{N}}\mathbb{I}_{\{\mathcal{M}_i=1\}}\log\frac{\exp(m_i\cdot e_i)}{\sum_{k=1}^{|\mathcal{V}|}\exp(m_i\cdot e_k)}, \tag{9}$$

Masked Image Modeling. The core concept of Masked Image Modeling (MIM) aligns with that of MLM. It involves masking certain pixel regions of the input image and reconstructing the original image based on the unmasked portions. Given that images lack the tokenizer structure inherent in natural language, the intuitive approach is to reconstruct pixel values directly. However, due to the high redundancy and dimensionality of image pixel information, pixel-level reconstruction is often challenging. This has historically hindered the progress of MIM. It wasn't until the introduction of the ViT, which segments images into patches that MIM began to emerge as a feasible approach. We formally define the problem of MIM as follows: A image $\mathbf{X} \in \mathbb{R}^{H\times W\times C}$ is partitioned into multiple patches $\boldsymbol{x} \in \mathbb{R}^{N\times(P^2C)}$, $\boldsymbol{x} = [\boldsymbol{x_1}, \boldsymbol{x_2}, \cdots, \boldsymbol{x_N}]$ where N denotes the number of patch. Masked sequence can be denoted as $\boldsymbol{x}\odot\mathcal{M}$. The remaining unmasked patches $\tilde{\boldsymbol{x}}$ is used to reconstruct the original pixel through an encoder $f_\theta(\cdot)$ and a decoder $g_\theta(\cdot)$. We use m_i to denote the hidden layer at the masked

portion as NLP and $m_i = f_\theta(\tilde{x})$, The learning object can be formulated as:

$$\mathcal{L}_{\text{MIM}} = \frac{1}{\|\mathcal{M}\|} \sum_{i \in \mathcal{N}} \mathbb{I}_{\mathcal{M}_i=1} \|m_i - \boldsymbol{x_i}\|^2. \tag{10}$$

Beyond. Beyond CV and NLP, masked modeling can also be applied to various data structures and multimodal domains. The core idea is to mask parts of the input vector with mask tokens and then reconstruct the data through an encoder-decoder framework. Masked data modeling can be formally described as: given an input sequence x of any modality, we generate the corrupted sample $x \odot \mathcal{M}$ by replacing elements in x_m with mask tokens [MASK]. We use $\mathcal{S}(\cdot, \cdot)$ to denote the similarity between the predicted mask tokens and the original data. The learning object can be formulated as:

$$\mathcal{L}_{\text{MDM}} = \frac{1}{\|\mathcal{M}\|} \sum_{i \in \mathcal{N}} \mathbb{I}_{\{\mathcal{M}_i=1\}} \mathcal{S}(m_i, x_i), \tag{11}$$

in which $\mathcal{S}(\cdot, \cdot)$ can be MSE and other functions which measure the similarities.

3 Basic Framework and A Unified Perspective

This section will introduce a unified perspective for Masked Modeling(MM), offering a comprehensive categorization of MM research. This will be complemented by an in-depth exposition of the basic framework of MM, ensuring a profound understanding of its intricacies. **Since masked modeling has been most thoroughly explored and developed in CV with the most comprehensive techniques and has laid the foundation for developments across domains, this paper takes MIM as an example to elucidate MM from the perspective of CV.**

3.1 A Unified Perspective

Based on the current research on MIM for self-supervised pre-training, this paper conducts an in-depth investigation. It proposes a unified research framework and paradigm for MIM, providing a detailed classification of existing studies. The framework mainly consists of four modules, namely: **Mask**, **Target**, **Encoder**, and **Head**. An overview of our framework is visually presented in Figure 4. In the following subsections, we will elaborate on the specific contents of these four modules.

Mask: Mask module is to generate a mask set $\mathcal{M}$. The masked image can be denoted as $\boldsymbol{x} \odot \mathcal{M}$. Typical mask strategies include Random Mask, Attention Mask, Contextual Mask, and so on.

Target: The Target module's role is to generate supervisory signals. The target module can be formulated as: $\mathcal{T}(f_\omega(\boldsymbol{x}))$, $f_\omega(\cdot)$ is a model with parameter ω. Within this module, models like VQ-GAN [7] and dVAE can be utilized as tools to extract these signals, and different supervisory signals can lead to different model preferences.

Encoder: The Encoder $f_\theta(\cdot)$ is the target for MIM pre-training and can adopt various network architectures, such as Transformer, CNN, or a hybrid of both. The input for the Encoder can be a visible patch and a combination of visible and masked patches.

Head: The Head module's purpose is to compare the supervisory signals with the encoded features. The primary task of MIM is to reconstruct the original image, so the most common head is the MIM head, which reconstructs the original image or its features. Additionally, there's the Contrastive head, which employs contrastive learning to enhance the model's performance.

Based on the unified perspective we proposed, the MIM problem can be mathematically represented as:

$$\mathcal{L}_{\text{MIM}} = \mathcal{S}(\mathcal{T}_1(f_\omega(\boldsymbol{x})), \mathcal{T}_2(g_\gamma(f_\theta(\boldsymbol{x} \odot \mathcal{M})))). \tag{12}$$

Permuting and combining these four modules, we have meticulously categorized the research on MIM. The detailed classification is elaborated in Figure 3.

3.2 Basic Framework

iGPT [21]: The input image $\mathbf{X}$, when arranged according to pixel values and subsequently downsampled, forms a pixel sequence $\boldsymbol{x}$ that is fed into a Transformer structure identical to GPT-2 [14]. This model predicts the value of the next pixel $\boldsymbol{x}_t$ based on the current pixel value $\boldsymbol{x}_{1:t}$. Given that iGPT predicts pixel values in sequence, its masking approach can be considered as "**Basic Masking**", with the target being the **Token**. Based on GPT, the encoder of the iGPT is **Transformer** and the decoder is a **Linear MIM Head**. The loss of iGPT can be formulated as Eq. 1.

MAE [23]: The overview of MAE can be seen in Figure 5. The input image $\mathbf{X} \in \mathbb{R}^{H \times W \times C}$ is partitioned into multiple patches $\boldsymbol{x} \in \mathbb{R}^{N \times (P^2 C)}$, where approximately 75% of the patches are **Randomly Masked**. The remaining unmasked patches $\tilde{x}$ are then fed into the **Transformer Encoder** $f_\theta(\cdot)$ which generates the features. These features, in conjunction with the masked patches, are input into the **Transformer Decoder** $g_\omega(\cdot)$ for the purpose of reconstructing the **Pixels** of the original image. The quality of the reconstruction is measured using the MSE loss function. MAE [23] is formulated as:

$$\frac{1}{\|\mathcal{M}\|} \|g_\omega(f_\theta(\tilde{x}))) - \tilde{\boldsymbol{x}}\|^2. \tag{13}$$

iGPT [21] and MAE [23] represent two distinct basic frameworks within MIM research: iGPT [21] is based on the Auto-Regressive MIM research paradigm, while MAE is grounded in the Auto-Encoder paradigm. Both are categorized within the classification we proposed. The four modules of iGPT [21] can be classified as: **Basic Masking(Auto-Regressive Masking) + Transformer + Tokenizer + MIM Head**, whereas MAE can be categorized as **Basic Masking (Random) + Transformer + Pixel + MIM Head**. Table 2 summarizes the difference between iGPT and MAE.

4 Method

In this section, we will sequentially introduce the four essential modules for the MIM Framework, *i.e.*, Mask Strategy, Targets, Architecture of the encoder, and MIM Head. Within each module, there are many studies; we will provide a more detailed classification and summary.

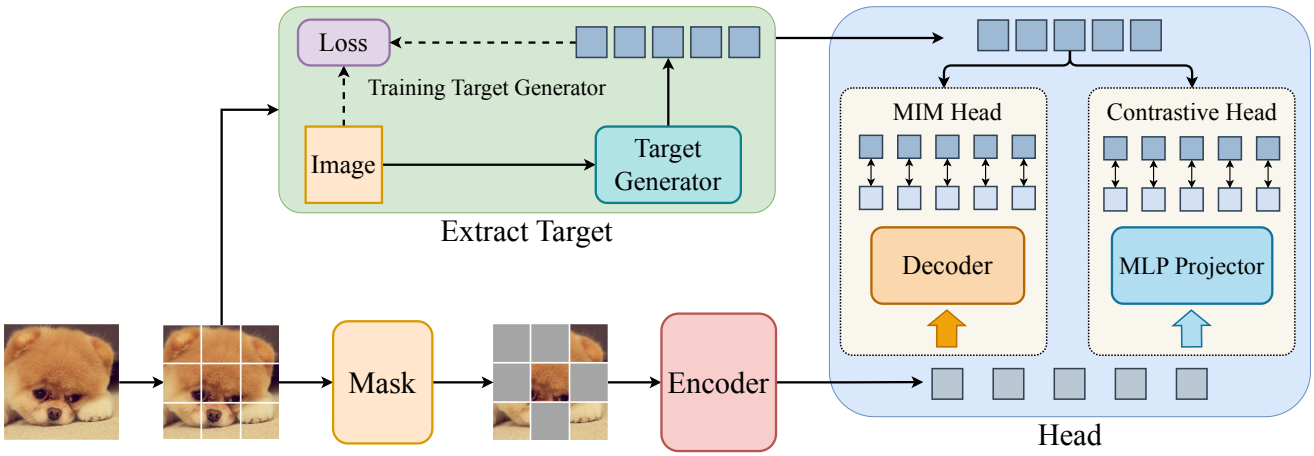


Fig. 4: The overview of the basic MIM framework, containing four building blocks with their internal components and functionalities. All MIM research can be summarized as innovations upon these four blocks, *i.e.*, Masking, Encoder, Target, and Head. Frameworks of masked modeling in other modalities are similar to this framework.

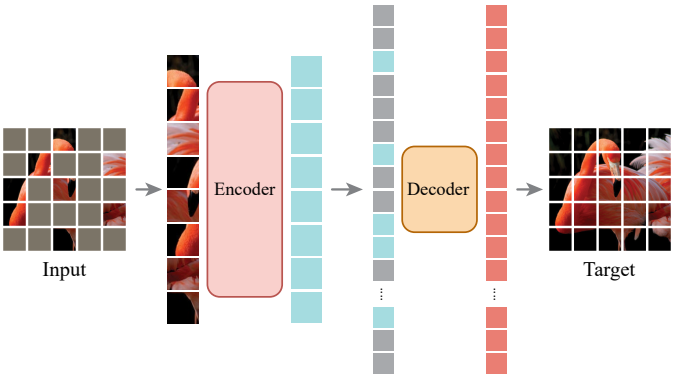


Fig. 5: MAE proposed a basic framework for MIM pre-training, where the visible patches are encoded while the encoded features are decoded together with masked patches to reconstruct the pixel. The figure is reproduced from [23].

Model	MAE	iGPT
Mask	Basic (Random)	Basic (AR Mask)
Encoder	Transformer	Transformer
Target	Pixel	Token
Head	MIM Head (Transformer)	MIM Head (Linear)
Category	BTPM	BTTM
Type	AE	AR

TABLE 2: This table outlines four parts of iGPT and MAE, where iGPT and MAE represent two different types of research. iGPT is a generative model based on AR methods, while MAE is based on AE.

4.1 Masking Strategy

This subsection will also spotlight typical masking strategies employed in MIM. For classification purposes, we bifurcate masking strategies into basic and advanced masking. *Basic masking, which encompasses pixel-wise predictions based on AR models and the Random Mask introduced by MAE, has been elaborated upon in Sec. 3.* Consequently, our ensuing discussion will primarily focus on Advanced Masking techniques. As illustrated in the accompanying figure, Advanced Masking can be further subdivided into four types: Hard Sampling, Mixture, Adversarial Mask, and Contextual Mask.

Remark: Despite the excellent performance, Mixture mask and Adversarial Mask have a more expensive computational cost. It can be concluded that an attention-based mask strategy usually performs better in mining hard samples and costs less.

4.1.1 Hard Sampling

In the **AttMask** [86] framework, a teacher model $f_{\theta'}$ is employed to extract the attention maps $\hat{a}$ and image features $f_\theta(\boldsymbol{x})$ from the input images $\mathbf{X}$ and patches $\boldsymbol{x}$. The student model f_θ then masks the regions with high attention scores in the attention maps. The reconstruct loss of AttMask can be formulated as:

$$\mathcal{L}_{\text{MIM}} = \sum_{v} \sum_{i \in \mathcal{N}} \mathbb{I}_{\{\mathcal{M}_i = 0\}} f_\theta(\boldsymbol{x}^v \odot \mathcal{M})_i \log f_{\theta'}(\boldsymbol{x}^v \odot \mathcal{M})_i. \quad (14)$$

Employing attentive masking, **AttMask not only delivers excellent results but also has relatively lower computational overhead**. In our classification, AttMask is categorized as **Advanced Mask + Transformer + Features + MIM Head (ATFM).**

HPM [82] introduces a teacher-student framework. The teacher model $f_{\theta'}$ predicts the reconstruction loss for each patch x_i, while the student model f_θ masks and reconstructs the image $\boldsymbol{x}$ using an "easy to hard" approach guided by the teacher model. In our classification, HPM [82] is categorized as **ATFM**. The object of HPM concludes a reconstruction loss and a prediction loss, and reconstruction loss is formulated as 13. l_1 distance and cross-entropy can also be used to measure the distance, and the predictor loss can be formulated as:

$$\mathcal{L}_{\text{pred}} = (g_\omega(f_\theta(\boldsymbol{x} \odot \mathcal{M})) - \mathcal{L}_{\text{rec}})^2 \odot (1 - \mathcal{M}), \quad (15)$$

Meanwhile, **SemMAE** [77] (Advanced Mask + Transformer + Pixel + MIM Head, ATPM) implements a semantic-based masking strategy through semantic information learned by ViT, **MILAN** [87] (ATFM) combines attention mask with an online feature as the target. **ObjMAE** [85] (ATFM) proposes an object-wise mask strategy that discards non-objective patches.

4.1.2 Mixture

MixedAE [102]: Building upon the framework proposed by MAE, MixedAE introduces a technique of blending portions from different images as input to the network. MixedAE

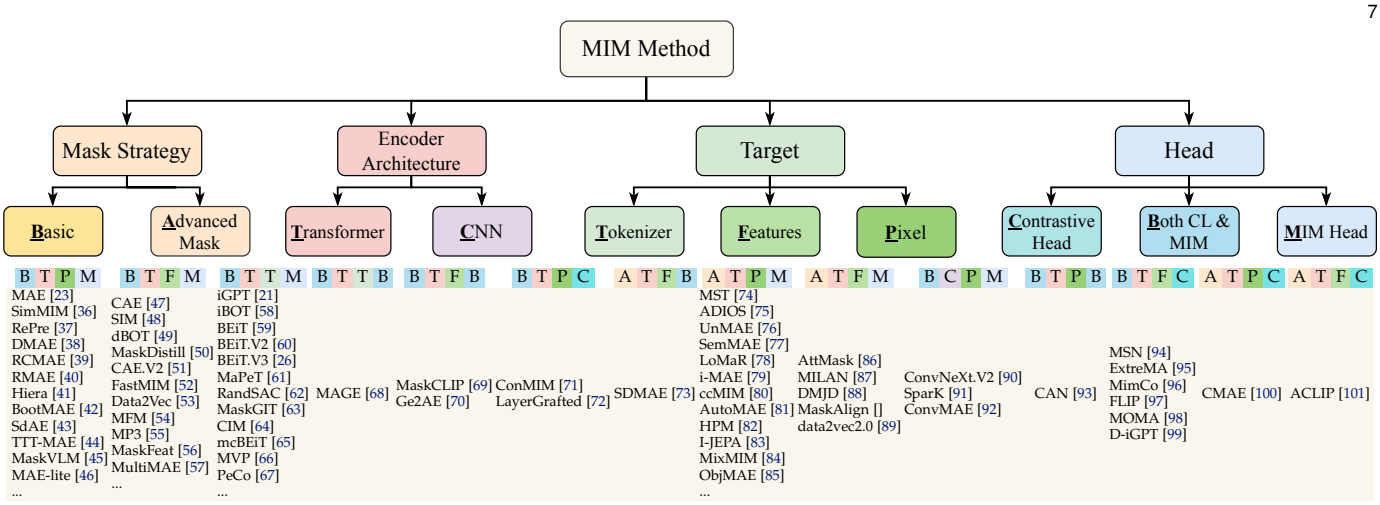


TABLE 3: We conducted a comprehensive survey of research related to MIM and categorized them according to the four modules we proposed. We divided the *Mask strategy* into Basic Mask and Advanced Mask, the *Encoder Architecture* into CNN and Transformer, the learning *Target* into Pixel, Tokenizer, and Feature, and the *Head* into MIM Head, Contrastive Head, and their combination. We use the initials of each module to form a category name; for example, MAE is categorized as **BTPM** because it uses a Transformer as the encoder structure, a Random Mask as the masking strategy, a Pixel as the target, and MIM Head for reconstruction. Note that we only list the widely known methods for **BTPM**, **BTFM**, **BTTM**, and **ATPM** because they cover most of the existing MIM algorithms. Refer to Table 4 for detailed information and categories.

enhances the model's representational capacity by incorporating contrastive learning. And MixedAE is categorized as **ATPM**. The loss function for this contrastive learning can be formulated as Eq. 21. **MixMIM** [84] (ATPM) utilizes both mixed masking and attention mask as masking methods and improves the network architecture to a hierarchical transformer. **i-MAE** [79] (ATPM) designs a mixed masking strategy for its input and simultaneously introduces a linear layer to separate the mixed input before reconstruction to improve the performance.

4.1.3 Adversarial

ADIOS [75] (ATPM) combines MIM with adversarial learning. Generator $\mathcal{G}$ produces images with different masks based on the original image, while Discriminator $\mathcal{D}$ aligns the generated images with the original ones. Since ADIOS does not rely on the block construction of the Transformer, it can be implemented in the backbone of CNNs. **AutoMAE** [81] (ATPM), on the other hand, introduces a Mask Generator based on the MAE architecture to generate different mask strategies. The encoder adaptively reconstructs the original image based on different mask methods.

4.1.4 Contextual Masking

UnMAE [76] (ATPM) proposes a Uniform Masking strategy for masking, with the selection of the masked portion consisting of two parts: Uniform Sampling and Secondary Masking. The former randomly samples a patch from a 2x2 grid, while the latter randomly masks a portion of the already sampled area. Additionally, UnMAE supports a pyramid-structured Transformer architecture. **LoMaR** [78] (ATPM), on the other hand, builds upon MAE by using small-window patches for local reconstruction prediction, improving efficiency and accuracy compared to MAE.

4.2 Different Targets

In this subsection, we will delve into the targets used during MIM training. For classification purposes, we categorize these targets into three main types: tokenizer, pixel, and features. Delving deeper, these categories can be further detailed, with comprehensive explanations provided in the accompanying Figure 6.

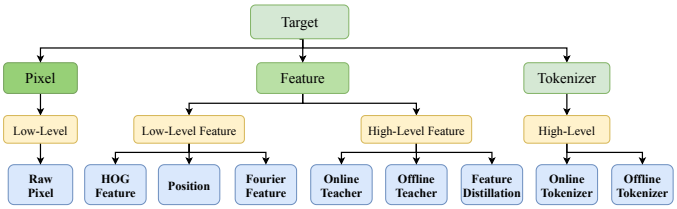


Fig. 6: The types of the MIM target include three categories, that is Pixel, Feature and Tokenizer.

4.2.1 Raw Pixel

Raw Pixel is the most fundamental target in MIM. Classic models like MAE and **SimMIM** [36] (BTPM) are based on Raw Pixel for image reconstruction. **I-JEPA** [83] (ATPM) uses a Context Patch as the input for the Encoder, and the reconstruction target is the three different patches adjacent to the Context Patch. By reconstructing through the Context Patch, I-JEPA can achieve better contextual representation capabilities while also reducing computational overhead.

4.2.2 Tokenizer

A tokenizer is a mapping function $\boldsymbol{q}_\phi(\boldsymbol{z}|\boldsymbol{x})$ that encodes image $\mathbf{X} \in \mathbb{R}^{H\times W\times C}$ into $z = z_1, \ldots, z_{|\mathcal{V}|} \in \mathcal{V}^{h\times w}$, where the vocabulary $\mathcal{V} = \{1, \ldots, |\mathcal{V}|\}$ contains token indices. These latent variables represent high-level semantic features of certain parts of the image. Hence, we can represent an image based on the dictionary $\mathcal{V}$, which can be used as the supervisory signal for MIM. The tokenizer $\boldsymbol{q}_\phi(\boldsymbol{z}|\boldsymbol{x})$ maps image pixels x into discrete tokens z according to a visual codebook [103] (*i.e.*, vocabulary), and decoder $\boldsymbol{p}_\psi(\boldsymbol{x}|\boldsymbol{z})$ learns to reconstruct the image based on visual tokens $\boldsymbol{z}$ [59]. The learning objective of the tokenizer can be formulated as:

$$\min \mathbb{E}_{\boldsymbol{z}\sim\boldsymbol{q}_\phi(\boldsymbol{z}|\boldsymbol{x})}(\log \boldsymbol{p}_\psi(\boldsymbol{x}|\boldsymbol{z})) \tag{16}$$

The training methods of tokenizer conclude **dVAE** and **VQ-GAN** [7].

