DENOISING WITH A JOINT-EMBEDDING PREDICTIVE ARCHITECTURE

Dengsheng Chen[1] **Jie Hu**[1,2] **Xiaoming Wei**[1] **Enhua Wu**[2*]
[1]Meituan [2]Key Laboratory of System Software (Chinese Academy of Sciences) and
State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences
{chendengsheng, weixiaoming}@meituan.com, {hujie, weh}@ios.ac.cn

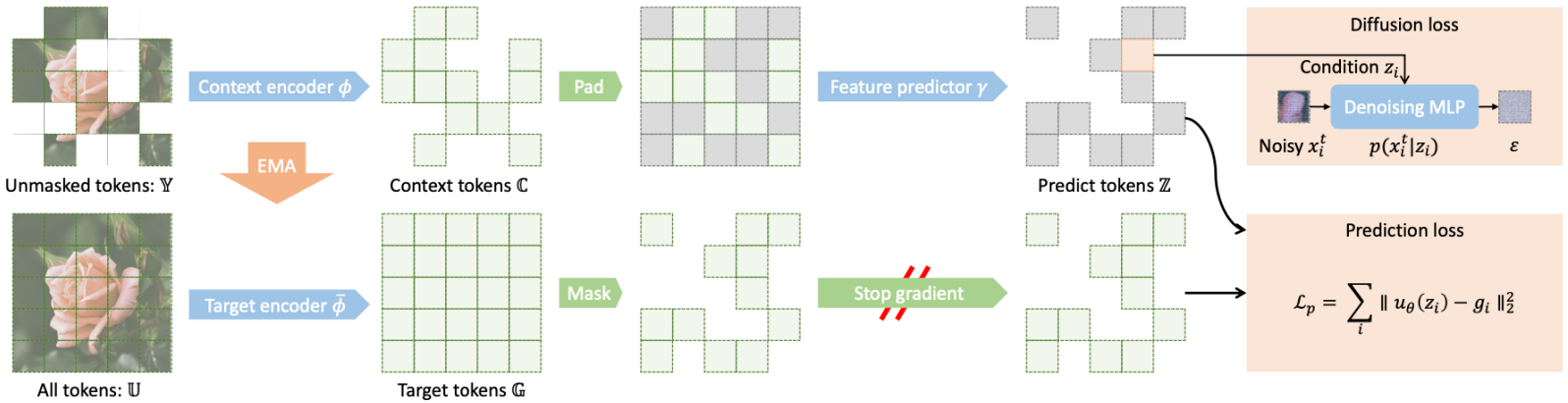

Figure 1: Data flow during D-JEPA training. Initially, the training data is divided into non-overlapping semantic tokens, which can be either in the raw space or in the latent space obtained after VAE encoding. A random subset of these input tokens is then masked. The feature predictor γ is employed to predict features for these masked tokens, utilizing the unmasked tokens as contextual information. Each masked token is concurrently subjected to a diffusion loss (or flow matching loss) to learn the distribution of each token $p(x_i|z_i)$, independently. Additionally, a prediction loss is applied, compelling each masked token to regress towards the target tokens g_i.

Algorithm 1 Generalized next token prediction with D-JEPA

Require: T: Number of auto-regressive steps, N: Total tokens to sample, τ: Temperature to control noise.

Initialize: $\mathbb{X} \leftarrow \emptyset$
for n in cosine-step-function(T, N) **do**
  $\mathbb{C} \leftarrow \phi(\mathbb{X})$ ▷ Encode the sampled tokens to obtain context features
  $\mathbb{Z} \leftarrow \gamma(\mathbb{C})$ ▷ Predict features of unsampled tokens using the feature predictor
  $\{z_0, \ldots, z_n\} \sim \mathbb{Z}$ ▷ Randomly select n tokens from $\mathbb{Z}$
  $\{x_0, \ldots, x_n\} \leftarrow \text{denoise}(\epsilon_\theta, \{z_0, \ldots, z_n\}, \tau)$ ▷ Perform denoising on the selected tokens
  $\mathbb{X} \leftarrow \mathbb{X} \cup \{x_0, \ldots, x_n\}$ ▷ Add the denoised tokens to $\mathbb{X}$
end for
Return: $\mathbb{X}$

