Emerging Property of Masked Token for Effective Pre-training

Hyesong Choi[1], Hunsang Lee[2], Seyoung Joung[1], Hyejin Park[1], Jiyeong Kim[1], and Dongbo Min[1†]

[1] Ewha W. University
[2] Hyundai Motor Company

Abstract. Driven by the success of Masked Language Modeling (MLM), the realm of self-supervised learning for computer vision has been invigorated by the central role of Masked Image Modeling (MIM) in driving recent breakthroughs. Notwithstanding the achievements of MIM across various downstream tasks, its overall efficiency is occasionally hampered by the lengthy duration of the pre-training phase. This paper presents a perspective that the optimization of masked tokens as a means of addressing the prevailing issue. Initially, we delve into an exploration of the inherent properties that a masked token ought to possess. Within the properties, we principally dedicated to articulating and emphasizing the 'data distinctiveness' attribute inherent in masked tokens. Through a comprehensive analysis of the heterogeneity between masked tokens and visible tokens within pre-trained models, we propose a novel approach termed **masked token optimization (MTO)**, specifically designed to improve model efficiency through weight recalibration and the enhancement of the key property of masked tokens. The proposed method serves as an adaptable solution that seamlessly integrates into any MIM approach that leverages masked tokens. As a result, MTO achieves a considerable improvement in pre-training efficiency, resulting in an approximately 50% reduction in pre-training epochs required to attain converged performance of the recent approaches. Code is available at `https://github.com/doihye/MTO`.

Keywords: Self-supervised Learning · Masked Image Modeling · Masked Token Optimization · Entropy · Heterogeneity

1 Introduction

Pre-training of universal language representations [7, 9, 22, 28, 29] has been a crucial area of Natural Language Processing (NLP), especially when training

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. RS-2023-00222385) and the Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No. RS-2022-00155966, Artificial Intelligence Convergence Innovation Human Resources Development (Ewha Womans University)). † Corresponding author: dbmin@ewha.ac.kr

large-scale models [3, 32]. Following the philosophy of Masked Language Modeling (MLM) [2, 7, 9, 22], Masked Image Modeling (MIM) [1, 10, 13, 38] has been at the core of recent advances in self-supervised learning for computer vision. MIM applies the principles of MLM to images, enabling effective pre-training of Transformers and improving transfer learning performances. The essence of these Masked Signal Modeling approaches lies in encouraging the model to predict the gaps in an input signal to learn the contextual relationships between signals while capturing an overall structure.

Despite the tremendous successes of MIM in diverse downstream tasks, the long pre-training phase that it entails tends to impede its efficiency. Concretely, a substantial amount of pre-training, typically from 800 to 1600 epochs, is essential to attain the convergence of the Transformer for transfer learning. Meanwhile, several methodologies have been employed for MIM in an effort to alleviate the disparities that exist between the linguistic and visual domains when leveraging the MLM concept. For instance, patch tokenization [1] is introduced to emulate the discrete nature of language tokens, while raw pixel regression [38] is adopted to attune with the continuous visual signals. However, the intrinsic properties of masked tokens, a vital component of MIM, have yet to be comprehensively surveyed by the vision community. In this paper, we address the lengthy pre-training issue caused by low convergence rates through the optimization of masked tokens, focusing on their inherent properties that have been previously overlooked.

Here we come to the pivotal inquiry that lies at the heart of the discourse: *What properties should a masked token have within the realm of MIM?* Given the premise that the masked token is selected and masked from the training data, it is imperative that the selected masked token exhibits certain specific attributes; (i) Spatial Randomness: Masked tokens must be randomly selected from input patches, so that the model can learn to predict tokens in various locations and semantics. Regarding this property, a research direction incorporating prior knowledge into the spatial randomness of masked tokens [16, 37] is currently debated. (ii) Substitutional Consistency, in the masking process, randomly selected visible tokens should consistently be replaced with the same learnable parameters [35], allowing the model to recognize and reconstruct them during pre-training. Lastly, (iii) **Data Distinctiveness**. This last facet represents a novel property that we aim to assert and demonstrate throughout the entire manuscript. It signifies that the masked token in the initial embedding should be unique token that are unlikely to manifest in the training data. Stated differently, the masked tokens should exhibit a negligible correlation with visible tokens to mitigate the possibility of obfuscation, when given as inputs to the attention layers. Employing masked tokens that are well differentiated from visible tokens enables the model to identify semantics within the training data, thereby improving focused pretext prediction capability.

Visual signals are inherently continuous, making it challenging for masked tokens to ensure data distinctiveness, as they cannot be explicitly differentiated like their discrete text token counterparts. To be specific, due to the clear semantics associated with each word in the text, the distinctiveness of masked tokens

can be easily preserved during the pretext prediction process in the linguistic domain. Contrarily, the Tokenizer-based approach [19] has reported that in image tokenizers, different semantic patches can have similar token under visual discretization. This finding indicates that the task of distinguishing masked tokens from visible tokens is adverse in the context of the visual tokenizer, where patches are represented as continuous values. Hence, attaining the desired distinctiveness of the masked token to the training data solely relies upon the model's convergence through a prolonged pre-training process, akin to the predictions of a black-box system devoid of explicit constraints. Therefore, we propose an analysis of masked tokens and optimization based on it by directing our attention toward the data distinctiveness among the trifecta of properties.

Our initial step encompassed a heterogeneity analysis of masked tokens against the visible tokens to demonstrate the manifestation of the masked token's data distinctiveness characteristic within the model upon reaching convergence. Moreover, the scope of this analysis is designed to investigate both the extent and the tendency of how heterogeneity unfolds throughout the different layers of the network's architecture. Building upon the insights from the heterogeneity analysis, we propose a sophisticated method for optimizing masked tokens. The proposed Masked Token Optimization (MTO) approach includes a strategic exclusion of semantically inconsequential masked tokens from the weight aggregation process associated with visible tokens, achieved through weight recalibration. At the same time, the proposed MTO method explicitly imposes constraints on data distinctiveness throughout the optimization of masked tokens to reinforce the model's capacity to differentiate between tasks, given the distinctive roles that masked tokens and visible tokens assume within the architecture; the masked token is integral to pretext prediction, whereas the visible token is essential for the encoding and decoding of representations.

The proposed Masked Token Optimization (MTO) represents a versatile and adaptable method capable of seamless integration into any MIM-based approach utilizing masked tokens, thus empowering pre-training operations with heightened efficiency and performance. The succeeding sections of the paper present the empirical evidence of the efficacy of the MTO approach when integrated into diverse MIM methodologies including SimMIM [38], MAE [13] and BootMAE [10]. The findings demonstrate that the application of MTO induces rapid model convergence and substantial improvements in representation learning. Notably, MTO improves the pre-training efficiency by approximately halving the pre-training epochs required to reach converged performance in the recent MIM approaches. Such outcomes provide a compelling justification for the wide-scale adoption of MTO as an useful plug-and-play tool in pre-training procedures.

2 Preliminaries

We start by revisiting the recent framework of MIM. The latest advancements in MIM [10,13,38] have surpassed the past two-stage methods [1,19] by integrating masked prediction and the autoencoder training in a single end-to-end process,

aiming at encoding valuable representation and predicting pretext for masked patches [42]. As these approaches are built upon Transformer architectures [11, 23, 24, 33], we assume the underlying framework is an attention model [11, 24] throughout the paper. An input image $I \in \mathbb{R}^{HW \times 3}$ is first divided into non-overlapping $N = H \times W/P^2$ patches. Then, patches are randomly sampled and masked with a high masking ratio, reflecting the information redundancy [13].

Let δ_M be defined as a set of masked indexes where visible tokens are replaced by a mask token. In general, the representation of masked modeling is trained via the minimization of the following self-supervised objective:

$$\mathcal{L}_{ss}(f(I;\Theta)) = \frac{1}{|\delta_M|} \sum_{i \in \delta_M} \|f(I;\Theta)_i - I_i\|_2^2, \tag{1}$$

with pretext prediction network f and its learnable parameters Θ. In SimMIM [38], f is jointly learned with semantic encoding using masked tokens within the encoder. In contrast, the encoder of MAE [13] solely leverages the visible image tokens. Then, the encoded visual patches are fed into the Transformer decoder, where masked tokens are employed for pretext prediction.

3 Analysis

To investigate the tendency of the heterogeneity between masked token and visible tokens, we analyze the pre-trained models of the recent approaches [13,38].

3.1 Heterogeneity Measure via Entropy

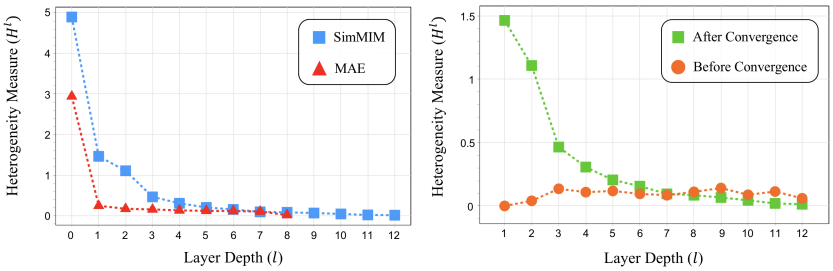
We define the degree of heterogeneity as the mutual dependence of masked tokens with respect to visible tokens in each layer as follows:

$$H = -\frac{1}{|\delta_M|} \sum_i \sum_j A_{i,j} \log(A_{i,j}) \qquad \text{where} \quad A = \psi(X_M X_V^\top). \tag{2}$$

We define $X_M = \{x_i | i \in \delta_M\}$ as a set of masked tokens, $X_V = \{x_i | i \notin \delta_M\}$ as a set of visible image token, $A \in \mathbb{R}^{|\delta_M| \times (N - |\delta_M|)}$ is the affinity matrix that represents the probabilistic similarity between X_M and X_V, and ψ is the scaling function, *i.e.* row-wise softmax, that scales logits into a probability distribution relative to the visible image token.