BEiT [59] (Basic Mask + Transformer +Tokenizer + MIM Head, BTTM) : In first stage, BEiT discretely encodes image $\mathbf{X} \in \mathbb{R}^{H\times W\times C}$ into $z = z_1, \ldots, z_N \in \mathcal{V}^{h\times w}$, where the vocabulary $\mathcal{V} = \{1, \ldots, |\mathcal{V}|\}$ contains discrete token indices. After the tokenizer is pre-trained, The encoder f encodes the unmasked regions of an image, and encoded features are then passed through the MIM Head, with discrete image tokens serving as the supervision signal for learning. The Learning object of **BEiT** [59] can be formulated as:

$$\max \sum_{\text{dataset}} \mathbb{E}_{\mathcal{M}} \left[\sum_{i\in\mathcal{N}} \mathbb{I}_{\{\mathcal{M}_i=0\}} \log p_{\text{MIM}}(z_i|\boldsymbol{x}\odot\mathcal{M}) \right], \quad (17)$$

where $\mathcal{D}$ denotes the traning corpus.

iBOT [58] (BTTM) formulate MIM as a knowledge-distillation task and perform self-distillation using a teacher-student framework. The online tokenizer, jointly optimized with MIM, progressively captures high-level visual semantics and eliminates the need for a separate pre-training stage. The teacher model is updated by the student model with EMA as 20. Building on the framework of BEiT, **BEiTv2** [60] (BTTM) employs distillation on VQ to transform the discrete semantic space into compact codes. Building further upon BEiTv2, **BEiTv3** [26] (BTTM) integrates MOE and multimodality to design specialized tokenizers for vision, language and vision-language tasks and scales up the model. **Peco** [67] (BTTM)utilizes a perceptual prediction target to train a perceptual codebook. **mc-BEiT** [65] (BCTM) represents a masked patch with a soft probability of vector instead of a unique token id. **CIM** [104] (BTTM) proposed an encoder-enhancer architecture in which a small pre-trained BEiT is used as an encoder, and a CNN-based model can be applied to the enhancer. Pixel reconstruction and GAN loss are used in CIM, respectively.

4.2.3 Low-Level Features

Low-level image features typically come in three types: HOG Features, positional information of the image, and Fourier Features.

HOG Features. **MaskFeat** [56](Basic Mask + Transformer + Feature + MIM Head, BTFM) proposes a framework based on MAE. Notably, the supervision signal for training the model is derived from the HOG features of the original image. **FastMIM** [52] (BTFM) designs a Hierarchical transformer and utilizes HOG features as the target.

Position. **DILEMMA** [105](BTFM) employs a teacher model to generate position encoding. The student model is trained to predict new positions and judge whether the prediction is true or not. **MP3** [106] (BTFM) trains a masked transformer to predict the position of patches using MAE as a loss function. **SDMAE** [107] (ATFM) combines position prediction loss, pixel loss, and global contrastive loss to train its backbone. **DropPos** [108] (BTFM) randomly selects a subset of patches and replaces their positional encodings with mask tokens. The positional encodings are then reconstructed.

Fourier Features. Models combined with Fourier Features can generally be divided into two main categories. **Calculating Loss In Fourier domain**: **Ge2AE** [70] (Basic Mask + Transformer + Feature + Both Head, BTFB) reconstructs in the Fourier domain while computing both contrastive loss and reconstruction loss. **A2MIM** [109] (BCFM) utilizes The intermediate layer features of the CNN-based and ViT- based encoder to reconstruct ground truth in the spatiotemporal domain and frequency domain. The discrete Fourier transform of each channel is defined as:

$$\mathcal{F}_{(u,v)} = \sum_{H,W} x(h,w) e^{-2\pi j(\frac{uh}{H}+\frac{vw}{W})}. \quad (18)$$

The learning objective in the frequency domain can be formulated as follows:

$$\mathcal{L}_{freq} = \sum_{C,H,W} \omega \big\| \mathcal{F}(x\odot\mathcal{M} + \text{de}(x)\odot(1-\mathcal{M})) - \mathcal{F}(x) \big\|, \quad (19)$$

where $\omega = \omega(u,v)$ is a dynamic frequency weighting matrix. **Masking In Fourier Domain**: **MFM** [54] (BTFM) masks in the frequency domain, adds noise, and then reconstructs the image. **MSCN** [110](BTFM), after masking in the frequency domain, integrates with contrastive learning and employs a contrastive loss. **PixMIM** [111] (BTFM) both reconstruct the image in both spatial and frequency domain.

4.2.4 High-Level Features

Research that takes high-level features extracted from images as training targets is often associated with knowledge distillation or teacher models, using the teacher model or distilled image features as training targets. This type of research can typically be categorized into off-line teachers, on-line teachers, and those combined with knowledge distillation.

Offline Teacher. MILAN [87] (ATFM) utilizes CLIP [25] to generate attention maps to guide the model to mask and generate features as the target. **MOMA** [98] (Basic Mask + Transformer + Feature + Contrastive Head, BTFC)builds upon the MAE and uses pre-trained Multiple Teacher features as the prediction target. **Img2vec** [112] (BTFM) uses a pre-trained ConvNet as the teacher model to extract features. Based on the MAE framework, it reconstructs patches and combines contrastive learning to compute the global loss. **TinyMIM** [113] (BTFM) discovered that using the intermediate layer features of the teacher model often yields better results, with a smaller gap to downstream tasks. As a result, TinyMIM utilizes features of each layer to be targeted.

Online Teacher. data2vec [53] (BTFM) utilizes contextualized representations of the online teacher model and combines several modalities, including speech, natural language process and computer vision. data2vec updates its parameter with the EMA:

$$\boldsymbol{\theta}' \leftarrow \tau\boldsymbol{\theta}' + (1-\tau)\boldsymbol{\theta} \quad (20)$$

data2vec.v2 [89] (ATFM), building on the foundation of data2vec, introduces a multi-mask training method to enhance efficiency and reduce computational costs. **dBOT** [49] (BTFM), based on iBOT, has designed a multi-stage distillation scheme, concluding that teacher models with different parameters tend to have consistent performance in student models after multi-stage distillation. **BootMAE** [42] (BTPM), while using online features as prediction targets, also adds

the task of reconstructing image pixels. Unlike directly calculating the loss between features, **RC-MAE** [39] (BTPM) inputs the masked image into two transformer encoders with EMA-updated parameters. It then computes the contrastive loss of the reconstructed image, supplemented by a task of pixel-level image reconstruction. **MaskDistill** [50] (BTFM) **MaskCLIP** [69] (Basic Mask + Transformer + Feature + Both Head, BTFB) integrates multiple techniques, including MIM, multi-modality, online features, and contrastive learning.

Feature Distillation. **DMJD** [88] (ATFM) introduced a disjoint mask and simultaneously trained the encoder using features distillation and prediction reconstruction methods. **CAE.v2** [51](BTFM) distills CLIP and is supplemented with a task to reconstruct CLIP features. **SdAE** [43] (BTPM) delves into creating effective views for the teacher branch and proposes a multi-fold masking strategy to reduce computational complexity.

4.3 Different Network Architecture

Compared to the traditional Transformer, hierarchical Transformers often have lower computational overhead, faster training speeds, and better generalization capabilities on downstream tasks. Some research focuses on enhancing the encoder structure to improve the operational efficiency of MIM or devising novel masking techniques to make MIM adaptable to various network architectures.

Transfer encoder to hierarchical vision transformer: **GreenMIM** [114] (BTPM) inputs the masked image $\mathbf{X} \odot \mathcal{M}$ into a Hierarchical Transformer encoder. To reduce unnecessary computations in areas that are masked or do not contain useful information, the sparse convolution is introduced to discard invisible patches and only processes on the visible patches, achieving patch merging, similar to Figure 7. The processed data is then input into the Transformer decoder to reconstruct the original image. **HiViT** [115] (BTPM) removes local inter-unit operations, resulting in structurally simple hierarchical vision transformers. **Hiera** [41] (BTPM) eliminates the need for many of the complex components found in other hierarchical vision transformers and achieves superior accuracy. **ConvMAE** [92] (Basic Mask + CNN + Pixel + MIM Head, BCPM) proposes a multi-scale hybrid convolution-transformer, employs a masked convolution to prevent information leakage in the convolution blocks and a block-wise mask to reduce the computational cost. **SparseMAE** [116] (BCPM) introduces sparse MHSA and FFN blocks for sparse pre-training.

Make MIM Compatible with Convolutional Neural Networks: **CIM** [104] (Basic Mask + CNN + Tokenizer + MIM Head, BCTM) employs an auxiliary generator equipped with a compact trainable BEiT to corrupt the input images, thereby enhancing the network's capability to either restore the original image pixels or predict whether each visual token has been replaced by a sample from the generator. Due to CIM's approach of using an auxiliary generator to corrupt the input, there's no need for specific input formats or preprocessing, which is compatible with CNNs. Since the objective of the enhancing network is to either restore the original image or predict if each visual token has been replaced by a generator sample, only forward propagation is required, ensuring compatibility with CNN architectures. **A2MIM** [109] (BCFM) introduces a unified architecture compatible with both Transformers and CNNs. Specifically, A2MIM posits that masking at the block embedding layer aligns well with the attention mechanism of Transformers, offering robustness against occlusion. For CNNs, masking at the network's input stages leads to low-order interactions, undermining CNN's context extraction capability. Therefore, the authors suggest masking intermediate features encompassing semantic and spatial information, allowing the mask token to encode interactions with a moderate number of tokens.

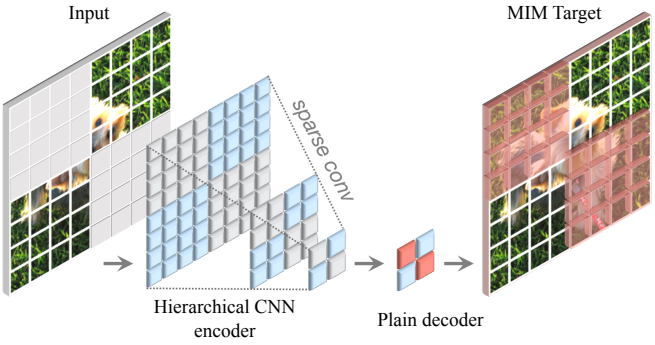


Fig. 7: Illustration of MIM for CNN architectures with the sparse convolutions and masking [90], [91], where the encoder only aggregates information of visible tokens. The figure is reproduced from [90].

Specially designed for Convolutional Neural Network: **Spark** [91] (BCPM) identified that the inability of convolutional operations to recognize irregularly randomly masked input images and the single-scale nature of BERT pre-training is fundamentally inconsistent with the hierarchical structure of convolutional networks, which is the primary reason MIM cannot be implemented on CNNs. To address this, Spark proposed treating unmasked pixels as 3D point clouds and using sparse convolution for encoding, allowing the model to operate on irregularly masked images, as in Figure 7. To integrate the hierarchical structure of convolutional networks, they introduced a hierarchical decoder to reconstruct images from multi-scale features. The authors validated this approach on traditional convolutional neural network models such as ResNet and ConvNeXt, and its performance showed significant improvements compared to contrastive learning and Transformer-based MIM. **ConvNext.v2** [90] (BCPM) introduces a fully convolutional masked autoencoder. Its core structure is based on ConvNext, where the convolution operation is transformed into sparse convolution. The decoder employs a lightweight ConvNext block, which simultaneously processes encoded pixels and masked tokens for image reconstruction, effectively migrating MIM to the CNN structure. Additionally, ConvNext.v2 proposes a Global Response Normalization layer that normalizes the feature map on each channel, capable of handling batches of any size. The framework of ConvNext.v2 is shown in Figure 7.

4.4 Head

The choice of the head in MIM exhibits significant variations. In our classification, we distinguish the heads into three categories: Contrastive Head, MIM Head, and a combination of Both Contrastive Head and MIM Head. It's essential to

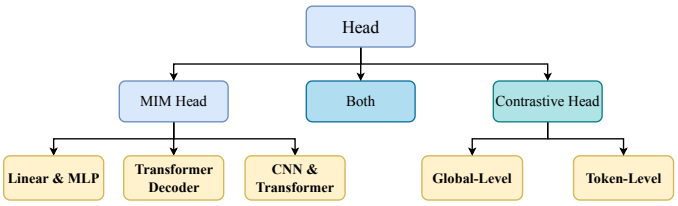


Fig. 8: The types of MIM Head include Linear or MLP, Transformer, or a combination of CNN and Transformer. The Contrastive Head section is categorized based on the algorithm type into Token-level and Global-level.

highlight that both the MIM Head and Contrastive Head can have diverse internal architectures. The specifics of these structures are visually represented in the provided figure. In the following sections, we will bifurcate our discussion into two primary segments, focusing separately on the MIM Head and the Contrastive Head.

4.4.1 MIM Decoder

Linear or MLP: **SimMIM** [36] (BTPM) essentially adopts the framework of MAE but with several significant modifications. In SimMIM, the encoder processes both the visible patches and the masked tokens simultaneously. Remarkably, SimMIM's decoder achieves satisfactory results using just a **Linear Prediction Head**. A detailed comparison between SimMIM and MAE can be found in the provided table. Other models utilize linear layers such as **BEiT** [59], **BEiT.v2** [60], **data2vec** [53] and so on.

Transformer Decoder: The Transformer Decoder is the most widely used in MIM. More details are represented in Figure 8.

Combined Transformer with CNN: **LocalMAE** [117] (BTFM) employs intermediate features from multiple stages for multi-scale reconstruction. In the reconstruction segment, LocalMAE introduces a Transformer-Deconvolution-MLP architecture for the task.

Remark: One might wonder why certain models can achieve commendable reconstruction results with just a simple, lightweight **Linear Head**, while others necessitate a more intricate **Transformer decoder** for reconstruction. The crux of the matter lies in whether the input to the Encoder includes the masked tokens. If the patches inputted to the Encoder encompass the masked tokens, these tokens can interact with the visible patches within the encoder. This interaction allows the encoder to capture certain image information early on, making it feasible to reconstruct the original image effectively with just a Linear Head. Conversely, if the encoder doesn't receive the masked tokens, these tokens would then need to interact with the visible patches within a more complex Transformer decoder to reconstruct the original image. Figure 9 compares SimMIM and MAE in detail.

4.4.2 Combined with Contrastive Head

There are typically two approaches combining contrastive learning and masked language modeling: The first incorporates masked images as a data augmentation technique and applies them within the contrastive learning framework to benefit contrastive learning. The second utilizes the standard masked language modeling framework and adds contrastive learning objectives in the prediction head to benefit masked language modeling. In this section, we will detail both lines of work and elaborate on the network architecture for the contrastive prediction head.

Mask as Data Augmentation: **MSN** [94] (BTFC) utilizes masked images as a data augmentation technique and incorporates them into the framework of PCL [118]. **MSCN** [110] (BTFM) and **Mimco** [96] (BTFC) incorporate masked images as data augmentation into the frameworks of SimCLR and BYOL contrastive learning, respectively, to benefit contrastive representation learning. This achieves an integration of masked modeling and contrastive learning.

Add Contrastive Loss: This line of work builds upon masked modeling and incorporates a contrastive prediction head by adding or replacing the original MIM head. It can be categorized into two groups: token-level contrastive learning and global-level contrastive learning. Details are illustrated in Figure 10. **Token Level Contrastive**: **ConMIM** [71] (Basic Mask + Transformer + Pixel + Contrastive Head, BTPC) utilizes two transformer encoders, one for masked images and another for unmasked images. The branch that takes the masked images as input predicts the original images. The features obtained from the prediction are contrasted with those from the unmasked images through contrastive learning. The contrastive loss can be written as:

$$\mathcal{L}_{\text{con}}(x) = -\log \frac{\exp\left(\langle f(\boldsymbol{x}_i), \boldsymbol{x}_j \rangle / \tau\right)}{\sum_{k=1}^{2N} \mathbb{I}_{\{k \neq i\}} \exp\left(\langle f(\boldsymbol{x}_i), \boldsymbol{x}_k \rangle / \tau\right)}, \quad (21)$$

Global Level Contrastive: **ccMIM** [80] (ATPM) employs an attention mechanism to rank each patch in the image x and selects the more challenging parts as masked set $\mathcal{M}$ for reconstruction by masking. Subsequently, global-level contrastive learning is performed on the CLS token. **CAN** [93] (Basic Mask +Transformer +Pixel +Both, BTPB) adds Gaussian noise to the masked images. Building upon MAE, it performs pooling before reconstructing the image and computes a global-level contrastive loss.

Architecture of Contrastive Head: The Contrastive Head is consistent with the head in classic contrastive learning architectures, often consisting of multiple **MLP** or **FNNs**. They typically have an appended **BN** layer, as seen in models like **SimCLR** [19] and **BYOL** [20]. A characteristic feature of these heads is that they often upscale the dimensions, having a larger number of channels. Some Contrastive Heads utilize the Transformer Decoder. For research that employs the **Transformer Decoder** as the Contrastive Head, considerations usually revolve around the depth and width of the Transformer blocks.

4.5 Theoretical Foundation

Supervised learning, often referred to as statistical learning methods, typically possesses profound mathematical theoretical guarantees, providing precise mathematical conditions under which learning is assuredly successful. Training and test datasets usually stem from the assumption of independent and identically distributed statistics. As the number of training iterations increases, one can often achieve lower training and test losses. This is because supervised learning is relatively straightforward. In contrast, unsupervised learning lacks the simple and intuitive theoretical guarantees present in supervised learning. Intuitively, we believe that the essence

Model	MAE	SimMIM
Mask	Random	Random
Encoder	Transformer	Transformer
Target	Raw Pixel	Raw Pixel
Input	Visible	Visible and Masked
Head	Transformer	Linear
Method	Auto-Encoder	Auto-Encoder

Fig. 9: The most significant difference between SimMIM and MAE lies in whether the input to the encoder includes the masked tokens and the structure of the MIM Head. An in-depth explanation of this aspect can be found in the designated Sec. 4.4.1.

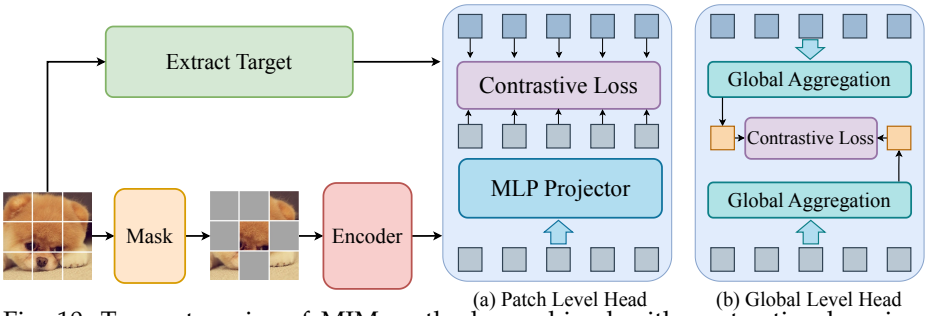


Fig. 10: Two categories of MIM methods combined with contrastive learning: token-level and global-level contrasting heads. For the token-level head, encoded tokens are subjected to an MLP Projection to compute contrastive learning loss. The global-level head aggregates global information on MIM targets and tokens before calculating the contrastive loss.

of unsupervised learning is a form of information **compression**. The compression algorithms learned from the training set represent the universal knowledge and structure inherent within the data. The way to evaluate these compression algorithms is to determine whether they extract all the knowledge from unlabeled data, i.e., whether they provide as much assistance as possible and yield the maximum benefit. We will elucidate and summarize the theoretical foundations of Masked modeling from three perspectives.