	#Params	#Epochs	FID↓	IS↑	Pre.↑	Rec.↑
Base scale model (less than 300M)						
MAR-B (2024)	208M	800	3.48	192.4	0.78	0.58
D-JEPA-B	212M	1400	**3.40**	**197.1**	0.77	**0.61**
MaskGIT (2022)	227M	300	6.18	182.1	**0.80**	0.51
MAGE (2023)	230M	1600	6.93	195.8	-	-
Large scale model (300∼700M)						
GIVT (2023)	304M	500	5.67	-	0.75	0.59
MAGVIT-v2 (2023b)	307M	1080	3.65	200.5	-	-
MAR-L (2024)	479M	800	2.60	221.4	**0.79**	0.60
DiT-XL (2023)	675M	1400	9.62	121.5	0.67	**0.67**
SiT-XL (2024a)	675M	1400	8.60	-	-	-
MDTv2-XL (2023)	676M	400	5.06	155.6	0.72	0.66
D-JEPA-L	687M	480	**2.32**	**233.5**	**0.79**	0.62
Huge scale model (900+M)						
MAR-H (2024)	943M	800	2.35	227.8	**0.79**	**0.62**
D-JEPA-H	1.4B	320	**2.04**	**239.3**	**0.79**	**0.62**
VDM++ (2024)	2.0B	1120	2.40	225.3	-	-

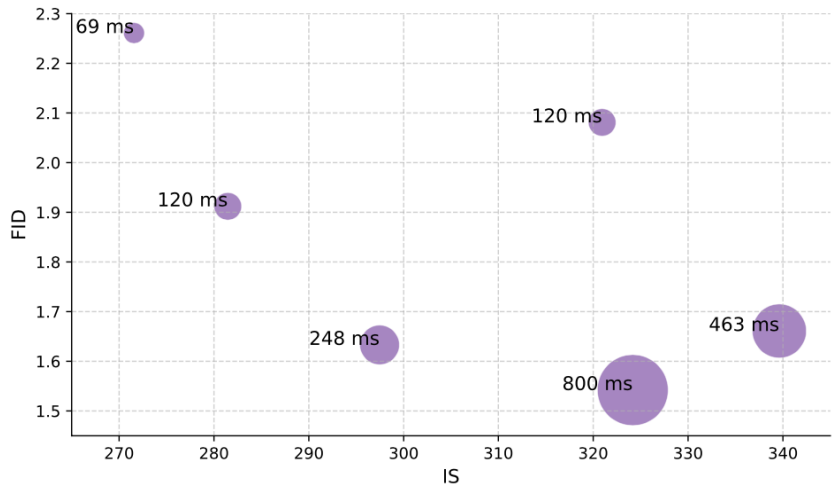

Figure 2: FID *vs.* IS under different sampling efficiencies. We can achieve FID ≈ 4.0 within 43 milliseconds (not plotted in the figure for compactness). Additionally, the figure shows that D-JEPA can achieve FID ≈ 2.0 within 120 milliseconds. Refer to Table 4 for more details. All times are the average inference time using a batch size of 256 on a single H800 GPU for generating 256×256 images.

	#params	FID↓	IS↑	Pre.↑	Rec.↑
Base scale model (less than 300M)					
MAR-B (2024)	208M	2.31	281.7	**0.82**	0.57
D-JEPA-B(cfg=3.1)	212M	**1.87**	282.5	0.80	**0.61**
D-JEPA-B(cfg=4.1)	212M	2.08	**320.9**	**0.82**	0.59
Large scale model (300∼700M)					
GIVT (2023)	304M	3.35	-	0.84	0.53
MAGVIT-v2 (2023b)	307M	1.78	319.4	-	-
VAR-d16 (2024)	310M	3.30	274.4	0.84	0.51
LDM-4 (2022)	400M	3.60	247.7	**0.87**	0.48
MAR-L (2024)	479M	1.78	296.0	0.81	0.60
U-ViT-H (2022)	501M	2.29	263.9	0.82	0.57
ADM (2021)	554M	4.59	186.7	0.82	0.52
Flag-DiT (2024b)	600M	2.40	243.4	0.81	0.58
Next-DiT (2024a)	600M	2.36	250.7	0.82	0.59
VAR-d20 (2024)	600M	2.57	302.6	0.83	0.56
DiT-XL (2023)	675M	2.27	278.2	0.83	0.57
SiT-XL (2024a)	675M	2.06	270.2	0.82	0.59
MDTv2-XL (2023)	676M	**1.58**	314.7	0.79	**0.65**
D-JEPA-L(cfg=3.0)	687M	**1.58**	303.1	0.80	0.61
D-JEPA-L(cfg=3.9)	687M	1.65	**327.2**	0.81	0.61
Huge scale model (900+M)					
MAR-H (2024)	943M	1.55	303.7	0.81	**0.62**
VAR-d24 (2024)	1.0B	2.09	312.9	0.82	0.59
D-JEPA-H(cfg=3.9)	1.4B	**1.54**	324.2	0.81	**0.62**
D-JEPA-H(cfg=4.3)	1.4B	1.68	**341.0**	**0.82**	0.61
VDM++ (2024)	2.0B	2.12	267.7	-	-

Table 2: System-level comparison on ImageNet 256 × 256 conditional generation.