3.2 Heterogeneity Analysis

We assume that it is necessary for masked tokens and visible tokens to exhibit substantial variability in terms of their distinct data properties in the initial embedding prior to being processed by the attention layers. This distinctiveness between masked tokens and visible tokens is instrumental in enhancing the model's ability to differentiate between the two tasks, owing to their distinct roles in the architecture; the former serves the purpose of pretext prediction



(a) Analysis of pre-trained models (b) Analysis of model convergence

Fig. 1: To investigate the heterogeneity between masked token and visible token, we analyze the pre-trained models of the recent approaches [13, 38]. (a) shows that the heterogeneity between two distinct types of tokens is highest on the initial embedding for both approaches, and it gradually decreases in subsequent layers. Unlike the pre-trained model, the heterogeneity of the non-converged SimMIM [38] model shown in (b) displays an erratic trend, indicating that the tendency of heterogeneity is acquired through model convergence.

while the latter aids in feature encoding and decoding. On the other hand, in subsequent layers, the masked tokens are gradually recovered by the neighboring visible tokens and will exhibit a heightened correlation with them.

Our investigation centered on determining whether models that demonstrate effective convergence uphold these hypotheses. To this end, we analyze the pre-trained models of the recent approaches [13,38] from two facets: 1) Heterogeneity analysis on pre-trained models of different methods and 2) heterogeneity analysis of converged and non-converged models.

Heterogeneity analysis on pre-trained models of different methods The heterogeneity between the masked and visible tokens across every layer of two pre-trained models, MAE and SimMIM [13,38] utilizing ViT [11] as a backbone is shown in Figure 1 (a). Note that 'layer depth 0' refers to the initial embedding stage before the masked token is fed into the attention layer as an input. Figure 1 (a) shows that the heterogeneity between two distinct types of tokens is highest on the initial embedding for both approaches, and it gradually decreases in subsequent layers. The high heterogeneity in the initial embedding phase reflects the data distinctiveness of the masked tokens, whereas the masked tokens are reconstructed to resemble the visible tokens, leading to reduced heterogeneity in subsequent layers.

Furthermore, Figure 2 presents the affinity map between every token pair for each layer of the pre-trained model [38]. Affinity maps are listed in order from initial embedding to subsequent layers, and we used min-max normalization for visualization. In the interest of understanding, the x-axis and y-axis of the affinity map are both arranged in the order of masked token X_M and visible image token X_V. Thus, we divided the affinity map into quadrants. Note that, the heterogeneity H is defined in the second quadrant as it corresponds to the mutual dependence of masked tokens with respect to visible tokens.

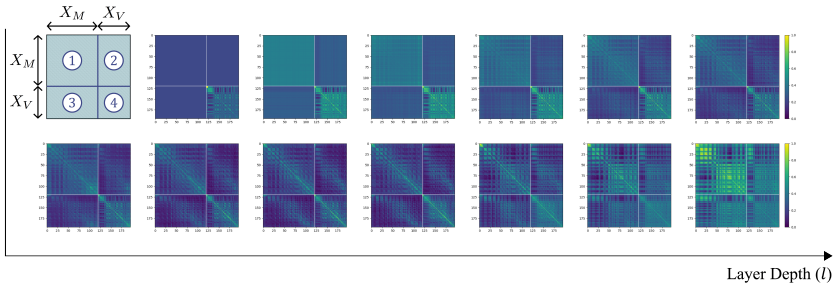


Fig. 2: We present the affinity map between every token pair for each layer of the pre-trained model [38]. Affinity maps are listed in order from initial embedding to subsequent layers, and the x-axis and y-axis of the affinity map are both arranged in the order of masked token X_M and the visible image token X_V. Note that, the layer depth increases from top-left to bottom-right. Min-max normalization was used for the visualization of the affinity maps.

In the first quadrant, the masked token, initially a singular parameter in the initial embedding, resembles the visible token in the subsequent layers, resulting in various correlation values. Within the second and third quadrants, the relationship between the masked token and the visible token exhibits a relatively low correlation during the initial layers. However, as the masked token undergoes the reconstruction process, increasingly higher correlations materialize in the subsequent layers. Finally, the correlation between the visible tokens in the fourth quadrant is relatively constant regardless of layer depth. In conclusion, the result of the affinity map qualitatively validates the orthogonality between the masked token and the visible token in the initial embedding while revealing a progressive enhancement in their similarity throughout the subsequent layers. In line with the above hypothesis, the heterogeneity from the initial input to the subsequent layers shows a distinct decrease in the overall results of Figure 1 and Figure 2.

Heterogeneity analysis of converged and non-converged models The heterogeneity of the converged model ('After Convergence') and the non-converged model at the early stage ('Before Convergence') of SimMIM [38] is shown in Figure 1 (b). Unlike the converged model, where the heterogeneity steadily decreases, the heterogeneity of the non-converged model displays an erratic trend, lacking any discernible pattern or structure. The results indicate that a model lacking a distinct inclination towards heterogeneity at the beginning of training achieves the desired attributes of the masked token through subsequent convergence.

4 Masked Token Optimization

Our endeavor lies in mitigating the issue of the prolonged pre-training phase by imposing explicit constraints on the optimization of masked tokens.

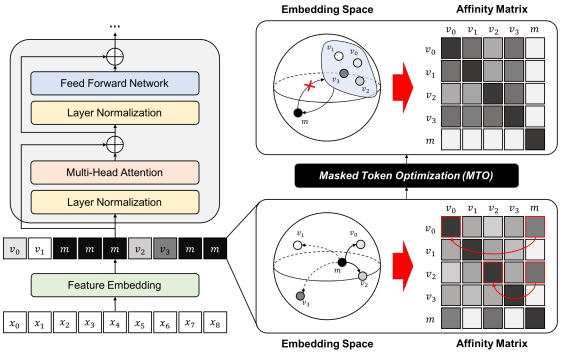


Fig. 3: The proposed Masked Token Optimization (MTO) approach encompasses the selective exclusion of semantically inconsequential masked tokens from the weight aggregation process pertaining to visible tokens with (3), and at the same time, it enforces data distinctiveness constraints (4) and (5) based on the depth of the layer to enhance the model's capability to accurately identify regions necessitating semantic restoration.

In the initial embedding, the parameter designated for masked tokens is identical across all corrupted tokens, with their respective values being set via a random initialization process. In this context, the *semantic voidness* within masked tokens exerts a negative impact on the process of learning features of visible tokens, impeding the overall efficacy of the representation learning. Therefore, we propose an explicit optimization in the initial embedding stage that allows the masked token to be reconstructed by being influenced by the visible token, but conversely, constrains the visible token from being affected by the masked tokens. This is achievable through the integration of a sparsity-inducing constraining term directly into the weight-learning mechanism of the affinity matrix between masked tokens and visible tokens. As shown in Figure 2, the x-axis and y-axis of the affinity map are both arranged in the order of masked token X_M and visible image token X_V, which allows the affinity map to be divided into four quadrants. Concretely, we propose to explore intuitive per-row sparsities within the third and fourth quadrants of the matrix as they correspond to the reciprocal dependencies between image tokens in relation to masked tokens and between image tokens themselves, respectively. The following constraint recalibrates the weight distribution between visible and masked tokens on a row-specific basis, increasing the weight between visible-visible tokens in comparison to the weight assigned to visible-masked token interactions:

$$\mathcal{L}_{spa}(f(I;\Theta)) = -\sum_{i \notin \delta_M} \sum_{j} (p_{i,j} \log p_{i,j}) \tag{3}$$

where p is an element of affinity matrix $\psi(XX^\top)$ that satisfies $0 < p_{i,j} < 1$ and $\sum_j p_{i,j} = 1$.

Minimizing the $\mathcal{L}_{spa}$ loss on a row-wise basis ensures the preferential allocation of maximum weight amongst the interaction of visible and visible tokens rather than the interaction of visible and masked tokens, which facilitates the

exclusion of semantically inconsequential masked tokens from the weight aggregation process of visible tokens. This assertion can be substantiated through the proof in Section 5 of the Supplementary material. The proof explicates the underlying operational principle of (3), elucidating that achieving minimum entropy per row is contingent upon the consistent assignment of maximum weight exclusively to interactions between visible tokens. This strategic approach plays a pivotal role in effectively preventing the influence of masked token values on the representation learning of visible tokens, thereby upholding the integrity of the learning algorithm.

Furthermore, as investigated in Section 3, the parameter pertaining to the masked token of the initial embedding is trained to exhibit a diminutive correlation with the visible token to fulfill the property of data distinctiveness. In existing methods, this property can solely be achieved by means of the model's convergence, which is secured through a long pre-training procedure. From the perspective of the distinctiveness of the masked token in the initial embedding, we propose to explicitly augment the heterogeneity from the visible token rather than solely relying on model convergence. By distinctly differentiating the masked token from the visible token, the network gains the capability to accurately identify regions necessitating semantic restoration, thereby paving the way for a more efficient learning process. The optimization for the initial masked token embedding can be formulated as:

$$\mathcal{L}_e(f(I;\Theta)) = \frac{1}{H^0 + \epsilon}, \tag{4}$$

where H^0 denotes heterogeneity defined in (2) of the initial embedding stage before passing through the attention layers and ϵ is a small value to prevent zero division. Eq. (4) augments the distinctiveness of masked tokens by maximizing the heterogeneity of masked tokens over visible tokens.