From Contrastive Learning: **Layer Grafted** [72] (Random + Transformer + Pixel + Contrastive Head, RTPC) finds that MIM and CL are suitable for lower and higher layers, respectively. The authors design a gradient surgery experiment by computing the cosine similarity between gradients of two tasks following [119] and verify the MIM loss and CL loss have different targets to optimize. The cosine similarity can be defined as:

$$\boldsymbol{C}_{\mathrm{MIM,CL}}(x)=\frac{\nabla_\theta L_{\mathrm{MIM}}(x)^T}{\|\nabla_\theta L_{\mathrm{MIM}}(x)\|}\frac{\nabla_\theta L_{\mathrm{CL}}(x)}{\|\nabla_\theta L_{\mathrm{CL}}(x)\|}. \tag{22}$$

They propose a "sequential cascade" approach where early layers are first trained under one MIM loss, and then later layers continue to be trained under another CL loss. That is:

$$\mathcal{L}_{\mathrm{MIM}}\rightarrow\mathcal{L}_{\mathrm{CL}}. \tag{23}$$

[120] demonstrates that the mask loss exhibits a lower bound compared to the align loss in contrastive learning, making it more effective than aligning within contrastive learning.

$$\mathcal{L}_{\mathrm{MAE}}\geq\frac{1}{2}\mathcal{L}_{\mathrm{align}}-\epsilon+\mathrm{const}. \tag{24}$$

Subsequently, a uniform loss, akin to that in contrastive learning, is incorporated into the mask loss.

From Masking: [121] models MIM as a hierarchical latent variable model. The objective of MIM is to recover the latent variable z shared between visible patches and invisible patches based on the lower-level visible patches. This latent variable encapsulates the information shared between the visible patch and the invisible portions. Both a very low mask ratio and an extremely high mask ratio tend to make the model focus on recovering low-level latent variable information, making it challenging to learn higher-level semantic features. Therefore, the mask ratio in MAE, which lies between the two, can assist the model in capturing higher-level latent variable information, enhancing its representation capability.

From Empirical Study: Many studies have extensively explored certain characteristics of masked language modeling through numerous experiments and obtained some valuable conclusions. [122] and [123] verified through extensive experiments that, compared to other self-supervised methods like jigsaw puzzles and image inpainting, masked language models demonstrate better transferability and superior performance on tasks like pose estimation, depth prediction, video object tracking, and object detection. [124] showed that masked models tend to underperform and are prone to overfitting on small datasets. As the dataset grows larger, the performance improvement of masked language models accelerates. [125] suggested that the efficacy of masked language modeling stems largely from the masking operation itself as the key to good performance, while different masking strategies contribute limited improvements.

We summarize some conclusions:

1) **From Contrastive Learning**: Unlike contrastive learning, MIM tends to capture low-level features and exhibits a strong local bias, while contrastive learning leans towards high-level features. This naturally explains why the development of contrastive learning preceded MIM. Before the advent of ViT, the dominant architecture in the visual domain was CNN, which inherently has a strong local bias. This made it complementary to contrastive learning, enhancing each other's strengths. However, both MIM and CNN share this pronounced local bias, leading to suboptimal performance of MIM on CNN architectures. With the rise of ViT, which emphasizes capturing global information, it pairs better with MIM, propelling MIM to the forefront of SSL algorithms.
2) **From Masking**: Masking is the most fundamental and crucial technique in MIM. Compared to NLP, the visual domain typically employs a higher mask ratio. This is because, in contrast to language, image information is more redundant. A small masking ratio doesn't significantly impact the overall semantic understanding of an image. Therefore, a larger mask ratio is used to obscure some of the image's key information, increasing the difficulty of the reconstruction task and enabling the model to learn more robust representations.
3) **From Empirical Study**: Models based on MIM exhibit certain characteristics and preferences. For instance, they rely more on large-scale data for training and tend to learn better representations with larger datasets. Masked

modeling performs better on tasks that require more detailed visual information, such as video object tracking and pose estimation. These tasks demand the model's ability to capture low-level information.

4.6 Auto-Regressive For Generation

The majority of MIM research is based on AE for generative SSL; however, AR modeling has always been one of the important methods in generative self-supervision. Consequently, a significant body of research combines AR generation with MIM, achieving both representation learning and generative tasks. In this section, we will introduce classic AR generative model architectures and then discuss research paradigms that integrate representation learning with AR. In Figure 11, we provide a detailed comparison of the differences between these two research paradigms.

4.6.1 VQ-Based Generation

Vector Quantization(VQ) is a significant technique in generative models, where it quantizes the continuous feature representations output by the encoder into discrete vectors in a codebook.

VQ-VAE [6] introduces a Generative Framework that encompasses both generation and training processes. During training, VQ-VAE encodes image pixels into feature vectors, searching for the token in the codebook that is closest to the feature vector. The image is then reconstructed through the decoder. Therefore, the training loss includes the quantization loss of the vectors and the reconstruction loss:

$$\begin{aligned}\mathcal{L}_{\text{VQ-VAE}} =&\|x - g(v_q)\|^2 + \|sg[f(x)] - v_q\|^2 \\ &+\beta\|f(x) - sg[v_q]\|^2.\end{aligned} \tag{25}$$

where β is a hyperparameter used to control the weights of the two losses. The generation process involves producing feature vectors through **PixelCNN**, followed by vector quantization of these feature vectors, and then generating new images via the decoder.

Subsequent research based on VQ-VAE has two main focuses: one is to improve the training process to enhance the quality of image generation, and the other is to improve the generation process to increase the speed of image generation.

Improve Generation Quality: **VQ-GAN** [7] is based on the VQ-VAE architecture, using GPT-2 as the generator in the workflow to produce discrete encodings. To enhance the reconstruction performance of the Decoder, an adversarial loss is added to the reconstruction loss. The learning object consists of the reconstruct loss and adversarial loss, which can be formulated as:

$$\begin{aligned}\mathcal{Q}^* = \min_{f,g,\mathcal{V}} \max_{\mathcal{D}} \mathbb{E}_{x\sim p(x)}\Big[&\mathcal{L}_{\text{VQ}}(f, g, \mathcal{V}) \\ &+\lambda\mathcal{L}_{\text{GAN}}(\{f, g, \mathcal{V}\}, \mathcal{D})\Big],\end{aligned} \tag{26}$$

The process of generation is based on the GPT-2, and the process can be formulated as:

$$\max_{\theta} p_\theta(\boldsymbol{v}) = \sum_{t=1}^{T} \log p_\theta(\boldsymbol{v}_t|\boldsymbol{v}_{1:t-1}). \tag{27}$$

Improve Generation Speed: Based on the VQ-VAE and VQ-GAN, **MaskGIT** [63] learns to predict randomly masked tokens by attending to tokens from all directions. In the inference stage, the model initially generates all tokens of the image simultaneously and subsequently refines the image iteratively based on prior generations. **RandSAC** [62] adopts a strategy of segmenting tokens into hierarchical sections. Within each section, it employs a parallel prediction mechanism akin to BERT [5], while between different sections, it utilizes a sequential prediction approach reminiscent of GPT [14]. By randomizing the sequencing of sections and leveraging parallel training, it significantly enhances computational efficiency.

4.6.2 Combining Pre-training with Image Generation

iGPT is one of the earliest models considered to perform both generation and pre-training. By predicting pixel values through the Transformer's autoregressive approach, iGPT achieves image generation capabilities. The unsupervised learning on large-scale unlabeled data makes iGPT a pre-trained model, which can achieve good results on downstream tasks through fine-tuning. **MAGE** [68] first maps images to tokens in a discrete latent space using VQ-GAN, then performs masked image modeling by masking tokens in the latent space. The training objective is to reconstruct unmasked tokens. In this way, MAGE is able to learn representations via masked image modeling in the latent space while achieving image generation. A contrastive loss is used in the latent space to improve the performance of the model. **RCG** [127] trains a representation generator by adding noise to the encoded representation and then removing it. Subsequently, it utilizes the generated representation within the MAGE architecture to achieve pixel generation, which unifies pre-training and representation learning.

4.7 Vision Fundation Model

Current research in artificial intelligence and deep learning is increasingly oriented towards the integration of data from multiple modalities. Consequently, multimodal research has emerged as one of the most significant directions in the field of artificial intelligence. We categorize multimodal studies into three broad types. The first category involves the use of multimodal data for pre-training, where the focus is on extending visual network architectures to multimodal contexts and exploring the upper limits of model capabilities through scaling up, as summarized in Table 8. The second category primarily concentrates on the generation of multimodal data, encompassing tasks such as text-to-image conversion, as summarized in Table 9. The third category represents a vision generalist model, aiming to unify various visual tasks under a single network architecture.

4.7.1 Pre-train With Multimodality

Masked Modeling Methods. **VL-BERT** [24] incorporates visual and linguistic inputs into a BERT-based architecture, allowing early and unrestricted interactions between modalities for joint representation learning. **MaskVLM** [45] applies to mask to image-text pairs, and then the masked images and masked texts are separately inputted into the image encoder and text encoder. Furthermore, a multimodal encoder is designed to encode the masked text and image, followed by simultaneous reconstruction of both the image

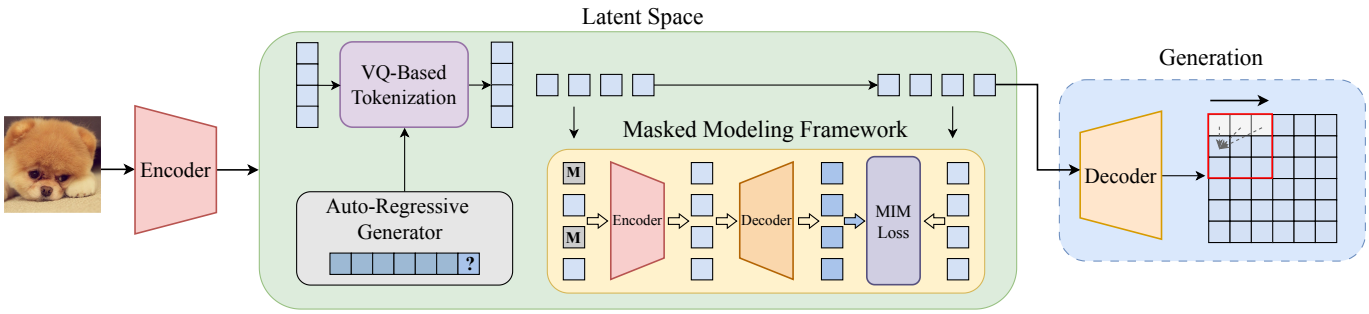


Fig. 11: Research on autoregression (AR) for generation and pre-training can be summarized by this flowchart. Some studies focus on improving the quality and speed of image generation, while others combine pre-training with image generation, performing further operations in the latent space. The figure is reproduced based on [7], [68], [126].

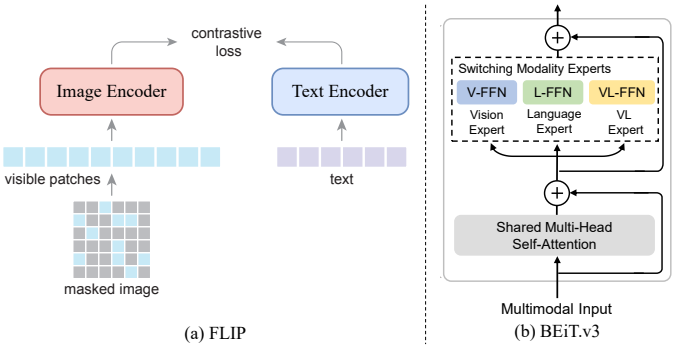


Fig. 12: Illustration of masked modeling with multimodality. (a) FLIP applies masking augmentations to the CLIP [25] framework for text-image alignment. (b) BEiT.v3 [26] designs a mixture-of-export encoder for text-image. The figures are reproduced from [26] and [97].

and text. **BEiT.v3** integrates MOE and multimodality to design specialized tokenizers for vision, language, and vision-language tasks and scales up the model.

Contrastive Methods. **A-CLIP** [101] comprises an online update vision encoder and a language encoder. After images go through extracted feature maps and are masked, they undergo V-L contrastive learning and compute loss with CLIP features. In Figure 12, **FLIP** [97] uses visible image patches and text, which compute a contrastive loss after passing through different encoders. **MaskCLIP** [69] incorporates textual encoding into the masked image modeling architecture and computes contrastive loss between language and images to improve model performance through contrastive learning.

Scaling up. Deep learning models often see substantial performance improvements when the number of model parameters reaches a certain scale. Models based on MAE also exhibit phenomenal changes when their parameter size is expanded to a certain extent. A series of studies have scaled up the MAE parameters and tested their performance in various downstream tasks. Models such as **EVA** [128], **EVA-02** [129], **WSP** [130], and others have achieved excellent results with large parameters. Table 8 summarizes information and performances of this category of models.

4.7.2 *Multimodality for Image Generation*

Another significant research direction in computer vision for multimodal models involves using multimodality for image generation. This encompasses various tasks, including Text-to-Image Generation and Image Generation. The study of image generation primarily falls into two approaches. The first employs an autoregressive method, predominantly based on Vector Quantization, and falls under **VQ-based** algorithms such as **DALLE**] [131]. We have delved further into this in Sec. 4.6.1. The other research category primarily utilizes diffusion with multimodality for image generation. Common models in this category include , **DALLE-2** [132], **DALLE-3** [133], **Stable Diffusion** [134], **GPT-4V** [135], among others.

4.7.3 *Vision Generalist Model*

Vision Generalist Model unifies multiple tasks within a single model, selecting different tasks through prompt input and setting the model's output to a specific target, thereby achieving the unification of various tasks. **Painter** [136] considers an image paired with its corresponding task output, such as text or features, as a sample pair. Such a pair can encompass multiple modalities. The corresponding task output of the image is masked, and then the image, serving as the task's prompt, is fed into the encoder to reconstruct the corresponding task output. Painter captures rich contextual information through this approach and demonstrates impressive performance across various tasks. **InstructDiffusion** [137] and **InstructCV** [138] build upon the foundation of stable diffusion, using prompts and the original image to reconstruct different task objectives, achieving a unification of various task architectures. **LVM** [139] employs a VQ-GAN encoder to transform images into a sequence of tokens, which are then trained using an autoregressive Transformer architecture. The model flexibly generates outputs by constructing partial visual sentences defining specific application tasks. Moreover, the authors propose a large-scale dataset for in-context learning based on LAION-5B, introducing visual sentences as a unified unit of visual data. This approach enables scalable model training from diverse data sources, thus leveraging the vast diversity present in visual data for comprehensive and robust model development.

5 Vision Downstream Task

In this section, we will introduce the specific applications of MIM in Vision downstream tasks. Broadly speaking, we categorize the applications of MIM in vision downstream tasks into four parts: recognition and detection, low-level vision, video representation, and 3D vision tasks. Figure 13 provides a classification of CV downstream tasks.

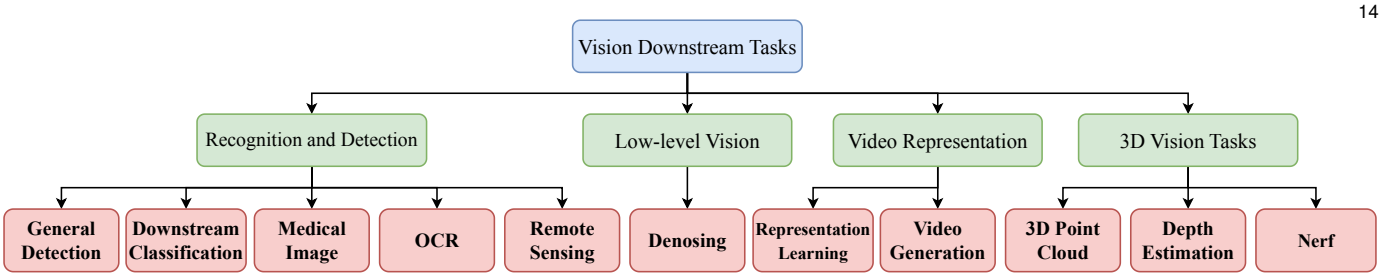


Fig. 13: Illustration of various downstream tasks in computer vision. We summarize them by the label (task) types and data modalities. For example, tasks under **recognition and detection** utilize sample-level (*e.g.*, classification) or sparse objective-level labels (*e.g.*, detection and OCR) on 2D images, while **low-level vision** tasks prefer pixel-level supervision.

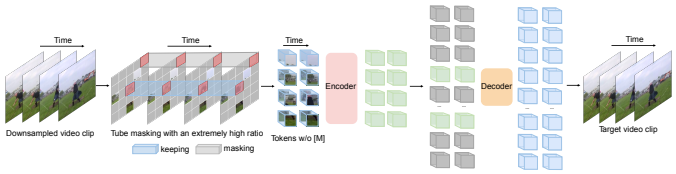


Fig. 14: Illustration of MIM on videos. Taking VideoMAE [140] as an example, it employs an asymmetric encoder-decoder architecture with random spatiotemporal cubic masks and reconstructs the missing ones. The figure is reproduced from [140].

5.1 Video Representation

Research applying Masked Modeling to the video domain primarily focuses on adapting models for high-dimensional video data. This category of research can be divided into two parts: one part is based on the AE architecture, adapting video data to the MAE framework, and the other is based on the AR architecture, using autoregressive methods (such as VQ-VAE, VQ-GAN) to predict video data.

5.1.1 AE-Based Representation Learning

AE-based models usually aim for Video Reconstruction as the task objective to achieve the purpose of Representation Learning. However, videos have higher dimensionality compared to images. Therefore, the focus is on adapting video data to fit within architectures like MAE and BEiT. To apply the 2D MAE framework to videos, a common approach is to mask out space-time tubes instead of spatial patches. This treats the video as a sequence of 2D frames and masks contiguous patches across time. More advanced methods mask at the 3D voxel level for finer spatio-temporal masking. Additional modifications, like introducing a motion-specific encoder, can help capture temporal dynamics.

Based on the framework of **MAE**, **VideoMAE** [140] performs spatial-temporal masking during pre-training by randomly occluding cubic patches in spatiotemporal spaces. Figure 14 shows the framework of VideoMAE. **AdaMAE** [141] adopts an adaptive sampling method that, based on semantic context, utilizes an auxiliary sampling network to sample visible tokens. It estimates a classification distribution concerning spatio-temporal block tokens, selecting tokens that increase the expected reconstruction error as visible tokens. **VideoMAE.v2** [142] introduces a dual-masking strategy where the encoder operates on a subset of video tokens, and the decoder deals with another subset of video tokens. **MotionMAE** [143] reconstructs masked video patches and predicts motion structure, leveraging an asymmetric MAE architecture to outperform existing baselines in action classification and video object segmentation by effectively capturing both static and dynamic information in videos. **OmniMAE** [144] uses masked autoencoding with spatiotemporal patches to train on both images and videos, achieving competitive results in downstream tasks by reconstructing missing patches and applying pixel reconstruction loss. **MAM2** [145] enhances self-supervised video transformer pre-training by separately decoding motion cues using RGB difference as a prediction target, achieving competitive video recognition performance with fewer pre-training epochs.

5.1.2 AR-Based Video Generation

AR-based models typically aim at video prediction or video generation tasks, often employing VQ or GPT architectures to model video data. Given that video information is more redundant and higher-dimensional compared to image information, autoregressive models usually predict sequentially along one dimension at a time. Therefore, it is necessary to convert video data into tokens. In AR-based models, the design of the tokenizer is often crucial. Typically, some methods break videos into 2D patches across space and time to get space-time tokens. More sophisticated tokenizers divide the video into 3D voxels and vector quantize these voxel features to obtain discrete visual tokens.

Different from existing methods applying VQ-encoders on super voxel (3D-VQ), **MGVIT** [126] expand all 2D convolutions inVQ-GAN to 3D convolutions with a temporal axis, and combines 3D-VQ with VQ-GAN to design a new 3D-VQGAN architecture. **MaskViT** [146] employs an MAE-based architecture for video prediction, utilizing spatial and spatiotemporal window attention to enhance memory and training efficiency. **FMNet** [147] predicts the depth of masked frames using adjacent frames, and by reconstructing the masked temporal features, it improves temporal consistency.

5.2 Detection And Recognition

5.2.1 General Detection

iTPN [148] enhances the pre-training phase by incorporating a feature pyramid, unifying the reconstruction and recognition neck, and supplementing MIM with masked feature modeling, providing multi-stage supervision.

MIMdet [149] finds that a MIM pre-trained Vanilla ViT encoder can perform surprisingly well in challenging object-level recognition scenarios, even with randomly sampled partial observations. **imTED** [150] migrates a pre-trained Transformer encoder-decoder to a target detector, constructing a "fully pre-trained" feature extraction pathway to maximize the detector's generalization capability while

introducing a Multi-Scale Feature Modulator to enhance scale adaptability.

5.2.2 Downstream Classification

Face Recognition. In **FaceMAE** [151], randomly masked face images are used to train the reconstruction module. An instance relation matching module is tailored to minimize the distribution gap between real faces and FaceMAE reconstructed ones.

