JEPA
(Joint-Embedding Predictivie Architecture)

Taekhyun Park

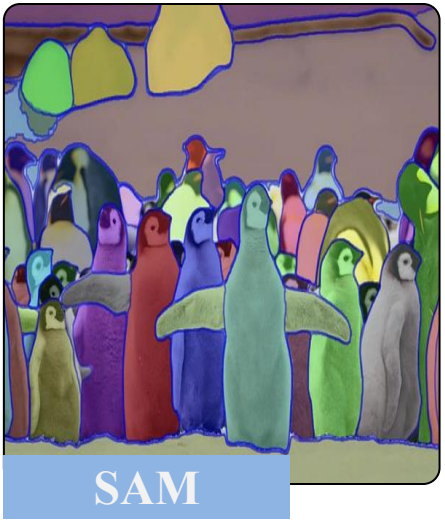
Yann Lecun

- 얀 르쿤 (Yann Lecun), 1960년생
  - AI 4대 천왕 (엔드류 응, 제프리 힌튼, 요슈아 벤자오, 얀 르쿤)
  - Citation: 1050896, h-index: 191 (2026.06.12 기준)
  - Jacob T. Schwartz Professor of Computer Science,
    Courant Institute of Mathematical Sciences,
    New York University. (2003 ~)
  - Facebook AI Research (FAIR, 2013 ~ 2026.01)
  - AMI (2026.01 ~)
  - Diplôme d'Ingénieur : ESIEE Paris (1977~1983, 응용수학) ,
    MS, Ph.D: Sorbonne Univ (1983~1987, 인공지능)
  - Post Doc: Univ of Toronto (1987~1988, 지도교수: 제프리힌튼)
  - AT&T, Bell Laboratories, Holmdel, NJ (1988~2002)

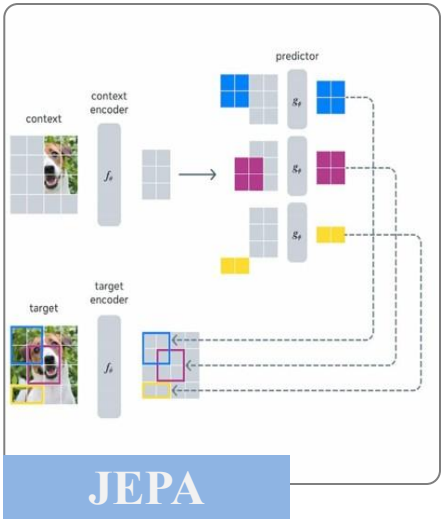
Yann Lecun in FAIR

- **FAIR 의 핵심 연구: GPT in Vision task**
  - GPT 처럼 모든 Vision task에서 Self-Supervised Learning을 통해 적용할 수 있는 Model을 구축하는 게 목표
  - 대표 Framework: **DINO** (Distillation with NO labels), **SAM** (Segment Anything Model), **JEPA** (Joint-Embedding Predictive Architecture)

DINO



SAM



JEPA

족보 of JEPA (Joint-Embedding Predictive Architecture)

Yann Lecun

Mahmoud Assran

Randall Balestriero

족보 of JEPA (Joint-Embedding Predictive Architecture)