On the other hand, masked tokens in the subsequent layers tend to exhibit a notable correlation towards the visible tokens as they are gradually reconstructed through interaction with neighboring tokens in the attention layers. In light of this, with regard to the distinctiveness among tokens, we impose a constraint on the subsequent layers to have progressively lower heterogeneity. Considering the common direction of both aspects, we pursue a gradual reduction of heterogeneity. To this end, we intuitively employ the form of a ranking loss, strategically applied to the subsequent layers:

$$\mathcal{L}_r(f(I;\Theta)) = \sum_{l=1}^{L} \log(1 + \exp(H^{l-1} - H^l)). \tag{5}$$

By means of Eq. (5), we enforce a constraint upon the masked tokens in the subsequent layers, forcing them to exhibit diminished distinguishability from the visible tokens. Concurrently, we grant masked tokens the capability to exert an enhanced influence over the feature-learning process of the visible tokens. This intricate interplay of masked tokens across entire layers strikes a balance, promoting the convergence of token representations while employing only essential

Algorithm 1 Masked Token Optimization (MTO)

1: **Initialize:**
$f(I;\Theta) \leftarrow$ initial embedding of I with parameters Θ
2: **Recalibrate Weights:**
Minimize a sparsity-inducing constraint on the affinity matrix to exclude semantically inconsequential masked tokens. $L_{\text{spa}}(f(I;\Theta)) = -\sum_{i \notin \delta_M} \sum_j (p_{i,j} \log p_{i,j})$
3: **Enhance Data Distinctiveness:**
Maximize the heterogeneity of masked tokens in the initial embedding: $L_e(f(I;\Theta)) = \frac{1}{H_0+\epsilon}$
4: **Progressive Refinement:**
Ensure gradual reduction of heterogeneity in subsequent layers using ranking loss: $L_r(f(I;\Theta)) = \sum_{l=1}^{L} \log(1 + \exp(H^{l-1} - H^l))$
5: **Final Objective:** Combine the losses for final optimization:
$L_{\text{total}} = L_{\text{spa}} + L_e + L_r$
6: **Update:** Update the parameters Θ to minimize L_{total}

information within the learning framework. Finally, our method is distilled into the steps outlined in Algorithm 1.

5 Experiments

In this section, we assess the efficacy of the proposed MTO approach through a series of pre-training and fine-tuning experiments. As MTO is an adaptable and plug-and-play method for any Masked Image Modeling (MIM)-based approach that utilizes masked tokens, we apply it to multiple baseline approaches [10, 13, 38, 40] to evaluate the effectiveness. Please refer to the Supplementary material for more experimental results, detailed analysis, and ablation studies.

5.1 Metric for Efficient Pre-training

The main objective of MTO is to reduce the substantial pre-training time of Transformer-based architecture, that is to say, accelerating the convergence speed. The area under the curve can be one of the metrics that quantify the rate of convergence because the faster the network converges and the better the network performance, the higher the value. Thus, to quantify the relative performance improvement over the baseline approaches, we propose the RAUC measure, denoting the relative area under the curve, as follows:

$$RAUC(S1, S2; E1, E2) = \frac{\int_{E1}^{E2} (S2(E) - S1(E1))dE}{\int_{E1}^{E2} (S1(E) - S1(E1))dE} \tag{6}$$

Here, $E1$ and $E2$ represent the number of epochs, and $S(E)$ is set to the performance of the target method S at specific epoch E. This measure serves as a quantitative indicator, precisely delineating the extent of relative performance improvement across a specified range of epochs.

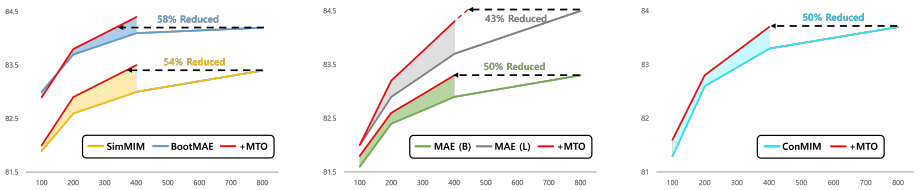


Fig. 4: The comprehensive performance results of applying MTO to various baselines [10, 13, 38, 40]. MTO achieves a substantial improvement in the efficiency of pre-training by attaining the standard performance within approximately 400 epochs across all baseline methods in common. This signifies that remarkable enhancement in efficiency is achievable across any MIM method through the application of MTO, rendering it a viable option for masked tokens.

5.2 Baseline Models

SimMIM [38] models masked image reconstruction as a pretext task for self-supervised pre-training of Transformer architecture. In SimMIM, masked tokens are semantically encoded with visible image tokens in the Transformer encoder and are reconstructed with shallow MLP layers. We pre-train the ViT-B following the same hyper-parameters as [38] and our losses are additionally adopted to the Transformer encoder.
MAE [13]. Different from SimMIM [38], only visible image tokens pass through the Transformer encoder for efficient pre-training. Thus, masked tokens are concatenated with encoded visible tokens and pass through the Transformer decoder, separating the semantic encoding task from the pretext prediction task. In MAE, We pre-train the ViT-B and ViT-L following the same hyper-parameters as [13] and losses are adopted to the Transformer decoder.
BootMAE [10] introduced bootstrapped MAE that combines a bootstrapped feature prediction task into the original MAE. BootMAE learns separate decoders for pixel regression and feature prediction with the same masking strategy as MAE. Thus, we adopted our masked token optimization strategy for both pixel regression and feature prediction Transformer decoders.
ConMIM [40] taps into the significant potential of contrastive learning within denoising auto-encoding frameworks. It focuses on generating straightforward intra-image inter-patch contrastive constraints as the primary learning goals for predicting masked patches. This approach eliminates the need for additional training stages often required in customizing image tokenizers.

5.3 Performance Comparisons

We report the results of the proposed method trained on the recent baselines [10, 13, 38, 40] in Figure 4 and Table 1. Following the same settings for each baseline method, we pre-trained and fine-tuned on ImageNet-1K classification dataset for main evaluation. Note that, training recent approaches require a huge hardware specification, making it increasingly difficult to reproduce the reported results. As all experiments were conducted in our hardware configuration (8×

$RAUC(S1, S2; E1, E2)$			
$S1$	$S2$	$(E1, E2)$ (100, 400)	(200, 400)
SimMIM [38]	SimMIM + MTO	1.47	1.44
BootMAE [10]	BootMAE + MTO	1.19	1.22
MAE [13]	MAE + MTO	1.32	1.29
ConMIM [40]	ConMIM + MTO	1.21	1.17

Table 1: We report the evaluation of the proposed relative area under the curve ($RAUC$) measure over the baseline approaches [10, 13, 38, 40]. The same backbone network (ViT-B) is used for pre-training.

	Method	Backbone	Epoch 400	Epoch 800	$RAUC$ (S1, S2; 400, 800)
S1	MAE [13]	ViT-L	68.5	72.7	1.20
S2	MAE + MTO		72.8	74.1	
S1	MAE [13]	ViT-B	57.4	63.9	1.23
S2	MAE + MTO		63.6	64.8	
S1	BootMAE [10]	ViT-B	63.8	65.4	1.22
S2	BootMAE + MTO		66.1	66.8	
S1	ConMIM [40]	ViT-B	32.2	39.2	1.20
S2	ConMIM + MTO		38.7	40.3	

Table 2: We report the linear evaluation accuracy as a means to assess the capacity of pre-trained representations to capture relevant features and demonstrate their applicability to specific tasks.

Proposed optimization			Top-1 Acc(%)
$\mathcal{L}_{spa}$	$\mathcal{L}_e$	$\mathcal{L}_r$	
✓			83.2
	✓		83.3
		✓	82.8
✓	✓		83.4
✓		✓	83.0
	✓	✓	83.4
✓	✓	✓	83.5

Table 3: We present the detailed ablation study conducted for the importance of the proposed objectives.

RTX 3090) for fair comparison, the results we reported may differ from those of their manuscript. For the fair experimental schedule, all the 400 epoch performances were equally measured in intermediate stages in the training towards 800 epochs.

Figure 4 presents a main comparison of the top-1 accuracy between the baseline models of SimMIM [38], MAE [13], BootMAE [10], ConMIM [40] and our method with the MTO applied. We conducted experiments on ViT-B [11] and ViT-L as a backbone attention network.

Through the application of MTO, the convergence process was significantly accelerated in all baseline methods, with the baseline's standard performance be-

ing achieved within the range of 300 to 500 epochs. Specifically, in the case of the SimMIM [38] and BootMAE [10] baseline, the application of MTO remarkably expedited performance, achieving in merely 300 to 400 epochs what typically requires 800 epochs, thereby realizing an impressive pre-training reduction rate of 54% and 58%. For both MAE [13] and ConMIM [40], the performances of the existing 800-epoch baseline were attained remarkably within just 400 epochs with the application of MTO. Our approach facilitated a substantial pre-training reduction rate of 50% for both methods. Moreover, even in the MAE [13] baseline utilizing the data-hungry model ViT-L, the introduction of MTO yielded impressive outcomes, manifesting in a pre-training epoch reduction rate of 43%.

The deployment of the proposed methodology across a range of baselines unequivocally validated the improvement of pre-training efficiency attributed to the implementation of MTO. This advancement stemmed from the effective training of masked tokens, a theme emphasized consistently throughout the manuscript, leading to a marked enhancement in the overall efficacy of pretext prediction and the refinement of representation learning.