Knowledge Distillation. **G2SD** [152] introduces two knowledge distillation processes to enhance the potential of smaller ViT models. During the generic distillation phase, the smaller model's decoder is encouraged to align its feature predictions with the hidden representations of the larger model, thereby transferring task-agnostic knowledge. In the specific distillation phase, the smaller model's predictions are constrained to be consistent with the larger model's predictions, transferring task-specific features that ensure task performance. **DMAE** [153]introduces a computationally efficient knowledge distillation framework that leverages MAE to align intermediate feature maps between teacher and student models, enabling robust knowledge transfer and improved performance with high masking ratios and limited visible patches.

Efficient Fine-tuning. **Robust Fine-tuing** [154] presents a technique that uses masked image patches for counterfactual sample generation, enhancing model robustness by breaking spurious correlations during fine-tuning of large pre-trained models. **MAE-CT** [155] employs Nearest Neighbor Contrastive Learning to refine the top layers of a pre-trained MAE, enabling it to form semantic clusters and improve performance on classification tasks without the need for labeled data. **MAE-CIL** [156] explores a bilateral MAE framework for Class Incremental Learning, enhancing image reconstruction quality and representation stability through a novel fusion of image-level and embedding-level learning,

5.2.3 Medical Image

SD-MAE [157] performs region masking and reconstruction on histology images to learn useful representations. Additionally, self-distillation is introduced by making the student model mimic the outputs of the teacher autoencoder via a hint loss. **MedMAE** [158] migrates MIM to medical images and appends task-specific Heads for specific tasks. It achieves commendable results in various tasks such as chest X-ray disease classification, abdominal CT multi-organ segmentation, and MRI brain tumor segmentation. **FreMAE** [159] explores the potential of using Fourier Transform for masked image modeling in medical image segmentation, integrating both global structural information and local details. This is achieved by leveraging the frequency domain and multi-stage supervision. **GCMAE** [160] employs MIM for representation learning in the computational pathology domain, effectively extracting both global and local features from pathological images.

5.2.4 OCR

DocMAE [161] proposes a self-supervised framework that leverages masked autoencoders to learn rectification models for document image correction without human annotation. **MaskOCR** [162] presents a novel pre-training approach that uses masked image modeling to learn robust encoder-decoder architectures for text recognition in a self-supervised manner without text annotations.

5.2.5 Remote sensing

Based on **MAE**, **SatMAE** [163] incorporates a temporal embedding and independently masks image patches across time to harness the temporal information present in the data. This approach allows the model to learn from the changes in the data over time, providing a richer and more nuanced understanding of the imagery. **CMID** [164] is capable of learning both global semantic separable and local spatial perceptible representations by combining contrastive learning with MIM in a self-distillation manner. This approach addresses the limitations of existing RS SSL methods, which typically focus on either global or local representations, and is better suited to the varied and complex representations required for different RS downstream tasks.

5.2.6 Low-Level Vision

Deep learning models have achieved state-of-the-art results in various image tasks, but they often struggle to generalize across different noise distributions. **MaskedDenoising** [165] proposes a method that masks random pixels in the input image and reconstructs the missing information during training. Additionally, MaskedDenoising masks feature in the self-attention layer to address inconsistencies between training and testing. The masking training approach introduced by MaskedDenoising enhances the generalization performance of denoising networks. **DreamTeacher** [166] employs two knowledge distillation methods for pre-training image backbones and performing image denoising: feature distillation and label distillation. Feature distillation transfers features from the generative model to the target backbone, while label distillation transfers task-specific labels to the target backbone.

5.3 3D Vision Task

5.3.1 Depth Estimation

Mesa [167] introduces a novel pre-training framework that synergizes masked, geometric, and supervised learning to enhance the representation of later layers in monocular depth estimation models.**UniPAD** [168] introduces a SSL paradigm that utilizes 3D volumetric differentiable rendering for encoding 3D space and reconstructing 3D shapes, significantly enhancing performance in autonomous driving tasks like 3D object detection and semantic segmentation.

5.3.2 3D Point CLoud

Research on 3D point clouds can primarily be divided into three categories: one applies the foundational architecture of MIM to 3D point cloud data, another combines it with contrastive learning, and the last category utilizes different network architectures based on the MIM framework.

Basic MIM. To adapt the 2D MAE framework to 3D point clouds, a common approach is voxelization - converting the irregular point cloud into a regular 3D voxel grid that can then be masked. One method masks contiguous 3D voxels to extend patch masking. Encoder architectures like sparse

3D CNNs help capture 3D spatial context. Alternately, some methods work directly on raw point clouds using specialized encoders. For tokenization, point clouds are often voxelized first before applying 3D convolutional autoencoders to learn discrete voxel tokens. Other approaches cluster point cloud features into visual words without voxelization. Hybrid tokenizers combine both voxel and raw point features. Choosing the right tokenizer is key to learning useful representations.

MAE-Based: **Voxel-MAE** [169] introduces a distance-based random masking strategy and an occupancy prediction pretext task, which helps the model predict the occluded occupancy structure of 3D scenes. **PointMAE** [170] divides the input point cloud into patches, randomly masks them, and uses a Transformer-based autoencoder to learn high-level latent features from unmasked patches. **I2P-MAE** [171] focuses on geometric feature reconstruction and identifies three self-supervised learning objectives specific to point clouds: centroid prediction, normal estimation, and curvature prediction. **ACT** [172] utilizes pre-trained 2D image or language Transformers as teachers for 3D representation learning, transferring their latent features to a 3D Transformer student through masked point modeling. **MaskPoint** [173] introduces a discriminative masked pre-training Transformer framework that represents point clouds as discrete occupancy values and performs binary classification between points of masked objects and sampled noise. **GeoMAE** [174] randomly masks a set of points, employs a Transformer-based point cloud encoder, and then uses a lightweight Transformer decoder to predict the centroid, normals, and curvature for each voxel in the point, enabling the model to infer the fine-grained geometric structure of the point cloud. **BEiT-Based**: **PointBERT** [175] partitions point clouds into local point chunks and employs a point cloud Tokenizer with dVAE to generate discrete tokens. It randomly masks certain chunks of the input point cloud and trains the Transformer to recover the original point tokens at the masked positions, as shown in Figure 15.

Combined with contrastive Learning. **PointCMP** [176] integrates the learning of both local and global spatiotemporal features using a two-branch structure. A mutual similarity-based augmentation module is introduced to generate hard samples at the feature level. The framework achieves state-of-the-art performance on benchmark datasets and demonstrates the superiority of learned representations across different datasets and tasks. **ReCon** [177] combines the merits of both contrastive and generative modeling paradigms through ensemble distillation. It trains a generative student to guide a contrastive student using an encoder-decoder style RECON-block that transfers knowledge through cross attention with stop-gradient. This approach avoids overfitting and pattern difference issues, achieving state-of-the-art results in 3D representation learning and improving performance on downstream tasks.

Different Architecture. **Point-M2AE** [170]: The encoder and decoder are redesigned into a pyramid structure to capture the spatial geometry and semantic information of 3D shapes. Additionally, a multi-scale masking strategy is introduced to generate consistently visible regions across different scales, and skip connections are employed to reconstruct from a global to local-perspective.

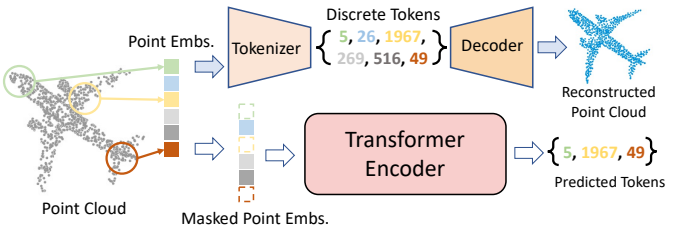


Fig. 15: Illustration of MIM on 3D Point Cloud. Taking PointBERT [175] as an example, PointBERT partitions point clouds into local point chunks and employs a point cloud Tokenizer with dVAE to generate discrete tokens. The figure is reproduced from [175].

6 Masked Modeling on Other Modalities

This section further extends masked modeling pre-training to other mainstream domains beyond CV and NLP and summarizes the essential design and applications.

6.1 Audio and Speech

Combining CL with Masked Modeling. The concept of applying the masked modeling mechanism for SSL can be expanded to audio signals. **VQ-wav2vec** [29] introduces **BERT**-style masked modeling as pre-training on top of **wav2vec** [178]. In **wav2vec**, the input audio signal is first mapped into dense latent representations by an encoder network. Aggregating latent representations from multiple time steps, the context network generates a contextualized representation. A contrastive loss is adopted as the objective function motivated by Contrastive Predictive Coding (CPC) [6]. **VQ-wav2vec** [29] introduces a quantization module to replace the dense latent representations with discrete representations, similar to **VQ-VAE**. The resulting discretized audio representations facilitate a seamless application of the original **BERT**-style masked modeling, which requires a discrete vocabulary. **wav2vec 2.0** adopts a transformer as the context network in contrast to the **wav2vec**, which uses CNNs for both networks. The output from the convolutional encoder is randomly masked before feeding into the transformer. **InfoNCE** is adopted to maximize the similarity between the contextualized representation at the masked time stamps and the corresponding quantized version of the localized representation where negative samples are drawn from other masked time steps. Apart from creating the discrete inputs as input to **BERT** using a quantization module, **Hidden Unit BERT (HuBERT)** [179] discretize the prediction target by coming up with cluster labels provided by applying K-means to Mel Frequency Cepstral Coefficients (MFCC) of the input audio. HuBERT adopts the same architecture design as in **wav2vec 2.0**, where a CNN is adopted as the encoder network and a transformer for the **BERT** encoder. The categorical cross-entropy loss is employed to assess the hidden cluster assignment performance for masked and unmasked tokens, similar to a frame-level acoustic unit discovery problem. It is essential to highlight that while the masking operation is a common element in **VQ-wav2vec**, **wav2vec 2.0**, and **HuBERT**, only **VQ-wav2vec** and **HuBERT** incorporate a **BERT**-style masked modeling approach, whereas **wav2vec 2.0** employs the **BERT**-style

masking operation as a means to enhance the performance of contrastive learning.

Masked Audio Modeling as MIM. In contrast to the common practice in MIM, where the prediction task usually takes the form of regression, regardless of whether the prediction target involves tokenizers, pixels, or features, it is worth noting that **VQ-wav2vec** and **HuBERT**, rigorously adhere to categorization. The pivotal connection uniting MIM and masked audio modeling (MAM) is the transformation from raw audio signals to a visual representation of either spectrogram or mel-spectrogram. Treating the spectrogram as a greyscale image, the problem of MAM can be naturally and directly transformed into the problem of MIM [30], [31], [180], [181], [182], [183]. The difference between these works again resides in the design of the modules for **Mask**, **Target**, **Encoder**, and **Head**. Since the spectrogram itself has already extracted features of the audio signal, the main difference is whether the masked patches are fed into the encoder. Only unmasked patches are fed into the encoder in **Audio-MAE** while works like **Mockingjay** [180] and **Audio ALBERT** [181] pass both masked and unmasked patches into the encoder. **Audio-MAE** [31] explores different masking strategies of unstructured masking (random patch masking), time masking (column-wise masking), and frequency masking (row-wise masking). The framework of Audio-MAE is shown in Figure 16. Combining MAM and MIM, **Audiovisual MAE** [184] proposed that the masked modeling could be simultaneously applied to audio and image for video pre-training.

6.2 Graph

Graph data are in real-world practice, e.g., social networks. Masked modeling has also achieved overwhelming success in graph data analysis. Initially, **AttrMasking** [185] first masks some proportions of nodes and edges within each graph and trains the GNN encoder to predict them. Analogously, **GROVER** [186] attempts to predict the masked subgraphs. Subsequently, **GPT-GNN** [14] proposes an autoregressive framework to perform node and edge reconstruction iteratively, which generates one masked node (atom) and its connected edges (bonds) and optimizes the likelihood of the node and edges generation in the next iteration. More recently, inspired by the huge success of MAE [23] in CV, **GraphMAE** [187] masks some input node features with special tokens and enforces the graph autoencoder to reconstruct the masked ones. **GraphMAE2** [188] argues that GraphMAE is usually vulnerable to disturbance in the features. To mitigate this issue, they designed the multi-view random re-mask decoding and latent representation prediction to regularize the feature reconstruction. Similarly, **MGAE** [189] observes that a high masking ratio of the input graph edges could benefit the downstream tasks. Also, they propose a tailored cross-correlation decoder to reconstruct the large number of masked edges. With the increasing attention paid to graph transformer, **GMAEs** [190] designs an asymmetric graph transformer [191] architecture, where the encoder is a deep transformer and the decoder is a shallow transformer. Equipped with the masking mechanism, GMAE is more memory-efficient than conventional transformers. Despite the fruitful progress, the masking operations create an undesirable dispensary between pre-training and finetuning because the masks would not appear in the downstream tasks. It remains promising to tackle this crucial issue.

6.3 Biology and Chemistry

Masked modeling has recently been extended to various biological applications to accelerate biochemical experiments, especially for research on proteins and molecules.

Sequence Modeling for Protein Considering an amino acid in the protein sequence as a word in the sentence, a number of self-supervised tasks proposed for natural language can be naturally extended to protein sequences. **TAPE** [192] proposes to predict the type of the next amino acid based on a set of masked sequence fragments. **ESM-1b** [193] randomly masks out a single or a set of contiguous amino acids and then predicts the masked amino acids from the remaining sequences. Unlike random masking, **AC-MLM** [194] combines adversarial training with masked language modeling and proposes to mask amino acids in a learnable and adversarial manner. Taking into account the dependence between masked amino acids, **Pairwise MLM (PMLM)** [195] proposes to model the probability of a pair of masked amino acids instead of predicting the probability of a single amino acid. Different from these generative methods, **CPCProt** [196] applies different masking transformations on the input sequences to generate different views and then applies **InfoNCE** to maximize the similarity of two jointly sampled pairs. The antibody is a special kind of protein, and **ABGNN** [197] enables pre-training of antibody sequences by masking the residues on the Compound Determining Regions (CDRs) and predicting the types of masked residues.

Sequence-Structure Co-modeling for Protein The amino acid sequences of proteins can be folded into stable 3D structures in the real physicochemical world, forming a special kind of sequence-structure data. The concept of the masked modeling mechanism for SSL can also be expanded to protein structure pre-training. For example, **GearNet** [198] proposes multiview contrasting that randomly samples two sub-structures from each protein by masking, encoders them into two representations, and finally maximizes the similarity between representations from the same protein while minimizing the similarity between representations from different proteins. **GraphComp** [199] proposes graph completion, which takes as input a protein graph with partially masked residues and then makes predictions for those masked tokens. **AlphaFold2** [32] takes masked language modeling as a pre-training task and full-atomic structure prediction as a downstream task. It was found by [200] that the representations from AlphFold2's **Evoformer** could work well on various protein-related downstream tasks, including fold classification, stability prediction, etc. Moreover, **Masked Inverse Folding (MIF)** [201] trains a model to reconstruct the original amino acids conditioned on the masked sequence and the masked backbone structure. Similar to MAGE in CV, more recently proposed pre-training methods [202], [203] like **FoldSeek** [204] first expand the codebook for amino acid sequences with VQVAE and than perform masked modeling for the latent Transformer encoder.

Graph Representation for Molecules Most molecule data can be represented as SMILE sequences or 2D/3D

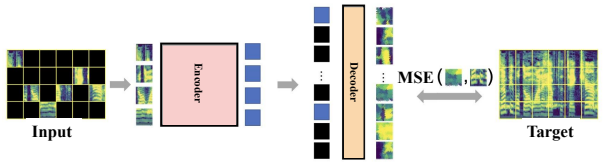


Fig. 16: Illustration of MIM on Audio. Taking Audio-MAE [208] as an example, it applies the MAE framework to audio directly. The figure is reproduced from [208].

graphs. Therefore, many methods developed for languages or graphs can also be directly transferred to molecules. **AttrMasking** [185] randomly masks the input node and edge attributes (e.g., atom type in the molecular graph) and applies GNNs to predict the masked attributes. For sequence-based masking, **SMILES-BERT** [205] and **Molformer** [206] randomly mask the characters in the SMILES sequences and then reconstruct them based on the output of the encoder. To alleviate the problem of imbalance atom types in nature, **Mole-BERT** [207] designs a context-aware tokenizer that encodes atoms as chemically meaningful discrete codes for masking modeling.

7 Discussions and Future Directions

How to design an efficient MIM Model? This paper sets out from its main arguments to offer recommendations and heuristic considerations for designing efficient Masked Image Modeling models. The essence of Masked Modeling lies in the reconstruction using masked data. In NLP, the masked tokens are often several consecutive tokens, an operation grounded in a critical principle: preventing information leakage and enabling the model to work with minimal prior information, thereby increasing the difficulty of the reconstruction task. Therefore, when designing the structure of Masked Modeling, the Masked part should adhere to the principle of preventing information leakage. The **attention-based masking** strategy, while considering the avoidance of data information leakage, utilizes the least computational resources. Furthermore, as introduced in section 3, Masked Modeling's task of reconstructing low-level features and details compensates for the inadequacies of **Transformers** in detail modeling. Coupled with the Transformer's inherent global modeling capabilities, the combination of Masked Modeling and Transformer enables the model to accommodate both low-level modeling capabilities and global modeling abilities, thereby further raising the upper limit of model performance. The selection of Head and Target parts should be contingent upon the specific task at hand. Different Targets will induce varying biases in the model and yield different effects in diverse tasks. Feature maps are generally more suitable for detection tasks. As for whether the Head part should be combined with contrastive learning, this should depend on the choice of Target. If the selected Target necessitates the extraction of a feature map, contrastive learning could be conveniently used to enhance model performance. Conversely, if the model uses Pixels as the Target, employing contrastive learning would not significantly improve performance and would incur substantial computational costs.

Explainability of MIM. Compared to contrastive learning, Masked Modeling still lacks a more comprehensive explanation. The task of contrastive learning, utilizing the InfoNCE loss function, offers a complete loss function and a relatively unified architecture with clearer task objectives. In contrast, Masked Modeling involves complex processing techniques within its various modules and across different modalities. For Masked Modeling, employing different masking strategies and tokenization methods to compress data can result in significant structural and computational differences, making it challenging to develop a comprehensive and unified theoretical explanation. Currently, most theoretical explanations are specific to particular tasks or based on empirical studies, and they fail to generalize across various modalities. The prevailing explanatory approaches mainly unfold in three directions: interpretation based on hierarchical structures, explanations derived from the theoretical foundations of contrastive learning, and interpretations from the perspective of information compression. Although these research efforts provide a certain degree of interpretability to Masked Modeling, they still lack a profound theoretical basis. This makes the interpretability of Masked Modeling a challenging research direction.

Downstream Task Current research on downstream tasks mainly focuses on applying the MAE architecture to specific downstream task structures. However, with the robust growth of Masked Modeling, more complex technologies are gradually being introduced into these tasks. In video research, GPT and MAE are two critical backbones, but a series of studies combining VQ-based models with Masked Modeling are increasingly emerging in the field. These studies employ VQ technology for more efficient data compression and tokenize data to achieve higher-quality reconstruction. Therefore, we believe that research on 3D point clouds will follow this development trend, combining VQ-Based models with Masked Modeling to achieve better information compression efficiency.

Beyond Vision. Multimodal research is currently a significant direction in artificial intelligence, and the application of Masked Modeling in multimodal contexts is one of the most promising future directions. Early multimodal research primarily employed contrastive learning, aligning different modalities and computing contrastive loss. With the advancement of diffusion techniques, studies aligning different modalities through diffusion are also increasing. Masked Modeling holds potential in multimodal applications. The current research paradigm mainly involves aligning different modalities after masking them, increasing task complexity. A new research paradigm is also emerging, where data from different modalities are aligned to a central modality, and then Masked Modeling is applied using the central modality's data. Moreover, applying Masked Modeling to various modalities technically poses more challenges. Extending masking to 3D, 4D, or even higher-dimensional data and tokenizing higher-dimensional data are technical details that need attention and resolution when expanding Masked Modeling to higher dimensions. Therefore, integration with multimodal approaches will be an important research direction for Masked Modeling.

8 Conclusion

This survey, grounded in Computer Vision CV, proposes a unified architecture for Masked Modeling, successfully integrating various technical details and data modalities within this framework. Additionally, we have meticulously organized and elucidated technologies related to Masked Modeling, such as contrastive learning, generative models, and autoregressive models, offering readers a more comprehensive perspective. This paper presents a complete exposition of Masked Modeling's applications and theoretical aspects, detailing its use in various visual tasks as well as Beyond Vision tasks and discussing the current theoretical achievements and progress in Masked Modeling. Based on this, we propose promising future directions for Masked Modeling, aligned with current hot research topics in the artificial intelligence community, such as multimodality and large models, providing readers with ideas for proposing new models and methods based on this article.

ACKNOWLEDGMENTS

This work was supported by the National Key R&D Program of China (No. 2022ZD0115100), the National Natural Science Foundation of China Project (No. U21A20427), and Project (No. WU2022A009) from the Center of Synthetic Biology and Integrated Bioengineering of Westlake University. This work was done by Luyuan Zhang and Zedong Wang during their internship at Westlake University.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778. 1
[2] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *ICCV*, 2017. 1
[3] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A convnet for the 2020s," in *CVPR*, 2022. 1
[4] S. Li, Z. Wang, Z. Liu, C. Tan, H. Lin, D. Wu, Z. Chen, J. Zheng, and S. Z. Li, "Efficient multi-order gated aggregation network," *ArXiv*, 2022. 1
[5] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint*, 2019. 1, 12
[6] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint*, 2018. 1, 4, 12, 16
[7] P. Esser, R. Rombach, and B. Ommer, "Taming transformers for high-resolution image synthesis," 2021. 1, 5, 8, 12, 13
[8] D. Bau, J.-Y. Zhu, H. Strobelt, B. Zhou, J. B. Tenenbaum, W. T. Freeman, and A. Torralba, "Gan dissection: Visualizing and understanding generative adversarial networks," *arXiv*, 2018. 1
[9] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *ArXiv*, 2020. 1
[10] C. Doersch, A. K. Gupta, and A. A. Efros, "Unsupervised visual representation learning by context prediction," *2015 ICCV*, pp. 1422–1430, 2015. 1
[11] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," *ArXiv*, 2016. 1
[12] A. Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, and L. V. Gool, "Repaint: Inpainting using denoising diffusion probabilistic models," *CVPR*, pp. 11 451–11 461, 2022. 1
[13] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint*, 2018. 1, 2
[14] Z. Hu, Y. Dong, K. Wang, K.-W. Chang, and Y. Sun, "Gpt-gnn: Generative pre-training of graph neural networks," in *Proceedings of the 26th SIGKDD*, 2020, pp. 1857–1867. 1, 3, 5, 12, 17
[15] T. Gao, X. Yao, and D. Chen, "Simcse: Simple contrastive learning of sentence embeddings," *ArXiv*, 2021. 1
[16] C.-I. Lai, "Contrastive predictive coding based feature for automatic speaker verification," *ArXiv*, 2019. 1
[17] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance-level discrimination," *ArXiv*, 2018. 1
[18] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *CVPR*, 2020. 1
[19] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," *arXiv preprint*, 2020. 1, 10
[20] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, M. G. Azar *et al.*, "Bootstrap your own latent: A new approach to self-supervised learning," *arXiv preprint*, 2020. 1, 10
[21] M. Chen, A. Radford, J. Wu, H. Jun, P. Dhariwal, D. Luan, and I. Sutskever, "Generative pretraining from pixels," in *ICML*, 2020. 1, 5, 7, 25
[22] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," *ArXiv*, vol. abs/2010.11929, 2020. 2
[23] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *CVPR*, 2022, pp. 16 000–16 009. 2, 5, 6, 7, 17, 25, 27
[24] W. Su, X. Zhu, Y. Cao, B. Li, L. Lu, F. Wei, and J. Dai, "Vl-bert: Pre-training of generic visual-linguistic representations," *ArXiv*, 2019. 2, 12, 25, 28
[25] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *ICML*, 2021. 2, 8, 13
[26] W. Wang, H. Bao, L. Dong, J. Bjorck, Z. Peng, Q. Liu, K. Aggarwal, O. Mohammed, S. Singhal, S. Som, and F. Wei, "Image as a foreign language: Beit pretraining for all vision and vision-language tasks," *ArXiv*, 2022. 2, 7, 8, 13, 25, 28
[27] Y. Zhang, K. Gong, K. Zhang, H. Li, Y. J. Qiao, W. Ouyang, and X. Yue, "Meta-transformer: A unified framework for multimodal learning," *ArXiv*, 2023. 2
[28] Y.-A. Chung and J. R. Glass, "Speech2vec: A sequence-to-sequence framework for learning word embeddings from speech," *ArXiv*, 2018. 2
[29] A. Baevski, H. Zhou, A. rahman Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *ArXiv*, 2020. 2, 16
[30] J. Chen, M. Ma, R. Zheng, and L. Huang, "Mam: Masked acoustic modeling for end-to-end speech-to-text translation," *ArXiv*, 2020. 2, 17
[31] P.-Y. Huang, H. Xu, J. Li, A. Baevski, M. Auli, W. Galuba, F. Metze, and C. Feichtenhofer, "Masked autoencoders that listen," *NeurIPS*, vol. 35, pp. 28 708–28 720, 2022. 2, 17
[32] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko *et al.*, "Highly accurate protein structure prediction with alphafold," *Nature*, vol. 596, no. 7873, pp. 583–589, 2021. 2, 17
[33] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, and J. Tang, "Self-supervised learning: Generative or contrastive," *TKDE*, vol. 35, no. 1, pp. 857–876, 2021. 3, 4
[34] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *NeurIPS*, 2017, pp. 5998–6008. 3
[35] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," *arXiv preprint*, 2019. 4
[36] Z. Xie, Z. Zhang, Y. Cao, Y. Lin, J. Bao, Z. Yao, Q. Dai, and H. Hu, "Simmim: a simple framework for masked image modeling," *CVPR*, 2021. 7, 10, 25
[37] L. Wang, F. Liang, Y. Li, H. Zhang, W. Ouyang, and J. Shao, "Repre: Improving self-supervised vision transformer with reconstructive pre-training," in *IJCAI*, 2022. 7, 25
[38] Q. Wu, H. Ye, Y. Gu, H. Zhang, L. Wang, and D. He, "Denoising masked autoencoders are certifiable robust vision learners," *ArXiv*, 2022. 7, 25
[39] Y. Lee, J. Willette, J. Kim, J. Lee, and S. J. Hwang, "Exploring the role of mean teachers in self-supervised masked auto-encoders," *ArXiv*, 2022. 7, 9, 25

[40] D.-K. Nguyen, V. Aggarwal, Y. Li, M. R. Oswald, A. Kirillov, C. G. M. Snoek, and X. Chen, "R-mae: Regions meet masked autoencoders," *ArXiv*, 2023. 7, 26

[41] C. K. Ryali, Y.-T. Hu, D. Bolya, C. Wei, H. Fan, P.-Y. Huang, V. Aggarwal, A. Chowdhury, O. Poursaeed, J. Hoffman, J. Malik, Y. Li, and C. Feichtenhofer, "Hiera: A hierarchical vision transformer without the bells-and-whistles," in *ICML*, 2023. 7, 9, 25

[42] X. Dong, J. Bao, T. Zhang, D. Chen, W. Zhang, L. Yuan, D. Chen, F. Wen, and N. Yu, "Bootstrapped masked autoencoders for vision bert pretraining," in *ECCV*, 2022. 7, 8, 25

[43] Y. Chen, Y. Liu, D. Jiang, X. Zhang, W. Dai, H. Xiong, and Q. Tian, "Sdae: Self-distillated masked autoencoder," in *ECCV*, 2022. 7, 9, 25

[44] Y. Gandelsman, Y. Sun, X. Chen, and A. A. Efros, "Test-time training with masked autoencoders," *ArXiv*, 2022. 7, 25

[45] G. Kwon, Z. Cai, A. Ravichandran, E. Bas, R. Bhotika, and S. . Soatto, "Masked vision and language modeling for multi-modal representation learning," in *ICLR*, 2023. 7, 12, 25, 28

[46] S. Wang, J. Gao, Z. Li, J. Sun, and W. Hu, "A closer look at self-supervised lightweight vision transformers," *ArXiv*, 2022. 7, 25

[47] X. Chen, M. Ding, X. Wang, Y. Xin, S. Mo, Y. Wang, S. Han, P. Luo, G. Zeng, and J. Wang, "Context autoencoder for self-supervised representation learning," *ArXiv*, 2022. 7, 25

[48] C. Tao, X. Zhu, G. Huang, Y. Qiao, X. Wang, and J. Dai, "Siamese image modeling for self-supervised vision representation learning," *CVPR*, pp. 2132–2141, 2022. 7, 25

[49] X. Liu, J. Zhou, T. Kong, X. Lin, and R. Ji, "Exploring target representations for masked autoencoders," *arXiv preprint*, 2022. 7, 8, 25

[50] Z. Peng, L. Dong, H. Bao, Q. Ye, and F. Wei, "A unified view of masked image modeling," 2022. 7, 9, 25

[51] X. Zhang, J. Chen, J. Yuan, Q. Chen, J. Wang, X. Wang, S. Han, X. Chen, J. Pi, K. Yao, J. Han, E. Ding, and J. Wang, "Cae v2: Context autoencoder with clip target," *ArXiv*, 2022. 7, 9, 25

[52] J. Guo, K. Han, H. Wu, Y. Tang, Y. Wang, and C. Xu, "Fastmim: Expediting masked image modeling pre-training for vision," 2022. 7, 8, 25

[53] A. Baevski, W.-N. Hsu, Q. Xu, A. Babu, J. Gu, and M. Auli, "data2vec: A general framework for self-supervised learning in speech, vision and language," in *ICML*, 2022. 7, 8, 10, 25

[54] J. Xie, W. Li, X. Zhan, Z. Liu, Y. S. Ong, and C. C. Loy, "Masked frequency modeling for self-supervised visual pre-training," *ArXiv*, 2022. 7, 8, 25

[55] S. Casas, A. Sadat, and R. Urtasun, "Mp3: A unified model to map, perceive, predict and plan," *CVPR*, pp. 14 398–14 407, 2021. 7, 25

[56] C. Wei, H. Fan, S. Xie, C. Wu, A. L. Yuille, and C. Feichtenhofer, "Masked feature prediction for self-supervised visual pre-training," *CVPR*, pp. 14 648–14 658, 2021. 7, 8, 25

[57] R. Bachmann, D. Mizrahi, A. Atanov, and A. R. Zamir, "Multimae: Multi-modal multi-task masked autoencoders," *ArXiv*, 2022. 7, 25

[58] J. Zhou, C. Wei, H. Wang, W. Shen, C. Xie, A. Yuille, and T. Kong, "ibot: Image bert pre-training with online tokenizer," *ICLR*, 2022. 7, 8, 25

[59] H. Bao, L. Dong, and F. Wei, "Beit: Bert pre-training of image transformers," in *ICLR*, 2022. 7, 8, 10, 25

[60] Z. Peng, L. Dong, H. Bao, Q. Ye, and F. Wei, "Beit v2: Masked image modeling with vector-quantized visual tokenizers," *ArXiv*, 2022. 7, 8, 10, 25

[61] L. Baraldi, R. Amoroso, M. Cornia, A. Pilzer, and R. Cucchiara, "Learning to mask and permute visual tokens for vision transformer pre-training," *ArXiv*, 2023. 7, 26

[62] T. Hua, Y. Tian, S. Ren, H. Zhao, and L. Sigal, "Self-supervision through random segments with autoregressive coding (randsac)," *ArXiv*, 2022. 7, 12, 25

[63] H. Chang, H. Zhang, L. Jiang, C. Liu, and W. T. Freeman, "Maskgit: Masked generative image transformer," *CVPR*, pp. 11 305–11 315, 2022. 7, 12, 25, 28

[64] X. Zheng, X. Ma, and C. Wang, "Cim: Constrained intrinsic motivation for sparse-reward continuous control," *ArXiv*, 2022. 7, 25

[65] X. Li, Y. Ge, K. Yi, Z. Hu, Y. Shan, and L. yu Duan, "mc-beit: Multi-choice discretization for image bert pre-training," in *ECCV*, 2022. 7, 8, 25

[66] L. Wei, L. Xie, W. gang Zhou, H. Li, and Q. Tian, "Mvp: Multimodality-guided visual pre-training," *ArXiv*, 2022. 7, 25

[67] X. Dong, J. Bao, T. Zhang, D. Chen, W. Zhang, L. Yuan, D. Chen, F. Wen, and N. Yu, "Peco: Perceptual codebook for bert pre-training of vision transformers," in *AAAI*, 2021. 7, 8, 25

[68] T. Li, H. Chang, S. K. Mishra, H. Zhang, D. Katabi, and D. Krishnan, "Mage: Masked generative encoder to unify representation learning and image synthesis," *arXiv preprint*, 2022. 7, 12, 13, 25

[69] X. Dong, Y. Zheng, J. Bao, T. Zhang, D. Chen, H. Yang, M. Zeng, W. Zhang, L. Yuan, D. Chen, F. Wen, and N. Yu, "Maskclip: Masked self-distillation advances contrastive language-image pretraining," *ArXiv*, 2022. 7, 9, 13, 25, 28

[70] H. Liu, X. Jiang, X. Li, A. Guo, D. Jiang, and B. Ren, "The devil is in the frequency: Geminated gestalt autoencoder for self-supervised visual pre-training," in *AAAI*, 2022. 7, 8, 25

[71] K. Yi, Y. Ge, X. Li, S. Yang, D. Li, J. Wu, Y. Shan, and X. Qie, "Masked image modeling with denoising contrast," *ArXiv*, 2022. 7, 10, 25

[72] Z. Jiang, Y. Chen, M. Liu, D. Chen, X. Dai, L. Yuan, Z. Liu, and Z. Wang, "Layer grafted pre-training: Bridging contrastive learning and masked image modeling for label-efficient representations," in *ICLR*, 2023. 7, 11, 25

[73] J. ju Mao, H. Zhou, X. Yin, Y. Chang, B. Nie, and R. Xu, "Masked autoencoders are effective solution to transformer data-hungry," *ArXiv*, 2022. 7, 25, 26

[74] Z. Li, Z. Chen, F. Yang, W. Li, Y. Zhu, C. Zhao, R. Deng, L. Wu, R. Zhao, M. Tang, and J. Wang, "Mst: Masked self-supervised transformer for visual representation," in *NeurIPS*, 2021. 7, 25

[75] Y. Shi, N. Siddharth, P. Torr, and A. R. Kosiorek, "Adversarial masking for self-supervised learning," in *ICML*. PMLR, 2022, pp. 20 026–20 040. 7, 25

[76] X. Li, W. Wang, L. Yang, and J. Yang, "Uniform masking: Enabling mae pre-training for pyramid-based vision transformers with locality," *ArXiv*, 2022. 7, 25

[77] G. Li, H. Zheng, D. Liu, B. Su, and C. Zheng, "Semmae: Semantic-guided masking for learning masked autoencoders," *ArXiv*, 2022. 6, 7, 25

[78] J. Chen, M. Hu, B. Li, and M. Elhoseiny, "Efficient self-supervised vision pretraining with local masked reconstruction," *arXiv preprint*, 2022. 7, 25

[79] K. Zhang and Z. Shen, "i-mae: Are latent representations in masked autoencoders linearly separable?" *ArXiv*, 2022. 7, 25

[80] S. Zhang, F. Zhu, R. Zhao, and J. Yan, "Contextual image masking modeling via synergized contrasting without view augmentation for faster and better visual pretraining," in *ICLR*, 2023. 7, 10, 25

[81] H. Chen, W. Zhang, Y. Wang, and X. Yang, "Improving masked autoencoders by learning where to mask," *ArXiv*, 2023. 7, 25

[82] H. Wang, K. Song, J. Fan, Y. Wang, J. Xie, and Z. Zhang, "Hard patches mining for masked image modeling," in *CVPR*, 2023. 6, 7, 25

[83] M. Assran, Q. Duval, I. Misra, P. Bojanowski, P. Vincent, M. Rabbat, Y. LeCun, and N. Ballas, "Self-supervised learning from images with a joint-embedding predictive architecture," in *CVPR*, 2023, pp. 15 619–15 629. 7, 25

[84] J. Liu, X. Huang, Y. Liu, and H. Li, "Mixmim: Mixed and masked image modeling for efficient visual representation learning," *ArXiv*, 2022. 7, 25

[85] J. Wu and S. Mo, "Object-wise masked autoencoders for fast pre-training," *ArXiv*, 2022. 6, 7, 25

[86] I. Kakogeorgiou, S. Gidaris, B. Psomas, Y. Avrithis, A. Bursuc, K. Karantzalos, and N. Komodakis, "What to hide from your students: Attention-guided masked image modeling," in *ECCV*, 2022. 6, 7, 25

[87] Z. Hou, F. Sun, Y.-K. Chen, Y. Xie, and S. Y. Kung, "Milan: Masked image pretraining on language assisted representation," *ArXiv*, 2022. 6, 7, 8, 25

[88] X. Ma, C.-S. Liu, C. Xie, L. Ye, Y. Deng, and X. Ji, "Disjoint masking with joint distillation for efficient masked image modeling," *ArXiv*, 2022. 7, 9, 26

[89] A. Baevski, A. Babu, W.-N. Hsu, and M. Auli, "Efficient self-supervised learning with contextualized target representations for vision, speech and language," 2022. 7, 8, 25

[90] S. Woo, S. Debnath, R. Hu, X. Chen, Z. Liu, I.-S. Kweon, and S. Xie, "Convnext v2: Co-designing and scaling convnets with masked autoencoders," *ArXiv*, 2023. 7, 9, 25

[91] K. Tian, Y. Jiang, Q. Diao, C. Lin, L. Wang, and Z. Yuan, "Designing bert for convolutional networks: Sparse and hierarchical masked modeling," *ArXiv*, 2023. 7, 9, 25

[92] P. Gao, T. Ma, H. Li, J. Dai, and Y. J. Qiao, "Convmae: Masked convolution meets masked autoencoders," *ArXiv*, 2022. 7, 9, 25

[93] S. K. Mishra, J. Robinson, H. Chang, D. Jacobs, A. Sarna, A. Maschinot, and D. Krishnan, "A simple, efficient and scalable contrastive masked autoencoder for learning visual representations," *ArXiv*, 2022. 7, 10, 25

[94] M. Assran, M. Caron, I. Misra, P. Bojanowski, F. Bordes, P. Vincent, A. Joulin, M. G. Rabbat, and N. Ballas, "Masked siamese networks for label-efficient learning," in *ECCV*, 2022. 7, 10, 25

[95] Z. Wu, Z. Lai, X. Sun, and S. Lin, "Extreme masking for learning instance and distributed visual representations," *ArXiv*, 2022. 7, 25

[96] Q. feng Zhou, C. Yu, H. Luo, Z. Wang, and H. Li, "Mimco: Masked image modeling pre-training with contrastive teacher," in *MM*, 2022. 7, 10, 25

[97] Y. Li, H. Fan, R. Hu, C. Feichtenhofer, and K. He, "Scaling language-image pre-training via masking," *ArXiv*, 2022. 7, 13, 25, 28

[98] Y. Yao, N. Desai, and M. S. Palaniswami, "Moma: Distill from self-supervised teachers," *ArXiv*, 2023. 7, 8, 26

[99] S. Ren, Z. Wang, H. Zhu, J. Xiao, A. Yuille, and C. Xie, "Rejuvenating image-gpt as strong visual representation learners," 2023. 7, 26

[100] Z. Huang, X. Jin, C. Lu, Q. Hou, M.-M. Cheng, D. Fu, X. Shen, and J. Feng, "Contrastive masked autoencoders are stronger vision learners," *ArXiv*, 2022. 7, 25

[101] Y. Yang, W. Huang, Y. Wei, H. Peng, X. Jiang, H. Jiang, F. Wei, Y. Wang, H. Hu, L. Qiu, and Y. Yang, "Attentive mask clip," 2022. 7, 13, 25, 28

[102] K. Chen, Z. Liu, L. Hong, H. Xu, Z. Li, and D.-Y. Yeung, "Mixed autoencoder for self-supervised visual representation learning," *ArXiv*, 2023. 6

[103] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," *ArXiv*, 2017. 7

[104] Y. Fang, L. Dong, H. Bao, X. Wang, and F. Wei, "Corrupted image modeling for self-supervised visual pre-training," *arXiv preprint*, 2022. 8, 9

[105] S. Sameni, S. Jenni, and P. Favaro, "Representation learning by detecting incorrect location embeddings," *AAAI*, 2022. 8, 25

[106] S. Zhai, N. Jaitly, J. Ramapuram, D. Busbridge, T. Likhomanenko, J. Y. Cheng, W. A. Talbott, C. Huang, H. Goh, and J. M. Susskind, "Position prediction as an effective pretraining strategy," in *ICML*, 2022. 8

[107] H. Xu, S. Ding, X. Zhang, H. Xiong, and Q. Tian, "Masked autoencoders are robust data augmentors," *ArXiv*, 2022. 8

[108] H. Wang, J. Fan, Y. Wang, K. Song, T. Wang, and Z. Zhang, "Droppos: Pre-training vision transformers by reconstructing dropped positions," *ArXiv*, 2023. 8, 26

[109] S. Li, D. Wu, F. Wu, Z. Zang, K. Wang, L. Shang, B. Sun, H. Li, and Stan.Z.Li, "Architecture-agnostic masked image modeling - from vit back to cnn," in *ICML*, 2023. 8, 9, 25