RAUC: Table 1 reports the evaluation using the proposed relative area under the curve (*RAUC*) measure over the baseline approaches [10, 13, 38, 40]. This introduced metric effectively highlights the relative performance enhancements, offering a clear and immediate understanding upon initial observation. Across the range from 200 to 400 epochs, the relative performance improvement rates for SimMIM [38], BootMAE [10], MAE [13], and ConMIM [40] are 44%, 22%, 29% and 17%, respectively. Besides, in the overall range from 100 to 400 epochs, the relative performance increase rate was amplified further, showing rates of 47%, 19%, 32%, and 21% for each baseline. The result accentuates the diverse degrees of enhancement achieved by applying MTO to each method. Moreover, it confirms MTO's efficacy in reducing the pre-training epochs across all baselines, encompassing the full spectrum of the training procedure.

Furthermore, we report the linear evaluation accuracy in Table 2 as a means to assess the capacity of pre-trained representations to capture relevant features and demonstrate their applicability to specific tasks. Upon the application of MTO, the performance achieved at 400 epochs highly surpasses the performance achieved at the same epoch by all the baseline methods [10, 13, 40], manifesting an acceleration in the convergence process. Especially for MAE, the application of MTO demonstrated superior efficiency with both ViT-B and ViT-L architectures, highlighting the method's adaptability and effectiveness across different architectures.

Table 2 additionally reports the evaluation using the proposed relative area under the curve (RAUC) measure over the baseline approaches [13,38] by setting the baseline method as S1 and the MTO application as S2, using the equation (8). Within the range of 400 to 800 epochs, MTO exhibits a substantial relative performance improvements of 20% overall. The comprehensive results in Table 2 reveal that the MTO method significantly enhances performance and accelerates convergence in methods employing masked tokens. This broad efficacy suggests

that MTO exhibits excellent generalizability across a range of masked image modeling methods.

5.4 Ablation on Objectives

In our approach, we introduce three novel objectives, denoted as '$\mathcal{L}_{spa}$', '$\mathcal{L}_e$' and '$\mathcal{L}_r$' specifically designed for masked token optimization. Table 3 presents the detailed ablation study conducted for the importance of these objectives. All experiments report the ImageNet-1K classification accuracy on 400 epochs of SimMIM [38] using ViT-B as a backbone architecture.

Broadly speaking, each objective contributed notably to enhancing the performance and expediting the convergence process. Upon delving into the specifics, it becomes apparent that $\mathcal{L}_r$, when utilized in isolation, emerged as a factor contributing to the destabilization of performance outcomes. This phenomenon arises due to the fact that ranking loss merely modulates the magnitude of heterogeneity for each layer sequentially. Such a singular approach carries the risk of systemic collapse, potentially leading to scenarios where all heterogeneity converges to zero, thus undermining the model's structural integrity. Nevertheless, the simultaneous application of $\mathcal{L}_r$ with $\mathcal{L}_e$ leads to a harmonized effect. The entropy maximization impact of $\mathcal{L}_e$ at the first layer acts as a regulatory mechanism, effectively elevating the overall performance to a notable 83.4. Moreover, both $\mathcal{L}_{spa}$ and $\mathcal{L}_e$ demonstrate meaningful importance within the overall methodology. The sole application of each led to favourable enhancements in overall performance, elevating it to 83.2 and 83.3, respectively, underscoring their individual efficacy in the process. Consequently, the synergistic integration of all three optimizations emerged as a requisite for achieving the zenith of convergence acceleration and performance enhancement, highlighting the necessity of their collective implementation for optimal results.

6 Related Work

6.1 Masked Language Modeling

Masked Language Modeling (MLM) [2, 7–9, 12, 22, 29–31, 41] predicts removed tokens based on remained ones to inject the ability of learning semantic representation of a corpus to the network. While MLM has brought rapid advances in natural language processing (NLP) and have been shown to scale and generalize well on downstream tasks [3], the problem of prelonged convergence time and immense computation of naive MLM still remained. Amid these challenges, more efficient self-supervised pre-training approaches [6, 17, 18, 20, 34] have been proposed. ALBERT [17] proposes two parameter reduction techniques, factorized embedding parameterization and cross-layer parameter sharing, for memory efficiency and shortening the training time. Based on the Lottery Ticket Hypothesis (LTH), EaryBERT [6] prunes the network for efficient pre-training and fine-tuning. CCM [18] designs a curriculum masking framework that gradually

masks similar tokens of similar concepts in an easy-to-difficult order. Similar to MAE [13], 3ML [20] learns the encoder by separating the mask tokens from the sequence and conducts reconstruction only through the decoder.

6.2 Masked Image Modeling

Masked Image Modeling (MIM) [1, 4, 5, 10, 13–15, 21, 25–27, 36, 38, 39, 43–45] is a relatively new technique that has gained popularity in the field of computer vision and machine learning in recent years. The basic idea behind MIM is to predict missing or occluded parts of an image using a neural network trained on partially masked images. Inspired by NLP, iGPT [5] and iBERT [36] have attempted to transfer the pretext task of masked prediction from language to image data, but these have caught less attention due to their inferior performance to other approaches. Different from iBERT which directly reconstructs the masked patches, BEiT [1] uses a two-stage approach that requires a pre-trained discrete variational autoencoder (dVAE) to generate discretized target visual tokens. In contrast, MAE [13] and SimMIM [38] are end-to-end training methods of masked autoencoders. MAE predicts masked patches directly from unmasked ones with a high masking ratio of 75%. SimMIM has a similar structure to MAE but with a larger patch size and multiple masking strategies.

Despite the impressive performance, masked autoencoder approaches require a large amount of computation with large-scale training datasets. Researchers have explored using hierarchical Vision Transformers (ViTs) to improve the efficiency of pre-training models for masked image modeling by enabling the ViTs to discard masked patches and only operate on visible ones. GreenMIM [15] introduced group window attention, while HiViT [44] and MixMAE [21] enable masking in hierarchical ViT. In contrast to prior methodologies, the proposed method considers the inherent properties of the tokens employed by MIM as a fundamental approach to effective pre-training.

7 Conclusion

This work delves into the properties of masked tokens, examines their heterogeneity with visible tokens, and proposes a novel approach termed masked token optimization (MTO). MTO boosts both pretext prediction and semantic encoding by emphasizing data distinctiveness of the masked token, achieving a considerable improvement on pre-training efficiency. Also, our method can be applied to any method in a plug-and-play manner thanks to a simple approach that only adds loss functions.

Limitations Within the scope of this manuscript, the triad of properties attributed to masked tokens, are not to be deemed immutable, but rather dynamic concepts subject to evolution. With the continual progression in the realm of Masked Image Modeling, it becomes imperative that these attributes undergo constant updates and refinement. Embracing this process of perpetual revision and advancement is vital to remain abreast of the ever-evolving landscape of research in this specialized field.

References

1. Bao, H., Dong, L., Piao, S., Wei, F.: Beit: Bert pre-training of image transformers. In: International Conference on Learning Representations (2021)
2. Bao, H., Dong, L., Wei, F., Wang, W., Yang, N., Liu, X., Wang, Y., Gao, J., Piao, S., Zhou, M., et al.: Unilmv2: Pseudo-masked language models for unified language model pre-training. In: International conference on machine learning. pp. 642–652. PMLR (2020)
3. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. Advances in neural information processing systems **33**, 1877–1901 (2020)
4. Cao, S., Xu, P., Clifton, D.A.: How to understand masked autoencoders. arXiv preprint arXiv:2202.03670 (2022)
5. Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Luan, D., Sutskever, I.: Generative pretraining from pixels. In: International conference on machine learning. pp. 1691–1703. PMLR (2020)
6. Chen, X., Cheng, Y., Wang, S., Gan, Z., Wang, Z., Liu, J.: Earlybert: Efficient bert training via early-bird lottery tickets. arXiv preprint arXiv:2101.00063 (2020)
7. Clark, K., Luong, M.T., Le, Q.V., Manning, C.D.: Electra: Pre-training text encoders as discriminators rather than generators. arXiv preprint arXiv:2003.10555 (2020)
8. Conneau, A., Lample, G.: Cross-lingual language model pretraining. Advances in neural information processing systems **32** (2019)
9. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
10. Dong, X., Bao, J., Zhang, T., Chen, D., Zhang, W., Yuan, L., Chen, D., Wen, F., Yu, N.: Bootstrapped masked autoencoders for vision bert pretraining. In: European Conference on Computer Vision. pp. 247–264. Springer (2022)
11. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
12. Ghazvininejad, M., Levy, O., Liu, Y., Zettlemoyer, L.: Mask-predict: Parallel decoding of conditional masked language models. arXiv preprint arXiv:1904.09324 (2019)
13. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16000–16009 (2022)
14. Hou, Z., Sun, F., Chen, Y.K., Xie, Y., Kung, S.Y.: Milan: Masked image pretraining on language assisted representation. arXiv preprint arXiv:2208.06049 (2022)
15. Huang, L., You, S., Zheng, M., Wang, F., Qian, C., Yamasaki, T.: Green hierarchical vision transformer for masked image modeling. arXiv preprint arXiv:2205.13515 (2022)
16. Kakogeorgiou, I., Gidaris, S., Psomas, B., Avrithis, Y., Bursuc, A., Karantzalos, K., Komodakis, N.: What to hide from your students: Attention-guided masked image modeling. arXiv preprint arXiv:2203.12719 (2022)
17. Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., Soricut, R.: Albert: A lite bert for self-supervised learning of language representations. In: International Conference on Learning Representations