[110] L. Jing, J. Zhu, and Y. LeCun, "Masked siamese convnets," *ArXiv*, 2022. 8, 10, 26

[111] Y. Liu, S. Zhang, J. Chen, K. Chen, and D. Lin, "Pixmim: Rethinking pixel reconstruction in masked image modeling," *ArXiv*, 2023. 8, 26

[112] H. Pan, C. Liu, W. Wang, L. Yuan, H. Wang, Z. Li, and W. Liu, "Img2vec: A teacher of high token-diversity helps masked autoencoders," 2023. 8, 26

[113] S. Ren, F. Wei, Z. Zhang, and H. Hu, "Tinymim: An empirical study of distilling mim pre-trained models," 2023. 8, 26

[114] L. Huang, S. You, M. Zheng, F. Wang, C. Qian, and T. Yamasaki, "Green hierarchical vision transformer for masked image modeling," *ArXiv*, 2022. 9, 25

[115] X. Zhang, Y. Tian, W. Huang, Q. Ye, Q. Dai, L. Xie, and Q. Tian, "Hivit: Hierarchical vision transformer meets masked image modeling," *ArXiv*, 2022. 9, 25

[116] A. Zhou, Y. Li, Z. Qin, J. Liu, J. Pan, R. Zhang, R. Zhao, P. Gao, and H. Li, "Sparsemae: Sparse training meets masked autoencoders," in *ICCV*, 2023, pp. 16 176–16 186. 9, 25

[117] H. Wang, Y. Tang, Y. Wang, J. Guo, Z. Deng, and K. Han, "Masked image modeling with local multi-scale reconstruction," *CVPR*, pp. 2122–2131, 2023. 10, 25

[118] J. Li, P. Zhou, C. Xiong, R. Socher, and S. C. H. Hoi, "Prototypical contrastive learning of unsupervised representations," *ArXiv*, 2020. 10

[119] T. Yu, S. Kumar, A. Gupta, S. Levine, K. Hausman, and C. Finn, "Gradient surgery for multi-task learning," *ArXiv*, 2020. 11

[120] Q. Zhang, Y. Wang, and Y. Wang, "How mask matters: Towards theoretical understandings of masked autoencoders," *ArXiv*, 2022. 11

[121] L. Kong, M. Q. Ma, G. Chen, E. P. Xing, Y. Chi, L.-P. Morency, and K. Zhang, "Understanding masked autoencoders via hierarchical latent variable models," in *CVPR*, 2023, pp. 7918–7928. 11

[122] Z. Xie, Z. Geng, J. Hu, Z. Zhang, H. Hu, and Y. Cao, "Revealing the dark secrets of masked image modeling," *ArXiv*, 2022. 11

[123] G. K. Kumar, S. S. Mullappilly, and A. S. Gehlot, "An empirical study of self-supervised learning approaches for object detection with transformers," *ArXiv*, 2022. 11

[124] Z. Xie, Z. Zhang, Y. Cao, Y. Lin, Y. Wei, Q. Dai, and H. Hu, "On data scaling in masked image modeling," *ArXiv*, 2022. 11

[125] X. Kong and X. Zhang, "Understanding masked image modeling via learning occlusion invariant feature," *ArXiv*, 2022. 11

[126] L. Yu, Y. Cheng, K. Sohn, J. Lezama, H. Zhang, H. Chang, A. G. Hauptmann, M.-H. Yang, Y. Hao, I. Essa, and L. Jiang, "Magvit: Masked generative video transformer," in *CVPR*, 2023. 13, 14, 26

[127] T. Li, D. Katabi, and K. He, "Self-conditioned image generation via generating representations," 2023. 12

[128] Y. Fang, W. Wang, B. Xie, Q.-S. Sun, L. Y. Wu, X. Wang, T. Huang, X. Wang, and Y. Cao, "Eva: Exploring the limits of masked visual representation learning at scale," *ArXiv*, 2022. 13, 27

[129] Y. Fang, Q. Sun, X. Wang, T. Huang, X. Wang, and Y. Cao, "Eva-02: A visual representation for neon genesis," *ArXiv*, 2023. 13, 27

[130] M. Singh, Q. Duval, K. V. Alwala, H. Fan, V. Aggarwal, A. B. Adcock, A. Joulin, P. Doll'ar, C. Feichtenhofer, R. B. Girshick, R. Girdhar, and I. Misra, "The effectiveness of mae pre-pretraining for billion-scale pretraining," *ArXiv*, 2023. 13, 27

[131] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever, "Zero-shot text-to-image generation," *ArXiv*, 2021. 13, 28

[132] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *ArXiv*, 2022. 13

[133] J. Betker, G. Goh, L. Jing, TimBrooks, J. Wang, L. Li, LongOuyang, JuntangZhuang, JoyceLee, YufeiGuo, WesamManassra, PrafullaDhariwal, CaseyChu, YunxinJiao, and A. Ramesh, "Improving image generation with better captions." 13

[134] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," *CVPR*, pp. 10 674–10 685, 2021. 13

[135] L. Wen, X. Yang, D. Fu, X. Wang, P. Cai, X. Li, T. Ma, Y. Li, L. Xu, D. Shang, Z. Zhu, S. Sun, Y. Bai, X. Cai, M. Dou, S. Hu, B. Shi, and Y. Qiao, "On the road with gpt-4v(ision): Early explorations of visual-language model on autonomous driving," 2023. 13

[136] X. Wang, W. Wang, Y. Cao, C. Shen, and T. Huang, "Images speak in images: A generalist painter for in-context visual learning," *CVPR*, pp. 6830–6839, 2022. 13, 27

[137] Z. Geng, B. Yang, T. Hang, C. Li, S. Gu, T. Zhang, J. Bao, Z. Zhang, H. Hu, D. Chen, and B. Guo, "Instructdiffusion: A generalist modeling interface for vision tasks," *ArXiv*, 2023. 13

[138] Y. Gan, S. Park, A. Schubert, A. Philippakis, and A. M. Alaa, "Instructcv: Instruction-tuned text-to-image diffusion models as vision generalists," *ArXiv*, 2023. 13, 28

[139] Y. Bai, X. Geng, K. Mangalam, A. Bar, A. Yuille, T. Darrell, J. Malik, and A. A. Efros, "Sequential modeling enables scalable learning for large vision models," 2023. 13, 26, 27

[140] Z. Tong, Y. Song, J. Wang, and L. Wang, "Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training," *ArXiv*, 2022. 14, 26

[141] W. G. C. Bandara, N. Patel, A. Gholami, M. Nikkhah, M. Agrawal, and V. M. Patel, "Adamae: Adaptive masking for efficient spatiotemporal learning with masked autoencoders," in *CVPR*, 2023, pp. 14 507–14 517. 14, 26

[142] L. Wang, B. Huang, Z. Zhao, Z. Tong, Y. He, Y. Wang, Y. Wang, and Y. Qiao, "Videomae v2: Scaling video masked autoencoders with dual masking," in *CVPR*, 2023. 14, 26

[143] H. Yang, D. Huang, B. Wen, J. Wu, H. Yao, Y. Jiang, X. Zhu, and Z. Yuan, "Self-supervised video representation learning with motion-aware masked autoencoders," 2022. 14, 26

[144] R. Girdhar, A. El-Nouby, M. Singh, K. V. Alwala, A. Joulin, and I. Misra, "Omnimae: Single model masked pretraining on images and videos," *ArXiv*, 2022. 14, 26

[145] Y. Song, M. Yang, W. Wu, D. He, F. Li, and J. Wang, "It takes two: Masked appearance-motion modeling for self-supervised video transformer pre-training," *ArXiv*, 2022. 14, 26

[146] A. Gupta, S. Tian, Y. Zhang, J. Wu, R. Mart'in-Mart'in, and L. Fei-Fei, "Maskvit: Masked visual pre-training for video prediction," *ArXiv*, 2022. 14, 26

[147] Y. Wang, Z. Pan, X. Li, Z. CAO, K. Xian, and J. Zhang, "Less is more: Consistent video depth estimation with masked frames modeling," *ArXiv*, 2022. 14, 26

[148] Y. Tian, L. Xie, Z. Wang, L. Wei, X. Zhang, J. Jiao, Y. Wang, Q. Tian, and Q. Ye, "Integrally pre-trained transformer pyramid networks," *CVPR*, pp. 18 610–18 620, 2022. 14, 26

[149] Y. Fang, S. Yang, S. Wang, Y. Ge, Y. Shan, and X. Wang, "Unleashing vanilla vision transformer with masked image modeling for object detection," *ArXiv*, 2022. 14, 26

[150] X. Zhang, F. Liu, Z. Peng, Z. Guo, F. Wan, X.-W. Ji, and Q. Ye, "Integrally migrating pre-trained transformer encoder-decoders for visual object detection," 2022. 14, 26

[151] K. Wang, B. Zhao, X. Peng, Z. H. Zhu, J. Deng, X. Wang, H. Bilen, and Y. You, "Facemae: Privacy-preserving face recognition via masked autoencoders," *ArXiv*, 2022. 15

[152] W. Huang, Z. Peng, L. Dong, F. Wei, J. Jiao, and Q. Ye, "Generic-to-specific distillation of masked autoencoders," *ArXiv*, 2023. 15, 26

[153] Y. Bai, Z. Wang, J. Xiao, C. Wei, H. Wang, A. L. Yuille, Y. Zhou, and C. Xie, "Masked autoencoders enable efficient knowledge distillers," *CVPR*, pp. 24 256–24 265, 2022. 15

[154] Y. Xiao, Z. Tang, P. Wei, C. Liu, and L. Lin, "Masked images are counterfactual samples for robust fine-tuning," *CVPR*, pp. 20 301–20 310, 2023. 15

[155] J. Lehner, B. Alkin, A. Fürst, E. Rumetshofer, L. Miklautz, and S. Hochreiter, "Contrastive tuning: A little help to make masked autoencoders forget," *ArXiv*, 2023. 15

[156] J.-T. Zhai, X. Liu, A. D. Bagdanov, K.-C. Li, and M.-M. Cheng, "Masked autoencoders are efficient class incremental learners," *ArXiv*, 2023. 15

[157] Y. Luo, Z. Chen, and X. Gao, "Self-distillation augmented masked autoencoders for histopathological image classification," *ArXiv*, 2022. 15

[158] L. Zhou, H. Liu, J. Bae, J. He, D. Samaras, and P. Prasanna, "Self pre-training with masked autoencoders for medical image analysis," *ArXiv*, 2022. 15, 26

[159] W. Wang, J. Wang, C. Chen, J. Jiao, L. Sun, Y. Cai, S. Song, and J. Li, "Fremae: Fourier transform meets masked autoencoders for medical image segmentation," 2023. 15, 26

[160] H. Quan, X. Li, W. Chen, Q. Bai, M. Zou, R. Yang, T. Zheng, R. Qi, X. Gao, and X. Cui, "Global contrast masked autoencoders are powerful pathological representation learners," *arXiv*, 2022. 15, 26

[161] S. Liu, H. Feng, W. gang Zhou, H. Li, C. Liu, and F. Wu, "Docmae: Document image rectification via self-supervised representation learning," 2023. 15, 26

[162] P. Lyu, C. Zhang, S. Liu, M. Qiao, Y. Xu, L. Wu, K. Yao, J. Han, E. Ding, and J. Wang, "Maskocr: Text recognition with masked encoder-decoder pretraining," *ArXiv*, 2022. 15

[163] Y. Cong, S. Khanna, C. Meng, P. Liu, E. Rozi, Y. He, M. Burke, D. Lobell, and S. Ermon, "Satmae: Pre-training transformers for temporal and multi-spectral satellite imagery," *ArXiv*, 2022. 15, 26

[164] D. Muhtar, X. liang Zhang, P. Xiao, Z. Li, and F. Gu, "Cmid: A unified self-supervised learning framework for remote sensing image understanding," *TGRS*, 2023. 15, 26

[165] H. Chen, J. Gu, Y. Liu, S. A. Magid, C. Dong, Q. Wang, H. Pfister, and L. Zhu, "Masked image training for generalizable deep image denoising," *CVPR*, pp. 1692–1703, 2023. 15

[166] D. Li, H. Ling, A. Kar, D. Acuna, S. W. Kim, K. Kreis, A. Torralba, and S. Fidler, "Dreamteacher: Pretraining image backbones with deep generative models," *ArXiv*, 2023. 15

[167] M. O. Khan, J. Liang, C.-K. Wang, S. Yang, and Y. Lou, "Mesa: Masked, geometric, and supervised pre-training for monocular depth estimation," *ArXiv*, 2023. 15

[168] H. Yang, S. Zhang, D. Huang, X. Wu, H. Zhu, T. He, S. Tang, H. Zhao, Q. Qiu, B. Lin, X. He, and W. Ouyang, "Unipad: A universal pre-training paradigm for autonomous driving," *ArXiv*, 2023. 15

[169] C. Min, X. Xu, D. Zhao, L. Xiao, Y. Nie, and B. Dai, "Voxel-mae: Masked autoencoders for pre-training large-scale point clouds," *ArXiv*, 2022. 16, 26

[170] R. Zhang, Z. Guo, P. Gao, R. Fang, B. Zhao, D. Wang, Y. J. Qiao, and H. Li, "Point-m2ae: Multi-scale masked autoencoders for hierarchical point cloud pre-training," *ArXiv*, 2022. 16, 26

[171] R. Zhang, L. Wang, Y. J. Qiao, P. Gao, and H. Li, "Learning 3d representations from 2d pre-trained models via image-to-point masked autoencoders," in *CVPR*, 2023. 16, 26

[172] R. Dong, Z. Qi, L. Zhang, J. Zhang, J. Sun, Z. Ge, L. Yi, and K. Ma, "Autoencoders as cross-modal teachers: Can pretrained 2d image transformers help 3d representation learning?" in *ICLR*, 2023. 16, 26

[173] B. Liu, D. Hsu, P. Ravikumar, and A. Risteski, "Masked prediction tasks: a parameter identifiability view," *ArXiv*, 2022. 16

[174] X. Tian, H. Ran, Y. Wang, and H. Zhao, "Geomae: Masked geometric target prediction for self-supervised point cloud pre-training," in *CVPR*, 2023, pp. 13 570–13 580. 16, 26

[175] X. Yu, L. Tang, Y. Rao, T. Huang, J. Zhou, and J. Lu, "Point-bert: Pre-training 3d point cloud transformers with masked point modeling," in *CVPR*, 2022. 16, 26

[176] Z. Shen, X. Sheng, L. Wang, Y. K. Guo, Q. Liu, and X. Zhou, "Pointcmp: Contrastive mask prediction for self-supervised learning on point cloud videos," in *CVPR*, 2023. 16, 26

[177] Z. Qi, R. Dong, G. Fan, Z. Ge, X. Zhang, K. Ma, and L. Yi, "Contrast with reconstruct: Contrastive 3d representation learning guided by generative pretraining," *ArXiv*, 2023. 16, 26

[178] A. Baevski, S. Schneider, and M. Auli, "vq-wav2vec: Self-supervised learning of discrete speech representations," *ArXiv*, 2019. 16

[179] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *TASLP*, vol. 29, pp. 3451–3460, 2021. 16

[180] A. T. Liu, S.-w. Yang, P.-H. Chi, P.-c. Hsu, and H.-y. Lee, "Mockingjay: Unsupervised speech representation learning with deep bidirectional transformer encoders," in *ICASSP*. IEEE, 2020, pp. 6419–6423. 17

[181] P.-H. Chi, P.-H. Chung, T.-H. Wu, C.-C. Hsieh, Y.-H. Chen, S.-W. Li, and H.-y. Lee, "Audio albert: A lite bert for self-supervised learning of audio representation," in *SLT*. IEEE, 2021, pp. 344–350. 17

[182] A. Baade, P. Peng, and D. F. Harwath, "Mae-ast: Masked autoencoding audio spectrogram transformer," *ArXiv*, 2022. 17

[183] D. Chong, H. Wang, P. Zhou, and Q. jie Zeng, "Masked spectrogram prediction for self-supervised audio pre-training," *ArXiv*, 2022. 17

[184] M.-I. Georgescu, E. Fonseca, R. T. Ionescu, M. Lucic, C. Schmid, and A. Arnab, "Audiovisual masked autoencoders," in *ICCV*, 2023, pp. 16 144–16 154. 17

[185] W. Hu, B. Liu, J. Gomes, M. Zitnik, P. Liang, V. Pande, and J. Leskovec, "Strategies for pre-training graph neural networks," in *ICLR*, 2019. 17, 18

[186] Y. Rong, Y. Bian, T. Xu, W. Xie, Y. Wei, W. Huang, and J. Huang, "Self-supervised graph transformer on large-scale molecular data," *NeurIPS*, 2020. 17

[187] Z. Hou, X. Liu, Y. Cen, Y. Dong, H. Yang, C. Wang, and J. Tang, "Graphmae: Self-supervised masked graph autoencoders," in *SIGKDD*, 2022, pp. 594–604. 17

[188] Z. Hou, Y. He, Y. Cen, X. Liu, Y. Dong, E. Kharlamov, and J. Tang, "Graphmae2: A decoding-enhanced masked self-supervised graph learner," in *WWW*, 2023, pp. 737–746. 17

[189] Q. Tan, N. Liu, X. Huang, R. Chen, S.-H. Choi, and X. Hu, "Mgae: Masked autoencoders for self-supervised learning on graphs," *arXiv preprint*, 2022. 17

[190] S. Zhang, H. Chen, H. Yang, X. Sun, P. S. Yu, and G. Xu, "Graph masked autoencoders with transformers," *arXiv preprint*, 2022. 17

[191] E. Min, R. Chen, Y. Bian, T. Xu, K. Zhao, W. Huang, P. Zhao, J. Huang, S. Ananiadou, and Y. Rong, "Transformer for graphs: An overview from architecture perspective," *arXiv preprint*, 2022. 17

[192] R. Rao, N. Bhattacharya, N. Thomas, Y. Duan, P. Chen, J. Canny, P. Abbeel, and Y. Song, "Evaluating protein transfer learning with tape," *NeurIPS*, 2019. 17

[193] A. Rives, J. Meier, T. Sercu, S. Goyal, Z. Lin, J. Liu, D. Guo, M. Ott, C. L. Zitnick, J. Ma *et al.*, "Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences," *NAS*, vol. 118, no. 15, p. e2016239118, 2021. 17

[194] M. McDermott, B. Yap, H. Hsu, D. Jin, and P. Szolovits, "Adversarial contrastive pre-training for protein sequences," *arXiv preprint*, 2021. 17

[195] L. He, S. Zhang, L. Wu, H. Xia, F. Ju, H. Zhang, S. Liu, Y. Xia, J. Zhu, P. Deng *et al.*, "Pre-training co-evolutionary protein representation via a pairwise masked language model," *arXiv preprint*, 2021. 17

[196] A. X. Lu, H. Zhang, M. Ghassemi, and A. Moses, "Self-supervised contrastive learning of protein representations by mutual information maximization," *BioRxiv*, 2020. 17

[197] K. Gao, L. Wu, J. Zhu, T. Peng, Y. Xia, L. He, S. Xie, T. Qin, H. Liu, K. He *et al.*, "Pre-training antibody language models for antigen-specific computational antibody design," in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023, pp. 506–517. 17

[198] Z. Zhang, M. Xu, A. Jamasb, V. Chenthamarakshan, A. Lozano, P. Das, and J. Tang, "Protein representation learning by geometric structure pretraining," in *ICLR*, 2023. 17

[199] Y. You and Y. Shen, "Cross-modality and self-supervised protein embedding for compound–protein affinity and contact prediction," *Bioinformatics*, vol. 38, no. Supplement_2, pp. ii68–ii74, 2022. 17

[200] M. Hu, F. Yuan, K. K. Yang, F. Ju, J. Su, H. Wang, F. Yang, and Q. Ding, "Exploring evolution-based &-free protein language models as protein function predictors," *arXiv preprint*, 2022. 17

[201] K. K. Yang, N. Zanichelli, and H. Yeh, "Masked inverse folding with sequence transfer for protein representation learning," *bioRxiv*, 2022. 17

[202] J. Su, C. Han, Y. Zhou, J. Shan, X. Zhou, and F. Yuan, "Saprot: Protein language modeling with structure-aware vocabulary," *bioRxiv*, 2023. 17

[203] Z. Gao, C. Tan, and S. Z. Li, "Vqpl: Vector quantized protein language," *ArXiv*, 2023. 17

[204] M. van Kempen, S. S. Kim, C. Tumescheit, M. Mirdita, J. Söding, and M. Steinegger, "Foldseek: fast and accurate protein structure search," 2022. 17

[205] S. Wang, Y. Guo, Y. Wang, H. Sun, and J. Huang, "Smiles-bert: large scale unsupervised pre-training for molecular property prediction," in *ICBCB*, 2019, pp. 429–436. 18

[206] J. Ross, B. Belgodere, V. Chenthamarakshan, I. Padhi, Y. Mroueh, and P. Das, "Large-scale chemical language representations capture molecular structure and properties," *Nature Machine Intelligence*, vol. 4, no. 12, pp. 1256–1264, 2022. 18

[207] J. Xia, C. Zhao, B. Hu, Z. Gao, C. Tan, Y. Liu, S. Li, and S. Z. Li, "Mole-bert: Rethinking pre-training graph neural networks for molecules," in *ICLR*, 2022. 18

[208] P.-Y. Huang, H. Xu, J. B. Li, A. Baevski, M. Auli, W. Galuba, F. Metze, and C. Feichtenhofer, "Masked autoencoders that listen," *ArXiv*, 2022. 18

[209] A. El-Nouby, G. Izacard, H. Touvron, I. Laptev, H. Jégou, and E. Grave, "Are large-scale datasets necessary for self-supervised pre-training?" *ArXiv*, 2021. 25

[210] Y. Yao, N. Desai, and M. S. Palaniswami, "Masked contrastive representation learning," *ArXiv*, 2022. 25

[211] J. Lu, C. Clark, R. Zellers, R. Mottaghi, and A. Kembhavi, "Unified-io: A unified model for vision, language, and multi-modal tasks," *arXiv preprint*, 2022. 25

[212] Y. Wei, H. Hu, Z. Xie, Z. Zhang, Y. Cao, J. Bao, D. Chen, and B. Guo, "Contrastive learning rivals masked image modeling in fine-tuning via feature distillation," *ArXiv*, 2022. 25

[213] H. Xue, P. Gao, H. Li, Y. J. Qiao, H. Sun, H. Li, and J. Luo, "Stare at what you see: Masked image modeling without reconstruction," *ArXiv*, 2022. 25

[214] Y. Liu, S. Zhang, J. Chen, Z. Yu, K. Chen, and D. Lin, "Improving pixel-based mim by reducing wasted modeling capability," *ArXiv*, 2023. 25

[215] Q. Huang, X. Dong, D. Chen, Y. Chen, L. Yuan, G. Hua, W. Zhang, N. H. Yu, and M. Reaserch, "Improving adversarial robustness of masked autoencoders via test-time frequency-domain prompting," *ArXiv*, 2023. 25

[216] Q. Han, Y. Cai, and X. Zhang, "Revcolv2: Exploring disentangled representations in masked image modeling," *ArXiv*, 2023. 26

[217] J. Zhu, X. Ding, Y. Ge, Y. Ge, S. Zhao, H. Zhao, X. Wang, and Y. Shan, "Vl-gpt: A generative pre-trained transformer for vision and language understanding and generation," 2023. 26, 28

[218] A. Chen, K. Zhang, R. Zhang, Z. Wang, Y. Lu, Y. Guo, and S. Zhang, "Pimae: Point cloud and image interactive masked autoencoders for 3d object detection," in *CVPR*, June 2023, pp. 5291–5301. 26

[219] Z. Zhao, S. Wei, Q. Chen, D. Li, Y. Yang, Y. Peng, and Y. Liu, "Masked retraining teacher-student framework for domain adaptive object detection," in *ICCV*, October 2023, pp. 19 039–19 049. 26

[220] K. Yue, B.-C. Chen, J. Geiping, H. Li, T. Goldstein, and S.-N. Lim, "Object recognition as next token prediction," 2023. 26

[221] S. Lao, G. Song, B. Liu, Y. Liu, and Y. Yang, "Masked autoencoders are stronger knowledge distillers," in *ICCV*, October 2023, pp. 6384–6393. 26

[222] W. Yan, Y. Zhang, P. Abbeel, and A. Srinivas, "Videogpt: Video generation using vq-vae and transformers," *ArXiv*, 2021. 26

[223] R. Wang, D. Chen, Z. Wu, Y. Chen, X. Dai, M. Liu, Y.-G. Jiang, L. Zhou, and L. Yuan, "Bevt: Bert pretraining of video transformers," in *CVPR*, 2022, pp. 14 713–14 723. 26

[224] C. Feichtenhofer, H. Fan, Y. Li, and K. He, "Masked autoencoders as spatiotemporal learners," *ArXiv*, 2022. 26

[225] Y. Ge, Y. Ge, X. Liu, A. Wang, J. Wu, Y. Shan, X. Qie, and P. Luo, "Miles: Visual bert pre-training with injected language semantics for video-text retrieval," *ArXiv*, 2022. 26

[226] Z. Qing, S. Zhang, Z. Huang, X. Wang, Y. Wang, Y. Lv, C. Gao, and N. Sang, "Mar: Masked autoencoders for efficient action recognition," *ArXiv*, 2022. 26

[227] Q. Wu, T. Yang, Z. Liu, B. Wu, Y. Shan, and A. B. Chan, "Dropmae: Masked autoencoders with spatial-attention dropout for tracking tasks," *CVPR*, 2023. 26

[228] R. Wang, D. Chen, Z. Wu, Y. Chen, X. Dai, M. Liu, L. Yuan, and Y.-G. Jiang, "Masked video distillation: Rethinking masked feature modeling for self-supervised video representation learning," in *CVPR*, 2023, pp. 6312–6322. 26

[229] B. Huang, Z. Zhao, G. Zhang, Y. Qiao, and L. Wang, "Mgmae: Motion guided masking for video masked autoencoding," *ArXiv*, 2023. 26

[230] J. Cheng, X. Mei, and M.-Y. Liu, "Forecast-mae: Self-supervised pre-training for motion forecasting with masked autoencoders," *ArXiv*, 2023. 26

[231] H. Chen, J. Wang, K. Shao, F. Liu, J. Hao, C. Guan, G. Chen, and P.-A. Heng, "Traj-mae: Masked autoencoders for trajectory prediction," *ArXiv*, 2023. 26

[232] D. Fan, J. Wang, S. Liao, Y. Zhu, V. Bhat, H. J. Santos-Villalobos, M. V. Rohith, and X. Li, "Motion-guided masking for spatiotemporal representation learning," *ArXiv*, 2023. 26

[233] Y. Mao, J. Deng, W. gang Zhou, Y. Fang, W. Ouyang, and H. Li, "Masked motion predictors are strong 3d action representation learners," *ArXiv*, 2023. 26

[234] H. Yan, Y. Liu, Y. Wei, Z. Li, G. Li, and L. Lin, "Skeletonmae: Graph-based masked autoencoder for skeleton sequence pre-training," *ArXiv*, 2023. 26

[235] L. Chen, J. Zhang, Y. rong Li, Y. Pang, X. Xia, and T. Liu, "Humanmac: Masked motion completion for human motion prediction," *ArXiv*, 2023. 26

[236] J. Liu, T. Wang, B. Liu, Q. Zhang, Y. Liu, and H. Li, "Towards better 3d knowledge transfer via masked image modeling for multi-view 3d understanding," *ArXiv*, 2023. 26

[237] A. Gupta, J. Wu, J. Deng, and L. Fei-Fei, "Siamese masked autoencoders," *ArXiv*, 2023. 26

[238] C. Lu, X. Jin, Z. Huang, Q. Hou, M.-M. Cheng, and J. Feng, "Cmae-v: Contrastive masked autoencoders for video action recognition," *ArXiv*, 2023. 26

[239] Q. Yang, W. Li, B. Li, and Y. Yuan, "Mrm: Masked relation modeling for medical image pre-training with genetics," in *ICCV*, 2023, pp. 21 452–21 462. 26

[240] C. Reed, R. Gupta, S. Li, S. Brockman, C. Funk, B. Clipp, S. Candido, M. Uyttendaele, and T. Darrell, "Scale-mae: A scale-aware masked autoencoder for multiscale geospatial representation learning," *ArXiv*, 2022. 26

[241] Y. Chen, Z. Xiao, L. Zhao, L. Zhang, H. Dai, D. Liu, Z. Wu, C. Li, T. Zhang, C. Li, D. Zhu, T. Liu, and X. Jiang, "Mask-guided vision transformer (mg-vit) for few-shot learning," *ArXiv*, 2022. 26

[242] Y. Liang, S. Zhao, B. Yu, J. Zhang, and F. He, "Meshmae: Masked autoencoders for 3d mesh data analysis," in *ECCV*, 2022. 26

[243] Y. Pang, W. Wang, F. E. H. Tay, W. Liu, Y. Tian, and L. Yuan, "Masked autoencoders for point cloud self-supervised learning," in *ECCV*, 2022. 26

[244] H. Liu, M. Cai, and Y. J. Lee, "Masked discrimination for self-supervised learning on point clouds," in *ECCV*, 2022. 26

[245] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg,

and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *IJCV*, pp. 211–252, 2014. 27
[246] T.-Y. Lin, M. Maire, S. J. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, 2014. 27
[247] M. Wang and W. Deng, "Oracle-mnist: a realistic image dataset for benchmarking machine learning algorithms," *ArXiv*, 2022. 27
[248] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," *CVPR*, pp. 3213–3223, 2016. 27
[249] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, A. Natsev, M. Suleyman, and A. Zisserman, "The kinetics human action video dataset," *ArXiv*, 2017. 27
[250] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *ArXiv*, 2012. 27
[251] A. Miech, J.-B. Alayrac, I. Laptev, J. Sivic, and A. Zisserman, "Rareact: A video dataset of unusual interactions," *ArXiv*, 2020. 27
[252] G.-S. Xia, J. Hu, F. Hu, B. Shi, X. Bai, Y. Zhong, L. Zhang, and X. Lu, "Aid: A benchmark data set for performance evaluation of aerial scene classification," *TGRS*, vol. 55, pp. 3965–3981, 2016. 27
[253] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results." 27
[254] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *ICVGIP*, Dec 2008. 27
[255] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba, "Sun database: Large-scale scene recognition from abbey to zoo," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 3485–3492. 27
[256] S. H. Lee, S. Lee, and B. C. Song, "Vision transformer for small-size datasets," *ArXiv*, 2021. 27
[257] A. Krizhevsky, "Learning multiple layers of features from tiny images," 2009. 27
[258] A. Coates, A. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," in *AISTATS*, 2011. 27
[259] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The caltech-ucsd birds-200-2011 dataset," 2011. 27
[260] S. Maji, E. Rahtu, J. Kannala, M. B. Blaschko, and A. Vedaldi, "Fine-grained visual classification of aircraft," *ArXiv*, 2013. 27
[261] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3d object representations for fine-grained categorization," in *ICCV workshops*, 2013, pp. 554–561. 27
[262] B. Zhou, A. Khosla, À. Lapedriza, A. Torralba, and A. Oliva, "Places: An image database for deep scene understanding," *ArXiv*, 2016. 27
[263] G. V. Horn, O. M. Aodha, Y. Song, A. Shepard, H. Adam, P. Perona, and S. J. Belongie, "The inaturalist challenge 2017 dataset," *ArXiv*, 2017. 27
[264] S. Moschoglou, A. Papaioannou, C. Sagonas, J. Deng, I. Kotsia, and S. Zafeiriou, "Agedb: The first manually collected, in-the-wild age database," in *CVPRW*, 2017, pp. 1997–2005. 27
[265] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," *ArXiv*, 2017. 27
[266] Y. Liao, J. Xie, and A. Geiger, "Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d," *TPAMI*, pp. 3292–3310, 2021. 27
[267] A. X. Chang, T. A. Funkhouser, L. J. Guibas, P. Hanrahan, Q.-X. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "Shapenet: An information-rich 3d model repository," *ArXiv*, 2015. 27
[268] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," in *CVPRW*, 2004, pp. 178–178. 27
[269] G. A. Sigurdsson, G. Varol, X. Wang, A. Farhadi, I. Laptev, and A. K. Gupta, "Hollywood in homes: Crowdsourcing data collection for activity understanding," in *ECCV*, 2016. 27
[270] C. Gu, C. Sun, S. Vijayanarasimhan, C. Pantofaru, D. A. Ross, G. Toderici, Y. Li, S. Ricco, R. Sukthankar, C. Schmid, and J. Malik, "Ava: A video dataset of spatio-temporally localized atomic visual actions," in *CVPR*, 2017, pp. 6047–6056. 27
[271] A. Gupta, P. Dollár, and R. B. Girshick, "Lvis: A dataset for large vocabulary instance segmentation," *CVPR*, pp. 5351–5359, 2019. 27
[272] S. Changpinyo, P. K. Sharma, N. Ding, and R. Soricut, "Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts," in *CVPR*, 2021, pp. 3557–3567. 27
[273] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman, P. Schramowski, S. Kundurthy, K. Crowson, L. Schmidt, R. Kaczmarczyk, and J. Jitsev, "Laion-5b: An open large-scale dataset for training next generation image-text models," *ArXiv*, 2022. 27
[274] J. Carreira, E. Noland, C. Hillier, and A. Zisserman, "A short note on the kinetics-700 human action dataset," *ArXiv*, 2019. 27
[275] A. Rohrbach, M. Rohrbach, N. Tandon, and B. Schiele, "A dataset for movie description," in *CVPR*, 2015, pp. 3202–3212. 27
[276] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, "Scene parsing through ade20k dataset," in *CVPR*, 2017, pp. 5122–5130. 27
[277] A. Rohrbach, M. Rohrbach, W. Qiu, A. Friedrich, M. Pinkal, and B. Schiele, "Coherent multi-sentence video description with variable level of detail," in *GCPR*, 2014. 27
[278] G. Lai, Q. Xie, H. Liu, Y. Yang, and E. H. Hovy, "Race: Large-scale reading comprehension dataset from examinations," *ArXiv*, 2017. 27
[279] D. F. Campos, T. Nguyen, M. Rosenberg, X. Song, J. Gao, S. Tiwary, R. Majumder, L. Deng, and B. Mitra, "Ms marco: A human generated machine reading comprehension dataset," *ArXiv*, 2016. 27
[280] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *ICASSP*, 2017, pp. 776–780. 27
[281] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *ICASSP*, 2015, pp. 5206–5210. 27
[282] X. Zhai, A. Kolesnikov, N. Houlsby, and L. Beyer, "Scaling vision transformers," *CVPR*, pp. 1204–1213, 2021. 27
[283] Z. Chen, J. Wu, W. Wang, W. Su, G. Chen, S. Xing, Z. Muyan, Q. Zhang, X. Zhu, L. Lu, B. Li, P. Luo, T. Lu, Y. Qiao, and J. Dai, "Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks," 2023. 27
[284] H. Liu, W. Yan, and P. Abbeel, "Language quantized autoencoders: Towards unsupervised text-image alignment," *ArXiv*, 2023. 28
[285] L. Yu, Y. Cheng, Z. Wang, V. Kumar, W. Macherey, Y. Huang, D. A. Ross, I. Essa, Y. Bisk, M. Yang, K. P. Murphy, A. G. Hauptmann, and L. Jiang, "Spae: Semantic pyramid autoencoder for multimodal generation with frozen llms," *ArXiv*, 2023. 28

Model	Category	Type	Mask	Encoder	Target	MIM Head	CL Head	Loss	Publish
iGPT [21]	BTTM	AR	AR Mask	Transformer	Offline, Tokenizer	Linear	-	CE	ICML'2020
VL-BERT [24]	BTTM	AE	Random	Tansformer	Tokenizer	Linear	-	CE	ICLR'2020
MST [74]	ATPM	AE	Attention	Transformer	Feature, Pixel	MLP	-	CE, MSE	NIPS'2021
SplitMask [209]	BTTM	AE	Random	Transformer	Tokenizer	-	Softmax	CE	arXiv'2021
BEiT [59]	BTTM	AE	Random	Transformer	Offline Tokenizer	Linear	-	CE	ICLR'2022
iBOT [58]	BTTM	AE	Random	Transformer	Tokenizer	MLP	-	CE	ICLR'2022
data2vec [53]	BTFM	AE	Random	Transformer	Feature	Linear	-	ℓ_1	ICML'2022
ADIOS [75]	ATPM	AE	Adversarial	ResNet, Transformer	Pixel	MLP	-	MSE	ICML'2022
MP3 [55]	BTFM	AE	Random	Transformer	Feature	Linear	-	MSE	ICML'2022
MAE [23]	BTPM	AE	Random	Transformer	Pixel	Transformer	-	MSE	CVPR'2022
SimMIM [36]	BTPM	AE	Random	Transformer	Pixel	Linear	-	MSE	CVPR'2022
MaskFeat [56]	BTFM	AE	Random	Transformer	Feature	Linear	-	MSE	CVPR'2022
MaskGIT [63]	BTTM	AR	Random	Transformer	Tokenizer	Transformer	-	CE	CVPR'2022
AttMask [86]	ATFM	AE	Attention	Transformer	Feature	Transformer	-	CE	ECCV'2022
mc-BEiT [65]	BTTM	AE	Random	Transformer	Tokenizer	MLP	-	CE	ECCV'2022
BootMAE [42]	BTPM	AE	Random	Transformer	Pixel, Feature	Transformer	-	MSE	ECCV'2022
SdAE [43]	BTPM	AE	Random	Transformer	Pixel	Transformer	-	Cosine	ECCV'2022
MultiMAE [57]	BTFM	AE	Random	Transformer	Feature	Transformer	-	MSE	ECCV'2022
CAE [47]	BTFM	AE	Random	Transformer	Feature	Transformer	-	CE, MSE	IJCV'2023
CAE.v2 [51]	BTFM	AE	Random	Transformer	Feature	FC	-	Cosine	arXiv'2022
SemMAE [77]	ATPM	AE	Semantic Guided	Transformer	Pixel	Transformer	-	MSE	NIPS'2022
TTT-MAE [44]	BTPM	AE	Random	Transformer	Pixel	Transformer	-	MSE	NIPS'2022
GreenMIM [114]	BTPM	AE	Random	Transformer	Pixel	Transformer	-	MSE	NIPS'2022
ConvMAE [92]	BCPM	AE	Random	Transformer,CNN	Pixel	Transformer	-	MSE	NIPS'2022
MSN [94]	BTFC	AE	Random	Transformer	Feature	-	Softmax	CE	arXiv'2022
RePre [37]	BTPM	AE	Random	Transformer	Pixel	CNN Transformer	-	MSE	arXiv'2022
MACRL [210]	BTPM	AE	Random	Transformer	Pixel	Transformer	MLP	InfoNCE, MSE	arXiv'2022
Unified-IO [211]	BTFM	AE	Binary	Transformer	Feature	Transformer	-	InfoNCE	arXiv'2022
UnMAE [76]	ATPM	AE	Uniform Sampling	Transformer	Pixel	Transformer	-	MSE	arXiv'2022
SIM [48]	BTFM	AE	Random	Transformer	Feature	Transformer	-	MSE	arXiv'2022
ExtreMA [95]	BTFC	AE	Random	Transformer	Feature	-	FC	InfoNCE	arXiv'2022
LoMaR [78]	ATPM	AE	Local Mask	Transformer	Pixel	Transformer	-	MSE	arXiv'2022
CMAE [100]	ATPC	AE	Local Mask	Transformer	Pixel	-	FC	InfoNCE, MSE	arXiv'2022
MaskCLIP [69]	BTFB	AE	Random	Transformer	Feature	Transformer	FC	InfoNCE, MSE	arXiv'2022
BEiT.v2 [60]	BTTM	AE	Random	Transformer	Offline Tokenizer	Linear	-	CE	arXiv'2022
BEiT.v3 [26]	BTTM	AE	Random	Transformer	Tokenizer	Linear	-	CE	arXiv'2022
DMAE [38]	BTPM	AE	Random	Transformer	Pixel	Transformer	-	MSE	arXiv'2022
MILAN [87]	ATFM	AE	Attention	Transformer	Feature	Transformer	-	MSE	arXiv'2022
MimCo [96]	BTFC	AE	Random	Transformer	Feature	-	FC	InfoNCE	arXiv'2022
dBOT [49]	BTFM	AE	Random	Transformer	Feature	Transformer	-	ℓ_1	arXiv'2022
RC-MAE [39]	BTPM	AE	Random	Transformer	Pixel	Transformer	-	MSE	arXiv'2022
MaskDistill [50]	BTFM	AE	Random	Transformer	Feature	Transformer	-	ℓ_1, Cosine	arXiv'2022
i-MAE [79]	ATPM	AE	Mixture	Transformer	Pixel	Transformer	-	MSE	arXiv'2022
CAE.V2 [51]	BTFM	AE	Random	Transformer	Feature	FC	-	Cosine	arXiv'2022
FastMIM [52]	BTFM	AE	Random	Transformer	HOG Feature	Transformer	-	MSE	arXiv'2022
A-CLIP [101]	ATFC	AE	Attention	Transformer	Feature	-	FC	InfoNCE	arXiv'2022
MixMIM [84]	ATPM	AE	Mixture	Transformer	Pixel	Transformer	-	MSE	arXiv'2022
MVP [66]	BTTM	AE	Random	Transformer	Token	Linear	-	CE	arXiv'2022
FD [212]	BTFM	AE	Random	Transformer	Feature	FC	-	ℓ_1	arXiv'2022
ObjMAE [85]	ATPM	AE	Hard Sampling	Transformer	Pixel	Transformer	-	MSE	arXiv'2022
SDMAE [73]	ATFB	AE	Contextual	Transformer	Pixel, Feature	Transformer	FC	InfoNCE, MSE	arXiv'2022
Ge2AE [70]	BTFB	AE	Random	Transformer	Fourier Feature	Transformer	FC	Focal FFT, MSE	AAAI'2023
DILEMMA [105]	BTFM	AE	Random	Transformer	Feature	Transformer	-	CE	AAAI'2023
PeCo [67]	BTTM	AE	Random	Transformer	Token	Linear	-	CE	AAAI'2023
data2vec2.0 [89]	ATFM	AE	Multi-Masking	Transformer	Feature	CNN	-	MSE	ICML'2023
A2MIM [109]	BCFM	AE	Random	Transformer, CNN	Fourier, HOG Feature	Linear	-	ℓ_1, Focal FFT	ICML'2023
Hiera [41]	BTPM	AE	Random	Transformer	Pixel	Transformer	-	MSE	ICML'2023
MAE-Lite [46]	BTPM	AE	Random	Transformer	Pixel	Transformer	-	MSE	ICML'2023
ConMIM [71]	BTPC	AE	Random	Transformer	Pixel	-	FC	InfoNCE	ICLR'2023
HiViT [115]	BTPM	AE	Random	Transformer	Pixel	Transformer	-	MSE	ICLR'2023
Layer Grafted [72]	BTPC	AE	Random	Transformer	Pixel	-	FC	InfoNCE, MSE	ICLR'2023
ccMIM [80]	ATPM	AE	Attention	Transformer	Pixel	Transformer	-	MSE	ICLR'2023
RandSAC [62]	BTTM	AR	Random	Transformer	Tokenizer	Transformer	-	CE	ICLR'2023
Spark [91]	BCPM	AE	Random	CNN	Pixel	CNN	-	MSE	ICLR'2023
CIM [64]	BCTM	AE	Random	Transformer,CNN	Tokenizer	Transformer	-	CE	ICLR'2023
MaskVLM [45]	BTPM	AE	Random	Transformer	Pixel, Feature	Transformer	-	MSE	ICLR'2023
ConvNext.v2 [90]	BCPM	AE	Random	CNN	Pixel	CNN	-	MSE	CVPR'2023
MAGE [68]	BTTB	AE, AR	Random	Transformer	Tokenizer	Transformer	MLP	CE, InfoNCE	CVPR'2023
I-JEPA [83]	ATPM	AE	Contextual	Transformer	Pixel	Transformer	-	L2	CVPR'2023
HPM [82]	ATPM	AE	Hard Sampling	Transformer	Pixel	Transformer	-	MSE	CVPR'2023
FLIP [97]	BTFC	AE	Random	Transformer	Text, Feature	-	FC	InfoNCE	CVPR'2023
AutoMAE [81]	ATPM	AE	Adversarial	Transformer	Pixel	Transformer	-	MSE	CVPR'2023
LocalMAE [117]	BTFM	AE	Random	Transformer	Feature	Transformer	-	MSE	CVPR'2023
MaskAlign [213]	ATFM	AE	Attention	Transformer	Feature	MLP	-	MSE	CVPR'2023
MFM [214]	BTFM	AE	Random	Transformer	Feature	Transformer	-	MSE	ICCV'2023
SparseMAE [116]	BTFM	AE	Random	Transformer	Pixel	Transformer	-	MSE	ICCV'2023
MFM [54]	BCFM	AE	Random	Transformer, CNN	Fourier Feature	Linear	-	Fourier Loss	ICCV'2023
SparseMAE [116]	BTPM	AE	Random	Transformer	Pixel	Transformer	-	MSE	ICCV'2023
RobustMAE [215]	BTFM	AE	Random	Transformer	Feature	Transformer	-	CE	ICCV'2023
CAN [93]	BTPB	AE	Random	Transformer	Pixel	Transformer	FC	InfoNCE, MSE	ICCV'2023