18. Lee, M., Park, J.H., Kim, J., Kim, K.M., Lee, S.: Efficient pre-training of masked language model via concept-based curriculum masking. In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (2022)
19. Li, X., Ge, Y., Yi, K., Hu, Z., Shan, Y., Duan, L.Y.: mc-beit: Multi-choice discretization for image bert pre-training. In: Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXX. pp. 231–246. Springer (2022)
20. Liao, B., Thulke, D., Hewavitharana, S., Ney, H., Monz, C.: Mask more and mask later: Efficient pre-training of masked language models by disentangling the [MASK] token". In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (2022)
21. Liu, J., Huang, X., Liu, Y., Li, H.: Mixmim: Mixed and masked image modeling for efficient visual representation learning. arXiv preprint arXiv:2205.13137 (2022)
22. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019)
23. Liu, Z., Hu, H., Lin, Y., Yao, Z., Xie, Z., Wei, Y., Ning, J., Cao, Y., Zhang, Z., Dong, L., et al.: Swin transformer v2: Scaling up capacity and resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12009–12019 (2022)
24. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10012–10022 (2021)
25. Pan, J., Zhou, P., Yan, S.: Towards understanding why mask-reconstruction pretraining helps in downstream tasks. arXiv preprint arXiv:2206.03826 (2022)
26. Peng, Z., Dong, L., Bao, H., Ye, Q., Wei, F.: Beit v2: Masked image modeling with vector-quantized visual tokenizers. arXiv preprint arXiv:2208.06366 (2022)
27. Peng, Z., Dong, L., Bao, H., Ye, Q., Wei, F.: A unified view of masked image modeling. arXiv preprint arXiv:2210.10615 (2022)
28. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al.: Improving language understanding by generative pre-training (2018)
29. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. The Journal of Machine Learning Research **21**(1), 5485–5551 (2020)
30. Song, K., Tan, X., Qin, T., Lu, J., Liu, T.Y.: Mass: Masked sequence to sequence pre-training for language generation. In: International Conference on Machine Learning. pp. 5926–5936. PMLR (2019)
31. Song, K., Tan, X., Qin, T., Lu, J., Liu, T.Y.: Mpnet: Masked and permuted pre-training for language understanding. Advances in Neural Information Processing Systems **33**, 16857–16867 (2020)
32. Taylor, R., Kardas, M., Cucurull, G., Scialom, T., Hartshorn, A., Saravia, E., Poulton, A., Kerkez, V., Stojnic, R.: Galactica: A large language model for science. arXiv preprint arXiv:2211.09085 (2022)
33. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
34. Wettig, A., Gao, T., Zhong, Z., Chen, D.: Should you mask 15% in masked language modeling? arXiv preprint arXiv:2202.08005 (2022)

35. Wettig, A., Gao, T., Zhong, Z., Chen, D.: Should you mask 15% in masked language modeling? In: Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics. pp. 2977–2992 (2023)
36. v. Wintzingerode, F., Göbel, U.B., Stackebrandt, E.: Determination of microbial diversity in environmental samples: pitfalls of pcr-based rrna analysis. FEMS microbiology reviews **21**(3), 213–229 (1997)
37. Wu, J., Mo, S.: Object-wise masked autoencoders for fast pre-training. arXiv preprint arXiv:2205.14338 (2022)
38. Xie, Z., Zhang, Z., Cao, Y., Lin, Y., Bao, J., Yao, Z., Dai, Q., Hu, H.: Simmim: A simple framework for masked image modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9653–9663 (2022)
39. Xue, H., Gao, P., Li, H., Qiao, Y., Sun, H., Li, H., Luo, J.: Stare at what you see: Masked image modeling without reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22732–22741 (2023)
40. Yi, K., Ge, Y., Li, X., Yang, S., Li, D., Wu, J., Shan, Y., Qie, X.: Masked image modeling with denoising contrast. arXiv preprint arXiv:2205.09616 (2022)
41. Zaken, E.B., Ravfogel, S., Goldberg, Y.: Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. arXiv preprint arXiv:2106.10199 (2021)
42. Zhang, C., Zhang, C., Song, J., Yi, J.S.K., Zhang, K., Kweon, I.S.: A survey on masked autoencoder for self-supervised learning in vision and beyond. arXiv preprint arXiv:2208.00173 (2022)
43. Zhang, Q., Wang, Y., Wang, Y.: How mask matters: Towards theoretical understandings of masked autoencoders. arXiv preprint arXiv:2210.08344 (2022)
44. Zhang, X., Tian, Y., Xie, L., Huang, W., Dai, Q., Ye, Q., Tian, Q.: Hivit: A simpler and more efficient design of hierarchical vision transformer. In: The Eleventh International Conference on Learning Representations (2023)
45. Zhou, J., Wei, C., Wang, H., Shen, W., Xie, C., Yuille, A., Kong, T.: ibot: Image bert pre-training with online tokenizer. arXiv preprint arXiv:2111.07832 (2021)

Emerging Property of Masked Token for Effective Pre-training - Supplementary material

Hyesong Choi[1], Hunsang Lee[2], Seyoung Joung[1], Hyejin Park[1], Jiyeong Kim[1], and Dongbo Min[1†]

[1] Ewha W. University
[2] Hyundai Motor Company

In this supplementary document, we present more comprehensive results and in-depth details not provided in the original manuscript owing to page constraints.

We include the following material:

1 Evaluation on Transfer Learning

We further conducted an extensive evaluation of the transfer learning performance of pre-trained models utilizing the proposed MTO by examining their effectiveness in various downstream tasks, including semantic segmentation, object detection, and instance segmentation.

Semantic segmentation: Table 1 presents a comprehensive analysis of the transfer learning performance on ADE20K [16], highlighting a comparative evaluation between the application of our proposed MTO approach and the baseline methods [3, 6, 15] using ViT-B as the backbone. The application of MTO resulted in a remarkable improvement in the training efficiency of the semantic segmentation task, where training for just 400 epochs on both baselines exhibited a performance comparable to that achieved on 800 epochs of the baseline

Method	Epoch	mIoU
MAE [6]	800	47.5
MAE + MTO	400	44.1
MAE + MTO	800	48.4
BootMAE [3]	800	49.1
BootMAE + MTO	400	49.4
BootMAE + MTO	800	50.3
ConMIM [15]	800	49.8
ConMIM + MTO	400	49.1
ConMIM + MTO	800	50.2

Table 1: Our study includes an in-depth evaluation of transfer learning on ADE20K [16], comparing the efficiency of our MTO approach with baseline methods [3, 6, 15], all utilizing ViT-B as the backbone. Implementing MTO significantly enhanced the training efficiency in the semantic segmentation task. Remarkably, a 400-epoch training period using MTO on both baselines matched the performance level of the 800-epoch training period using the baseline methods.

Method	Epoch	AP^{bb}	AP^{mk}
MAE [6]	800	46.9	41.6
MAE + MTO	400	46.5	40.8
MAE + MTO	800	47.2	41.5
BootMAE [3]	800	48.5	43.4
BootMAE + MTO	400	48.4	43.1
BootMAE + MTO	800	49.1	43.4
ConMIM [15]	800	47.8	42.5
ConMIM + MTO	400	47.6	42.2
ConMIM + MTO	800	48.8	42.9

Table 2: Our study involved a detailed assessment of transfer learning on COCO [7], focusing on the effectiveness of the MTO approach relative to existing methods [3, 6, 15], all using the ViT-B architecture. The results of these experiments demonstrated a notable improvement in transfer learning in both the AP^{bb} and AP^{mk} measures, through the application of MTO. These findings robustly affirm the significant influence and efficiency of MTO in enhancing training processes and elevating the transferability of pre-trained models.

method. Furthermore, it is worth noting that the performance at 800 epochs with the proposed MTO exceeded the performance achieved on 800 epochs of all

the baseline methods, demonstrating a significant improvement in overall performance. This outcome signifies that an adept configuration of masked tokens not only catalyzes rapid convergence but also aids in the acquisition of vital features that exhibit a high degree of generalizability.

Object detection and instance segmentation: Table 2 shows an evaluation of the transfer learning performance on COCO [7], meticulously examining the impact of our MTO approach in comparison to the baseline methods [3,6,15] using ViT-B as the backbone. The experimental results on BootMAE [3] and ConMIM [15] showcased a remarkable enhancement in transfer learning performance by harnessing the power of MTO in both the AP^{bb} and AP^{mk} measures. This substantiates the compelling impact and effectiveness of MTO in facilitating superior training efficiency and improving the overall transferability of pre-trained models. However, in the MAE [6] experiment, the application of MTO posed challenges in improving the AP^{mk} measure, thereby indicating the marginal improvement of final converged performance. Nevertheless, we successfully accomplished our objective of enhancing training efficiency on the AP^{bb} measure of MAE. Overall, MTO not only augmented the training efficiency across all baseline methods but also enhanced their converged performance in the final 800 epochs. These favourable achievements underscores the effectiveness of the proposed masked token optimization in enhancing the overall efficiency of the training process of transfer learning.

2 Evaluation on Different Architectures

In the realm of masked image modeling, two distinct methodologies have garnered significant attention. In the first approach, masked tokens are employed at the encoder level. This method involves selectively masking certain image segments before the encoding process, where the visible tokens and masked tokens are encoded simultaneously. By doing so, the encoder is compelled to develop a more nuanced understanding of the image context and the relationships between visible and masked tokens, thereby improving the model's prediction accuracy and feature extraction capabilities. This encoder-centric method of utilizing masked tokens is also referred to as the 'Inpainting-style' [14] approach. Notable studies of this methodology include exemplars such as BEiT [1], SimMIM [13], ConMIM [15], MaskFeat [11], MVP [12] and BEiT V2 [10].