TABLE 4: Detailed information of fundamental masked image modeling (MIM) methods (**view Table 5 to continue**).

Model	Category	Type	Mask	Encoder	Target	MIM Head	CL Head	Loss	Publish
DropPos [108]	BTFM	AE	Random	Transformer	Feature	MLP	-	CE	NIPS'2023
RevColV2 [216]	BTPM	AE	Random	Transformer	Pixel	Transformer	-	MSE	NIPS'2023
MaPeT [61]	BTTM	AE, AR	Random	Transformer	Tokenizer	Transformer	-	Likehood	arXiv'2023
R-MAE [40]	BCPM	AE	Random	Transformer	Pixel	Transformer	-	CE	arXiv'2023
DMJD [88]	ATFM	AE	Disjoint	Transformer	Feature	Transformer	-	MSE	arXiv'2023
MOMA [98]	BTFC	AE	Random	Transformer	Feature	-	FC	InfoNCE	arXiv'2023
PixMIM [111]	BTFM	AE	Random	Transformer	Feature	Transformer	-	MSE	arXiv'2023
TinyMIM [113]	BTFM	AE	Random	Transformer	Feature	Transformer	-	MSE	arXiv'2023
MSCN [110]	BTFM	AE	Random	Transformer	Feature	MLP	-	MSE	arXiv'2023
Img2vec [112]	BTFM	AE	Random	Transformer	Feature	MLP	-	MSE	arXiv'2023
DeepMIM [99]	BTFM	AE	Random	Transformer	Pixel, Feature	Transformer	-	MSE	arXiv'2023
D-iGPT [99]	BTTB	AE	Random	Transformer	Tokenizer	Transformer	-	CE	arXiv'2023
VL-GPT [217]	BTTM	AE	Random	Transformer	Tokenizer	Transformer	-	CE, MSE	arXiv'2023
LVM [139]	BTTM	AR	AR Mask	Transformer	Tokenizer	Transformer	-	CE	arXiv'2023

TABLE 5: Detailed information of fundamental masked image modeling (MIM) methods (**continue Table** 4).

Model	Task	Type	Category	Mask	Encoder	Target	Head	Publication
MIMDet [149]	Detection	AE	RTTM	Random	Transformer	Token	MIM Head	arXiv'2022
iTPN [148]	Detection, Segmentation	AE	BTFM	Random	Transformer	Feature	MIM Head	CVPR'2023
imTED [150]	Detection	AE	BTFM	Random	Transformer	Feature	MIM Head	CVPR'2023
PiMAE [218]	Detection	AE	BTFM	Random	Transformer	Feature	MIM Head	ICCV'2023
MRT [219]	Detection	AE	ATFM	Hard Sampling	Transformer	Feature	MIM Head	ICCV'2023
NXTP [220]	Detection	AR	BTTM	AR Mask	Transformer	Token	MIM Head	arXiv'2023
FreMAE [159]	Medical Image	AE	BTFM	Random	Transformer	Fourier Feature	MIM Head	arXiv'2023
G2SD [152]	KD	AE	BTFM	Random	Transformer	Feature	MIM Head	CVPR'2023
MKD [221]	KD	AE	BTFM	Random	Transformer	Feature	MIM Head	ICCV'2023
VideoGPT [222]	Video	AR	BTTM	AR Mask	Transformer	Token	MIM Head	arXiv'2021
BEVT [223]	Video	AE	BTTM	Random	Transformer	Token	MIM Head	CVPR'2022
MAE [224]	Video	AE	BTPM	Random	Transformer	Pixel	MIM Head	NIPS'2022
VideoMAE [140]	Video	AE	BTPM	Random	Transformer	Pixel	MIM Head	NIPS'2022
FMNet [147]	Video	AE	BTFM	Random	Tranformer	Feature	MIM Head	ACMMM'2022
MILES [225]	Video	AE	ATFM	Contextual	Transformer	Feature	MIM Head	arXiv'2022
MAR [226]	Video	AE	ATPM	Cell Running	Transformer	Pixel	MIM Head	arXiv'2022
OmniMAE [144]	Video	AE	BTPM	Random	Transformer	Pixel	MIM Head	arXiv'2022
MotionMAE [143]	Video	AE	BTPM	Random	Transformer	Pixel	MIM Head	arXiv'2022
MAM2 [145]	Video	AE	BTTM	Random	Transformer	Token	MIM Head	arXiv'2022
MaskViT [146]	Video	AE, AR	BTTM	Random	Transformer	Token	MIM Head	CVPR'2023
DropMAE [227]	Video	AE	BTPM	Random	Transformer	Pixel	MIM Head	CVPR'2023
MAGVIT [126]	Video	AE, AR	BTTM	Random	Transformer	Token	MIM Head	CVPR'2023
AdaMAE [141]	Video	AE	BTPM	Random	Transformer	Pixel	MIM Head	CVPR'2023
VideoMAE.v2 [142]	Video	AE	BTPM	Random	Transformer	Pixel	MIM Head	CVPR'2023
MVD [228]	Video	AE	BTPM	Random	Transformer	Pixel, Feature	MIM Head	CVPR'2023
MGMAE [229]	Video	AE	BTFM	Random	Transformer	Feature	MIM Head	ICCV'2023
Forecast-MAE [230]	Video	AE	BTFM	Random	Transformer	Feature	MIM Head	ICCV'2023
Traj-MAE [231]	Video	AE	BTFM	Random	Transformer	Feature	MIM Head	ICCV'2023
MGM [232]	Video	AE	ATPM	Motion Guided	Transformer	Pixel	MIM Head	ICCV'2023
HumanMAC [233]	Video	AE	BTFM	Random	Transformer	Feature	MIM Head	ICCV'2023
SkeletonMAE [234]	Video	AE	ATFM	Joint Mask	Transformer	Feature	MIM Head	ICCV'2023
MAMP [235]	Video	AE	ATFM	Motion Aware	Transformer	Feature	MIM Head	ICCV'2023
GeoMIM [236]	Video	AE	BTFM	Random	Transformer	Feature	MIM Head	ICCV'2023
SiamMAE [237]	Video	AE	BTPM	Random	Transformer	Pixel	MIM Head	arXiv'2023
CMAE-V [238]	Video	AE	BTPB	Random	Transformer	Pixel	CL & MIM Head	arXiv'2023
MRM [239]	Medical Image	AE	ATPM	Relation Mask	Transformer	Pixel	MIM Head	ICCV'2023
SD-MAE [73]	Medical Image	AE	BTPM	Random	Transformer	Pixel	MIM Head	arXiv'2022
MedMAE [158]	Medical Image	AE	BTPM	Random	Transformer	Pixel	MIM Head	arXiv'2022
GCMAE [160]	Medical Image	AE	BTPM	Random	Transformer	Pixel	MIM Head	arXiv'2022
SatMAE [163]	Remote Sensing	AE	BTPM	Consistent Independent	Transformer	Pixel	MIM Head	arXiv'2022
Scale-MAE [240]	Remote Sensing	AE	BTPM	Random	Transformer	Pixel	MIM Head	ICCV'2023
CMID [164]	Remote Sensing	AE	BTFB	Random	Transformer	Fourier Feature	CL & MIM Head	TGRS'2023
DocMAE [161]	OCR	AE	BTPM	Random	Transformer	Pixel	MIM Head	ICME'2023
MGViT [241]	Few Shot	AE	BTPM	Random	Transformer	Pixel	MIM Head	NIPS'2022
MeshMAE [242]	3D Mesh	AE	BTPM	Random	Transformer	Pixel	MIM Head	ECCV'2022
VoxelMAE [169]	3D Point	AE	BTFM	Random	Transformer	Voxel	MIM Head	arXiv'2022
PointBERT [175]	3D Point	AE	BTTM	Random	Transformer	Token	MIM Head	CVPR'2022
PointMAE [243]	3D Point	AE	BTFM	Random	Transformer	Feature	MIM Head	ECCV'2022
MaskPoint [244]	3D Point	AE	BTFM	Random	Transformer	Real & Fake	MIM Head	ECCV'2022
Point-M2AE [170]	3D Point	AE	BTPM	Random	Transformer	Pixel	MIM Head	NIPS'2022
PointCMP [176]	3D Point	AE	BTTB	Random	Transformer	Token	CL & MIM Head	CVPR'2023
I2P-MAE [171]	3D Point	AE	BTFM	Random	Transformer	Feature	MIM Head	CVPR'2023
GeoMAE [174]	3D Point	AE	BTPM	Random	Transformer	Pixel	MIM Head	CVPR'2023
ACT [172]	3D Point	AE	BTFM	Random	Transformer	Feature	MIM Head	ICLR'2023
ReCon [177]	3D Point	AE	BTFB	Random	Transformer	Feature	CL & MIM Head	ICML'2023
MGM [232]	3D Point	AE	BTPM	Random	Transformer	Pixel	MIM Head	ICCV'2023

TABLE 6: Detailed information of MIM methods for vision downstream tasks.

Dataset	Modality	Type	Pre-training	Downstream Task	Training Set	Link
ImageNet-1K [245]	CV	Image	CL MIM	Classification	1,281,167	ImageNet
COCO 2014 Detection [246]	CV	Image	CL MIM	Detection, Segmentation	83000	COCO2014
COCO 2017 Detection [246]	CV	Image	CL MIM	Detection, Segmentation	118,000	COCO2017
PASCAL Content	CV	Image	CL MIM	Segmentation	4998	PASCAL Content
MNIST [247]	CV	Image	-	Classification	60,000	MNIST
Cityscapes [248]	CV	Image	CL	Segmentation	2975	Cityscapes
Kinetics700 [249]	CV	Video	CL, MIM	Action Recognition	494,801	Kinetics
UCF101 [250]	CV	Video	CL, MIM	Action Recognition	9,537	UCF-101
RareAct [251]	CV	Video	CL MIM	Action Recognition	7,607	RareAct
AID [252]	CV	Image	CL, MIM	Classification	10,000	AID
PASCAL VOC 2007 Classification [253]	CV	Image	CL,MIM	Classification	5011	PASCAL VOC
Oxford 102 Folwers [254]	CV	Image	CL	Classification	2040	Oxford 102 Flowers
SUN397 [255]	CV	Image	CL,MIM	Classification	19,850	SUN397
Tiny-ImageNet [256]	CV	Image	CL MIM	Classification	100,000	TinyIN
CIFAR-10 [257]	CV	Image	CL	Classification	50,000	CIFAR-10
CIFAR-100 [257]	CV	Image	CL	Classification	50,000	CIFAR-100
STL-10 [258]	CV	Image	CL MIM	Classification	1,000	STL
CUB-200-2011 [259]	CV	Image	CL MIM	Classification	11,788	CUB-200-2011
FGVC-Aircraft [260]	CV	Imgae	CL MIM	Classification	6,770	Aircraft
StanfordCars [261]	CV	Image	CL MIM	Classification	8,144	StanfordCars
Places205 [262]	CV	Image	CL MIM	Recognition	2,500,000	Places205
iNaturalist [263]	CV	Image	CL MIM	Classification	675,170	iNaturalist
AgeDB [264]	CV	Image	MIM	Age Estimation	16,488	AgeDB
Fashion-MNIST [265]	CV	Image	MIM	Classification	70,000	Fashion-MNIST
KITTI-360 [266]	CV	3D Point Cloud	CL MIM	Detection, Segmentation	43552	KITTI Vision
ShapeNet [267]	CV	3D PointCloud	CL MIM	Recognition, Classification	220,000	ShapeNet
Caltech-101 [268]	CV	Image	CL MIM	Classification	3060	Caltech-101
Charades [269]	CV	Video	CL MIM	Recognition	66,500	Charades
AVA [270]	CV	Video	CL MIM	Detection	211,000	AVA
LVIS [271]	CV	Image	CL MIM	Detection	118,000	LVIS
CC12M [272]	CV, NLP	Image, Text	MM CL	Classification	12,000,000	CC12M
LAION-5B [273]	CV, NLP	Image, Text	MM CL	Classification	400,000,000	LAION
Flickr30k [] [274]	CV, NLP	Image, Text	MM CL	Image-Text Retrieval	31783	Flickr30k
COCO Caption	CV, NLP	Image, Text	MM CL	Image-Text Retrieval	82783	COCO Caption
LSMDC [275]	CV, NLP	Video, Text	MM CL	Movie Description	118,081	LSMDC
ADE20K [276]	CV, NLP	Image, Text	CL, MIM	Scene Parsing	20,000	ADE-20K
TACoS [277]	CV, NLP	Text, Video	CL, MM	Detection	2,600	TACoS
RACE [278]	NLP	Text	MLM	Reading Comprehension	28,000	RACE
MS MARCO [279]	NLP	Text	MLM	Question Answering	1,000,000	MSMAECO
AudioSet [280]	Audio, NLP	Speech, Text	MM, MLM	Sound Classification	2,000,000	AudioSet
LibriSpeech [281]	Audio	Speech	MLM	Speech Recognition	1,789,621	LibriSpeech

TABLE 7: Summary of datasets for MIM pre-training and vision downstream tasks. Link to dataset websites is provided.

	EVA [128]	EVA-02 [129]	WSP [130]	Painter [136]	ViT-G [282]	MAE(ViT-L) [23]	LVM [139]	InternVL [283]
Layer	40	24	24	24	48	16	26	48
Attention Head	16	16	32	16	16	24	32	25
Parameters	1011M	304M	1.89B	307M	1.84B	307M	3B	5903M
Pre-training Dataset	IN-21K, CC3M, CC12M	IN-21K, CC3M, CC12M	IN-1K, IN-Real,	ADE20K NYUv2	IN-1K JFT-3B	IN-1K ADE20K	UVD	LAION-COCO,COYO CC12M
Downstream Dataset	ADE, COCO, Object365, Kinitics	ADE, COCO, Object365, Kinetics	COCO, ObjectNet Kinetics	COCO, Rain, SIDD	ObjectNet Real	COCO	IN-1K Kinetics	IN-1K ADE20K
Segmentation	62.3 mIoU	63.8 mIoU	51.8 mIoU	49.9 mIoU	-	53.6 mIoU	-	58.9 mIoU
Detection	64.7 AP	65.9 AP	58.0 AP	72.2AP	-	53.3AP	-	-
Video Recognition	89.8 acc	-	86.0 acc	-	-	-	-	71.5 acc
Classification	84.0 acc	85.5 acc	90.9 acc	-	84.86 acc	87.8 acc	-	82.5 acc

TABLE 8: Experimental details and results of vision foundation models. IN denotes ImageNet datasets. LVM only performs comparison experiments of visual prompting and lacks standard benchmark results.

Model	Modality	Pre-trained Method	Pre-trained Dataset	Downstream Task
BEiT.v3 [26]	CV, NLP	MIM, MLM	IN-1K, ADE20K, COCO, NLVR2	Classification, Detection, Segmentation
MaskVLM [45]	CV, NLP	MIM,MLM,CL	CC,COCO, SBU, Flickr30K	Image-Text Retrieval, Natural Language for Visual Reasoning, Visual Entailment, Visual Question Answering
FLIP [97]	CV, NLP	MIM,CL,MLM	LAION-5B, IN-1K, COCO, Flickr30K	Classification, Image-Text Retrieval, Image Captioning, Visual Question Answering
A-CLIP [101]	CV, NLP	MIM,CL	IN-1K, YFCC100M, COCO, Flickr30K, Aircraft, MNIST	Classification (Zero-shot), Image-Text Retrieval
VL-BERT [24]	CV, NLP	MLM,MIM	COCO, RefCOCO+, VCR	Classification, Segmentation, Visual Question Answering
MaskCLIP [69]	CV, NLP	MIM,MLM,CL	IN-1K, ADE20K, COCO, Flickr30K	Classification (Zero-shot), Detection, Segmentation
MaskGIT [63]	CV, NLP	MIM	IN-1K	Image-Text Generation
VL-GPT [217]	CV, NLP	MIM	CC3M,LAION-COCO,MMC4	Image Generation, Text-to-Image Generation
DALLE [131]	CV, NLP	MIM,MLM	IN-1K, CC, COCO, CUB200	Text-Image Generation
LQAE [284]	CV, NLP	MIM,MLM	IN-1K	Text-Image Alignment
SPAE [285]	CV, NLP	MLM,MIM	IN-1K, Kinetics	Text-Image Generation
InstructCV [138]	CV, NLP	MLM	IN-1K, MSCOCO, ADE20K	Text-Image Generation

TABLE 9: Details of MIM methods with both image and text data modalities.