Conversely, the second approach in masked image modeling applies masked tokens within the decoder framework. This technique exclusively utilizes visible tokens as the input for the encoder, followed by the application of a multi-layer Transformer. This architecture is designed to decode the masked features, strategically incorporating the use of masked tokens prior to initiating the decoding process. Also termed as the 'Decoder-style' method [14], this approach is distinguished by its relatively superior linear probing accuracy, a notable advantage over the inpainting-style technique. Representative studies of this approach include MAE [6], CAE [2], MCMAE [5], and BootMAE [3].

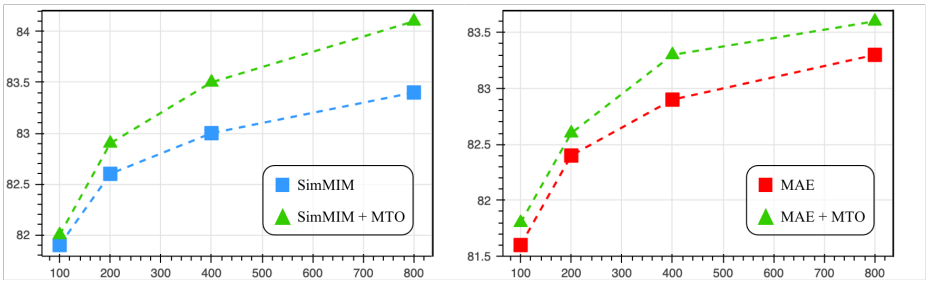


Fig. 1: The implementation of MTO in two different MIM architectures showed varying results. MTO successfully facilitated rapid convergence in both methodologies, extending up to 400 epochs. However, at the 800 epoch juncture, the efficacy of MTO becomes strikingly evident with SimMIM [13], an inpainting-style method, demonstrating an overwhelming superiority in performance over MAE [6], which adheres to the decoder style approach.

The result derived from implementing MTO within these two distinct MIM methodologies revealed variances. As shown in Figure 1, MTO successfully facilitated rapid convergence in both methodologies, extending up to 400 epochs. However, at the 800 epoch juncture, the efficacy of MTO becomes strikingly evident with SimMIM [13], an inpainting-style method, demonstrating an overwhelming superiority in performance over MAE [6], which adheres to the decoder style approach.

The root of this outcome can be traced back to the specific application locus of the proposed masked token optimization method. The concurrent occurrence of pretext prediction and representation learning within the inpainting-style method provides fertile ground for the MTO method to enhance representation learning performance, showcasing a synergistic interplay that elevates the overall efficacy of the process. However, within the decoder-style methodologies, the MTO method engages less intrusively in the depths of the representation learning process, maintaining a relatively peripheral influence compared to its impact in the inpainting-style approaches. Hence, while the MTO method universally enhances convergence speed and overall performance across all masked token methodologies, its zenith of potential is most prominently realized when applied to the inpainting style method, where its capabilities are optimally harnessed.

3 Ablation on Decoder Depth

We conducted a comprehensive investigation into the influence of layer depth on the performance of MTO. In essence, our study delved into the varying degrees of optimization efficacy exhibited by MTO, contingent upon the number of layers dedicated to pretext prediction under decoder conditions where feature encoding is absent, thus providing insights into the performance dynamics of MTO. Figure 2 presents a detailed ablation study on the number of decoder

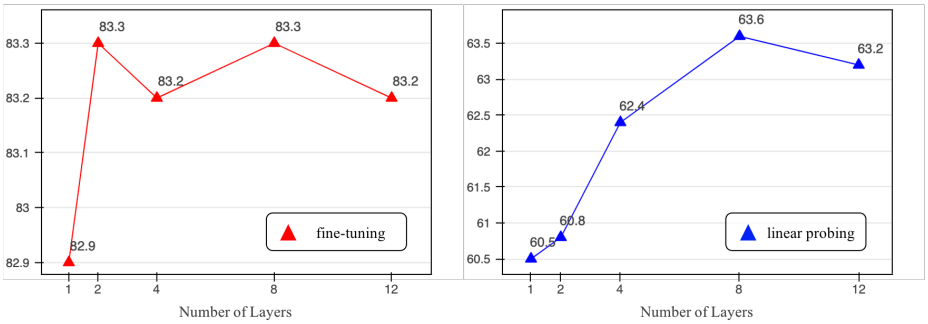


Fig. 2: We conducted a comprehensive study that delves into the varying degrees of optimization efficacy exhibited by MTO, contingent upon the number of layers dedicated to pretext prediction under decoder conditions where feature encoding is absent, thus providing insights into the performance dynamics of MTO. We report the ImageNet-1K classification accuracy achieved by the MAE [6] model trained for 400 epochs using ViT-B as the backbone. Superior MTO optimization performance was achieved when an ample number of layers were used in both fine-tuning and linear probing performance.

config	value
optimizer	AdamW [9]
pre-training base learning rate	1e-4
pre-training weight decay	0.05
pre-training optimizer momentum	$\beta_1, \beta_2 = 0.9, 0.999$
pre-training batch size	2048
learning rate schedule	cosine decay [8]
pre-training warmup epochs	10
fine-tuning base learning rate	5e-3
fine-tuning weight decay	0.05
fine-tuning optimizer momentum	$\beta_1, \beta_2 = 0.9, 0.999$
layer-wise learning rate decay	0.9
fine-tuning batch size	2048
fine-tuning warmup epochs	10
fine-tuning training epochs	100

Table 3: Hyperparameters used for SimMIM [13] experiments. All configs follow the setting of [13].

blocks. All experiments report the ImageNet-1K classification accuracy achieved by the MAE model trained for 400 epochs using ViT-B as the backbone.

Overall, superior MTO optimization performance was achieved when an ample number of layers were used in both fine-tuning and linear probing performance. When an insufficient number of layers are employed, the rank objective

config	value
optimizer	AdamW [9]
pre-training base learning rate	1.5e-4
pre-training weight decay	0.05
pre-training optimizer momentum	$\beta_1, \beta_2 = 0.9, 0.95$
pre-training batch size	4096
learning rate schedule	cosine decay [8]
pre-training warmup epochs	40
fine-tuning base learning rate	1e-3
fine-tuning weight decay	0.05
fine-tuning optimizer momentum	$\beta_1, \beta_2 = 0.9, 0.999$
layer-wise learning rate decay	0.75
fine-tuning batch size	1024
fine-tuning warmup epochs	5
fine-tuning training epochs	100 (ViT-B), 50 (ViT-L)

Table 4: Hyperparameters used for MAE [6] experiments. All configs follow the setting of [6].

integral to this optimization method falters, leading to suboptimal results. This inadequacy is reflected in the diminished fine-tuning accuracy and linear probing performance, which were recorded at the lowermost figures of 82.9 and 60.5, respectively.

Fine-tuning accuracy reached its pinnacle when the number of layers was set to either 2 or 8. This outcome suggests that MTO operates effectively with a layer count as minimal as two, showcasing its versatility and efficiency in various configurations.

Conversely, linear probing accuracy consistently exhibited superior results when the layer count was extended to 8 or more, indicating a positive correlation between increased layer numbers and enhanced linear probing efficacy. This observation aligns seamlessly with the experimental findings of MAE [6], which highlight the necessity of a sufficiently deep decoder to attain high linear probing accuracy. It is stated that this requirement stems from the inherent difference between pixel reconstruction and recognition tasks. Nevertheless, the efficiency of MTO exhibited a slight diminution when the layer count was increased to as many as 12, suggesting a balance between layer quantity and optimization efficacy. In contrast to fine-tuning, linear probing displayed a pronounced variance in performance contingent on the layer count. Hence, to achieve optimal results in linear probing, it becomes imperative to meticulously calibrate the number of layers through experimental fine-tuning within the masked token optimization process.

config	value
optimizer	AdamW [9]
pre-training base learning rate	1.5e-4
pre-training weight decay	0.05
pre-training optimizer momentum	$\beta_1, \beta_2 = 0.9, 0.95$
pre-training batch size	4096
learning rate schedule	cosine decay [8]
pre-training warmup epochs	40
fine-tuning base learning rate	5e-3
fine-tuning weight decay	0.05
fine-tuning optimizer momentum	$\beta_1, \beta_2 = 0.9, 0.999$
layer-wise learning rate decay	0.75
fine-tuning batch size	1024
fine-tuning warmup epochs	20
fine-tuning training epochs	100

Table 5: Hyperparameters used for BootMAE [3] experiments. All configs follow the setting of [3].

4 Implementation Details

We list configurations for each baseline approach [3,6,13] used in the experiments of the original manuscript. The hyperparameters are categorically segregated into two distinct domains: pre-training and fine-tuning, each representing a specific aspect of the model's training process. Note that we followed the settings of each method for a fair comparison.

4.1 SimMIM Experiments

Table 3 presents the hyperparameters utilized in the experimental setup of SimMIM [13], encompassing key configurations and settings employed throughout the study. In the SimMIM experiments, we adopt ViT-B [4] as the default backbone to align with the setting of the baseline approach.

4.2 MAE Experiments

Table 4 provides an overview of the hyperparameters employed in the experimental setup of MAE [6], offering a comprehensive insight into the crucial configurations and settings of the study. In an effort to evaluate scalability while maintaining fidelity to the fundamental approach, we strategically employ ViT-B and ViT-L as the backbones in the MAE experiments.

config	value
optimizer	AdamW [9]
pre-training peak learning rate	5e-4
pre-training base learning rate	1e-5
pre-training weight decay	0.05
pre-training optimizer momentum	$\beta_1, \beta_2 = 0.9, 0.98$
pre-training batch size	2048
learning rate schedule	cosine decay [8]
pre-training warmup epochs	10
fine-tuning peak learning rate	1e-3, 2e-3, 3e-3, 4e-3, 5e-3
fine-tuning minimal learning rate	1e-6
fine-tuning weight decay	0.05
fine-tuning optimizer momentum	$\beta_1, \beta_2 = 0.9, 0.999$
layer-wise learning rate decay	0.65
fine-tuning batch size	1024
fine-tuning warmup epochs	20
fine-tuning training epochs	100

Table 6: Hyperparameters used for ConMIM [15] experiments. All configs follow the setting of [15].

4.3 BootMAE Experiments

The hyperparameters employed in the experiment of BootMAE [3] are presented in Table 5. We adopt ViT-B as the default backbone in the BootMAE experiments. By doing so, we ensure an environment of consistent and reliable evaluation throughout our study.

4.4 ConMIM Experiments

Table 6 outlines the hyperparameters used in the ConMIM [15] experiment's design, detailing essential configurations and settings. For the ConMIM experiments, ViT-B is chosen as the backbone encoder, maintaining consistency with the baseline approach's settings.

5 Full Proof for Equation (3) of manuscript

Definition 1. *Let (X, d) be a metric space, E be a subset of X and f a real-valued function with domain E. Suppose that p is a limit point of E. The function f has a* ***limit*** *at p if there exists a number $L \in \mathbb{R}$ such that given any $\varepsilon > 0$, there exists a $\delta > 0$ for which*

$$|f(x) - L| < \varepsilon \tag{1}$$

for all points $x \in E$ *satisfying* $0 < d(x, p) < \delta$*. If this is the case, we write*

$$\lim_{x \to p} f(x) = L \quad or \quad f(x) \to L \quad as \quad x \to p \tag{2}$$

Let $\mathbf{U} = \{x \,|\, x \in \mathbf{A}\} \cup \{y \,|\, y \in \mathbf{B}\}$ be a single row of the affinity map where the x-axis and y-axis are both arranged in the order of masked token X_M and visible image token X_V. Without loss of generality, we assume that $\mathbf{A} = \{p_1, p_2, \ldots, p_{n-1}, p_n\}$ and $\mathbf{B} = \{p_{n+1}, p_{n+2}, \ldots, p_{N-1}, p_N\}$ refers to the weights between the visible-masked tokens and visible-visible tokens, respectively. Here, $n = |\delta_M|$ and N indicate the number of masked tokens and the number of all tokens, respectively. The entropy of U is expressed as $\mathbb{E}_{\mathbf{U}} = -\sum_{l=1}^{N}(p_l \log p_l)$. It can be decomposed as $\mathbb{E}_{\mathbf{U}} = \mathbb{E}_{\mathbf{A}} + \mathbb{E}_{\mathbf{B}}$, where $\mathbb{E}_{\mathbf{A}} = -\sum_{l=1}^{n}(p_l \log p_l)$ and $\mathbb{E}_{\mathbf{B}} = -\sum_{l=n+1}^{N}(p_l \log p_l)$.

$$\mathbf{U} = \{x \,|\, x \in \mathbf{A}\} \cup \{y \,|\, y \in \mathbf{B}\}, \tag{3}$$

$$\mathbf{A} = \{p_1, p_2, \ldots, p_{n-1}, p_n\}, \tag{4}$$

$$\mathbf{B} = \{p_{n+1}, p_{n+2}, \ldots, p_{N-1}, p_N\}, \tag{5}$$

$$\mathbb{E}_{\mathbf{U}} = -\sum_{l=1}^{N}(p_l \log p_l), \tag{6}$$

$$\mathbb{E}_{\mathbf{A}} = -\sum_{l=1}^{n}(p_l \log p_l), \tag{7}$$

$$\mathbb{E}_{\mathbf{B}} = -\sum_{l=n+1}^{N}(p_l \log p_l) \tag{8}$$

$$\varepsilon > 0 \tag{9}$$

The following conditions apply:
Condition 1. $p_1 = p_2 = \ldots = p_{n-1} = p_n$
Condition 2. $\forall p \in \mathbf{A} : 0 < p < \frac{1}{n}$
Condition 3. if $\max p_i \in \mathbf{A} : \max p_i \approx \frac{1}{n}$,
if $\max p_i \in \mathbf{B} : \max p_i \approx 1$,
$\min p_j = \varepsilon \ (i \neq j,\ 1 \leq i \leq N,\ 1 \leq j \leq N)$

Proof. We proceed by induction on $n \geq 1$. When $\max p_i \in \mathbf{B}$, the value of $\mathbb{E}_{\mathbf{U}}$ is lower than when $\max p_i \in \mathbf{A}$. (except for $n = 1$)

Step 1: if $n = 1$
$\mathbf{A} = \{p_1\}, \quad \mathbf{B} = \{p_2, p_3, \ldots, p_t, \ldots, p_{N-1}, p_N\}$

By **Condition 2.** $\forall p \in \mathbf{A} : 0 < p < 1$

i) Case 1
If $\max p_i \in \mathbf{A}$
By **Condition 3.** $p_1 \approx 1$
$p_2 = p_3 = \cdots = p_{N-1} = p_N = \varepsilon$

$$\begin{aligned}\mathbb{E}_{\mathbf{U}} &= \mathbb{E}_{\mathbf{A}} + \mathbb{E}_{\mathbf{B}} \\ &= -p_1 \log p_1 + \sum_{l=2}^{N} (-p_l \log p_l) \\ &= \lim_{x \to 1-} (-x \log x) + \sum_{l=2}^{N} \lim_{y_l \to 0+} (-y_l \log y_l) \\ &\to 0 \quad (|\mathbb{E}_{\mathbf{U}} - 0| < \varepsilon, \textit{ which satisfy Definition 1. })\end{aligned} \tag{10}$$

ii) Case 2
If $\max p_i \in \mathbf{B}$
$p_t = \max p_i (2 \leq t \leq N)$, by **Condition 3.** $p_t \approx 1$
By **Condition 3.** $\forall p \in \mathbf{A} : p < p_t$
$p_t \approx 1 - p_1$
$\therefore p_1 = \varepsilon, \quad p_3 = p_4 = \cdots = p_{t-1} = p_{t+1} = \cdots$
$= p_{N-1} = p_N = \varepsilon$

$$\begin{aligned}\mathbb{E}_{\mathbf{U}} &= \mathbb{E}_{\mathbf{A}} + \mathbb{E}_{\mathbf{B}} \\ &= (-p_1 \log p_1) + \left\{ -p_t \log p_t + \sum_{l=2, l \neq t}^{N} (-p_l \log p_l) \right\} \\ &= \lim_{x \to 0+} (-x \log x) \\ &\quad + \left\{ \lim_{y \to 1-} (-y \log y) + \sum_{l=2, l \neq t}^{N} \lim_{z_l \to 0+} (-z_l \log z_l) \right\} \\ &\to 0 \quad (|\mathbb{E}_{\mathbf{U}} - 0| < \varepsilon, \textit{ which satisfy Definition 1. })\end{aligned} \tag{11}$$

It is impossible to compare which of $\mathbb{E}_{\mathbf{U}_{case1}}$ and $\mathbb{E}_{\mathbf{U}_{case2}}$ is higher.

Step 2: if $n = 2$

$\mathbf{A} = \{p_1, p_2\}, \quad \mathbf{B} = \{p_3, p_4, \ldots, p_t, \ldots, p_{N-1}, p_N\}$

By **Condition 2.** $\forall p \in \mathbf{A} : 0 < p < \frac{1}{2}$

i) Case 1
If $\max p_i \in \mathbf{A}$
By **Condition 1.** and **Condition 2.** $\forall p \in \mathbf{A} : p \approx \frac{1}{2}$
$p_3 = p_4 = \cdots = p_{N-1} = p_N = \varepsilon$

$$\begin{aligned}
\mathbb{E}_{\mathbf{U}} &= \mathbb{E}_{\mathbf{A}} + \mathbb{E}_{\mathbf{B}} \\
&= \sum_{l=1}^{2}(-p_l \log p_l) + \sum_{l=3}^{N}(-p_l \log p_l) \\
&= \sum_{l=1}^{2} \lim_{x_l \to \frac{1}{2}-}(-x_l \log x_l) + \sum_{l=3}^{N} \lim_{y_l \to 0+}(-y_l \log y_l) \\
&\to \log 2 \quad (|\mathbb{E}_{\mathbf{U}} - \log 2| < \varepsilon, \textit{ which satisfy Definition 1. })
\end{aligned} \tag{12}$$

ii) Case 2
If $\max p_i \in \mathbf{B}$
$p_t = \max p_i$ $(3 \le t \le N)$, by **Condition 3.** $p_t \approx 1$
By **Condition 3.** $\forall p \in \mathbf{A} : p < p_t$
$p_t \approx 1 - (p_1 + p_2)$
$\therefore p_1 = p_2 = \varepsilon, \quad p_3 = p_4 = \cdots = p_{t-1}$
$= p_{t+1} = \cdots = p_{N-1} = p_N = \varepsilon$

$$\begin{aligned}
\mathbb{E}_{\mathbf{U}} &= \mathbb{E}_{\mathbf{A}} + \mathbb{E}_{\mathbf{B}} \\
&= \sum_{l=1}^{2}(-p_l \log p_l) + \left\{ -p_t \log p_t + \sum_{l=3, l\neq t}^{N}(-p_l \log p_l) \right\} \\
&= \sum_{l=1}^{2} \lim_{x_l \to 0+}(-x_l \log x_l) \\
&\quad + \left\{ \lim_{y \to 1-}(-y \log y) + \sum_{l=3, l\neq t}^{N} \lim_{z_l \to 0+}(-z_l \log z_l) \right\} \\
&\to 0 \quad (|\mathbb{E}_{\mathbf{U}} - 0| < \varepsilon, \textit{ which satisfy Definition 1. })
\end{aligned} \tag{13}$$

Always $\quad \mathbb{E}_{\mathbf{U}_{case1}} > \mathbb{E}_{\mathbf{U}_{case2}}$

Step 3: if $n = k$

$\mathbf{A} = \{p_1, p_2, \ldots, p_{k-1}, p_k\}$,
$\mathbf{B} = \{p_{k+1}, p_{k+2}, \ldots, p_t, \ldots, p_{N-1}, p_N\}$

By **Condition 2.** $\forall p \in \mathbf{A} : 0 < p < \frac{1}{k}$

i) Case 1
If $\max p_i \in \mathbf{A}$
By **Condition 1.** and **Condition 2.** $\forall p \in \mathbf{A} : p \approx \frac{1}{k}$
$p_{k+1} = p_{k+2} = \cdots = p_{N-1} = p_N = \varepsilon$

$$\begin{aligned}
\mathbb{E}_{\mathbf{U}} &= \mathbb{E}_{\mathbf{A}} + \mathbb{E}_{\mathbf{B}} \\
&= \sum_{l=1}^{k}(-p_l \log p_l) + \sum_{l=k+1}^{N}(-p_l \log p_l) \\
&= \sum_{l=1}^{k} \lim_{x_l \to \frac{1}{k}-} (-x_l \log x_l) + \sum_{l=k+1}^{N} \lim_{y_l \to 0+} (-y_l \log y_l) \\
&\to \log k \quad (|\mathbb{E}_{\mathbf{U}} - \log k| < \varepsilon, \textit{ which satisfy Definition 1.})
\end{aligned} \tag{14}$$

ii) Case 2
If $\max p_i \in \mathbf{B}$
$p_t = \max p_i$ $(n+1 \le t \le N)$, by **Condition 3.** $p_t \approx 1$
By **Condition 3.** $\forall p \in \mathbf{A} : p < p_t$
$p_t \approx 1 - \sum_{l=1}^{k} p_l$
$\therefore p_1 = p_2 = \cdots = p_{k-1} = p_k = \varepsilon$,
$p_{k+1} = p_{k+2} = \cdots = p_{t-1} = p_{t+1} = \cdots$
$= p_{N-1} = p_N = \varepsilon$

$$\begin{aligned}
\mathbb{E}_{\mathbf{U}} &= \mathbb{E}_{\mathbf{A}} + \mathbb{E}_{\mathbf{B}} \\
&= \sum_{l=1}^{k}(-p_l \log p_l) \\
&\quad + \left\{ -p_t \log p_t + \sum_{l=k+1, l\neq t}^{N} (-p_l \log p_l) \right\} \\
&= \sum_{l=1}^{k} \lim_{x_l \to 0+} (-x_l \log x_l) \\
&\quad + \left\{ \lim_{y \to 1-} (-y \log y) + \sum_{l=k+1, l\neq t}^{N} \lim_{z_l \to 0+} (-z_l \log z_l) \right\} \\
&\to 0 \quad (|\mathbb{E}_{\mathbf{U}} - 0| < \varepsilon, \textit{ which satisfy Definition 1.})
\end{aligned} \tag{15}$$

Always $\mathbb{E}_{\mathbf{U}_{case1}} > \mathbb{E}_{\mathbf{U}_{case2}}$

Step 4: if $n = k + 1$

$\mathbf{A} = \{p_1, p_2, \ldots, p_k, p_{k+1}\}$,
$\mathbf{B} = \{p_{k+2}, p_{k+3}, \ldots, p_t, \ldots, p_{N-1}, p_N\}$

By **Condition 2.** $\forall p \in \mathbf{A} : 0 < p < \frac{1}{k+1}$

i) Case 1
If $\max p_i \in \mathbf{A}$
By **Condition 1.** and **Condition 2.** $\forall p \in \mathbf{A} : p \approx \frac{1}{k+1}$
$p_{k+2} = p_{k+3} = \cdots = p_{N-1} = p_N = \varepsilon$

$$
\begin{aligned}
\mathbb{E}_{\mathbf{U}} &= \mathbb{E}_{\mathbf{A}} + \mathbb{E}_{\mathbf{B}} \\
&= \left\{ \sum_{l=1}^{k} (-p_l \log p_l) + (-p_{k+1} \log p_{k+1}) \right\} \\
&\quad + \left\{ \sum_{l=k+1}^{N} (-p_l \log p_l) - (-p_{k+1} \log p_{k+1}) \right\} \\
&= \sum_{l=1}^{k+1} \lim_{x_l \to \frac{1}{k+1}-} (-x_l \log x_l) + \sum_{l=k+2}^{N} \lim_{y_l \to 0+} (-y_l \log y_l) \\
&\to \log (k+1) \\
&(|\mathbb{E}_{\mathbf{U}} - \log (k+1)| < \varepsilon, \textit{ which satisfy Definition 1.})
\end{aligned}
\tag{16}
$$

ii) Case 2
If $\max p_i \in \mathbf{B}$
$p_t = \max p_i$ $(n + 2 \leq t \leq N)$, by **Condition 3.** $p_t \approx 1$
By **Condition 3.** $\forall p \in \mathbf{A} : p < p_t$
$p_t \approx 1 - \sum_{l=1}^{k+1} p_l$
$\therefore p_1 = p_2 = \cdots = p_k = p_{k+1} = \varepsilon$,
$p_{k+2} = p_{k+3} = \cdots = p_{t-1} = p_{t+1} = \cdots$
$= p_{N-1} = p_N = \varepsilon$

$$
\begin{aligned}
\mathbb{E}_{\mathbf{U}} &= \mathbb{E}_{\mathbf{A}} + \mathbb{E}_{\mathbf{B}} \\
&= \left\{ \sum_{l=1}^{k} (-p_l \log p_l) + (-p_{k+1} \log p_{k+1}) \right\} \\
&\quad + \left\{ \begin{array}{l} (-p_t \log p_t) + \\ \displaystyle\sum_{l=k+1, l\neq t}^{N} (-p_l \log p_l) - (-p_{k+1} \log p_{k+1}) \end{array} \right\} \\
&= \sum_{l=1}^{k+1} \lim_{x_l \to 0+} (-x_l \log x_l) \\
&\quad + \left\{ \lim_{y \to 1-} (-y \log y) + \sum_{l=k+2, l\neq t}^{N} \lim_{z_l \to 0+} (-z_l \log z_l) \right\} \\
&\to 0 \quad (|\mathbb{E}_{\mathbf{U}} - 0| < \varepsilon, \textit{ which satisfy Definition 1. })
\end{aligned}
\tag{17}
$$

When $n \geq 2$, the value of $\mathbb{E}_{\mathbf{U}}$ is lower which satisfy $\max p_i \in \mathbf{B}$ than the value of $\mathbb{E}_{\mathbf{U}}$ which satisfy $\max p_i \in \mathbf{A}$.

References

1. Bao, H., Dong, L., Piao, S., Wei, F.: Beit: Bert pre-training of image transformers. In: International Conference on Learning Representations (2021)
2. Chen, X., Ding, M., Wang, X., Xin, Y., Mo, S., Wang, Y., Han, S., Luo, P., Zeng, G., Wang, J.: Context autoencoder for self-supervised representation learning. arXiv preprint arXiv:2202.03026 (2022)
3. Dong, X., Bao, J., Zhang, T., Chen, D., Zhang, W., Yuan, L., Chen, D., Wen, F., Yu, N.: Bootstrapped masked autoencoders for vision bert pretraining. In: European Conference on Computer Vision. pp. 247–264. Springer (2022)
4. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
5. Gao, P., Ma, T., Li, H., Lin, Z., Dai, J., Qiao, Y.: Mcmae: Masked convolution meets masked autoencoders. Advances in Neural Information Processing Systems **35**, 35632–35644 (2022)
6. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16000–16009 (2022)
7. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
8. Loshchilov, I., Hutter, F.: Sgdr: Stochastic gradient descent with warm restarts. arXiv preprint arXiv:1608.03983 (2016)
9. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)
10. Peng, Z., Dong, L., Bao, H., Ye, Q., Wei, F.: Beit v2: Masked image modeling with vector-quantized visual tokenizers. arXiv preprint arXiv:2208.06366 (2022)
11. Wei, C., Fan, H., Xie, S., Wu, C.Y., Yuille, A., Feichtenhofer, C.: Masked feature prediction for self-supervised visual pre-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14668–14678 (2022)
12. Wei, L., Xie, L., Zhou, W., Li, H., Tian, Q.: Mvp: Multimodality-guided visual pre-training. In: European Conference on Computer Vision. pp. 337–353. Springer (2022)
13. Xie, Z., Zhang, Z., Cao, Y., Lin, Y., Bao, J., Yao, Z., Dai, Q., Hu, H.: Simmim: A simple framework for masked image modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9653–9663 (2022)
14. Xue, H., Gao, P., Li, H., Qiao, Y., Sun, H., Li, H., Luo, J.: Stare at what you see: Masked image modeling without reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22732–22741 (2023)
15. Yi, K., Ge, Y., Li, X., Yang, S., Li, D., Wu, J., Shan, Y., Qie, X.: Masked image modeling with denoising contrast. arXiv preprint arXiv:2205.09616 (2022)
16. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 633–641 (2017)