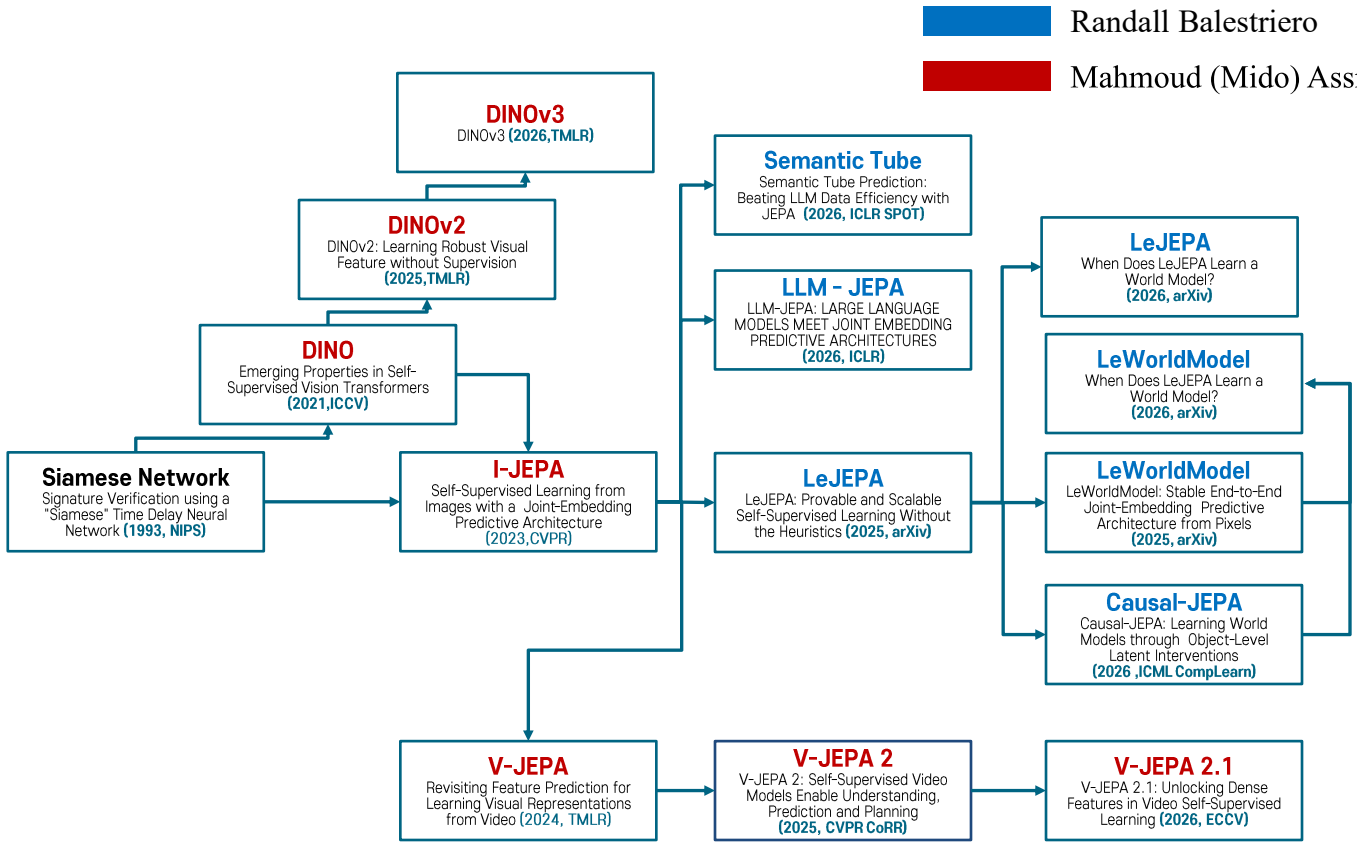
- Mahmoud (Mido) Assran
  - McGill University 학석박 (Supervisor: Micheal Rabbat) + Mila
  - Research scientist, Meta, Montreal, Canada (2017 ~)
- Papers
  - Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture
  - V-JEPA: LATENT VIDEO PREDICTION FOR VISUAL REPRESENTATION LEARNING
  - V-jepa 2: Self-supervised video models enable understanding, prediction and planning
  - V-jepa 2.1: Unlocking dense features in video self-supervised learning
- 특징
  - Engineering 적으로 매우 비직관적이면서 특이한 기법들(Stop Gradient, EMA, DINO)을 제안하며, 학술적으로 설명하기 어렵지만 성능이 좋은 테크닉들을 많이 제안함.
  - 사파 (정순하지 않은 패도적 무공을 사용)

족보 of JEPA (Joint-Embedding Predictive Architecture)

- Randall Balestriero
  - Assistant Professor of Computer Science, Brown Univ . (24.07 ~)
  - Postdoc: META (2021 ~ 2024, with Yann LeCun)
  - BS: Université du Sud Toulon Var (2014), MA: 파리 제6대학 (2016), PhD: Rice University (2021)
- Papers
  - Lejepa: Provable and scalable self-supervised learning without the heuristics
  - Semantic tube prediction: Beating LLM data efficiency with JEPA
  - Leworldmodel: Stable end-to-end joint-embedding predictive architecture from pixels
- 특징
  - JEPA 아키텍처의 학문적 당위성을 만들고, 이론적으로 동작을 잘 하는 원리들을 연구
  - 정파 (정순하고 근본 있는 무공을 추구)

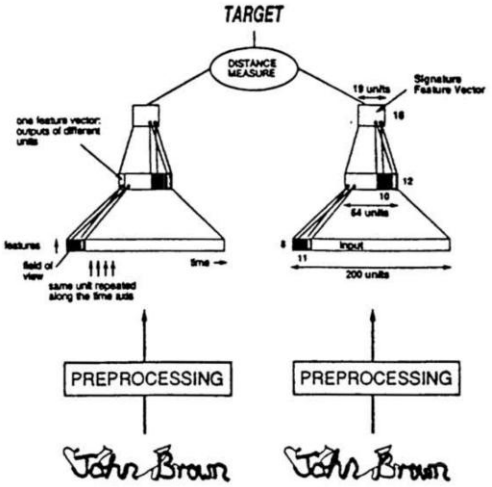
ICLR NEURAL INFORMATION PROCESSING SYSTEMS ICML International Conference On Machine Learning CVPR ECCV MSML nature COMMUNICATIONS IEEE Advancing Technology for Humanity Springer

족보 of JEPA (Joint-Embedding Predictive Architecture)



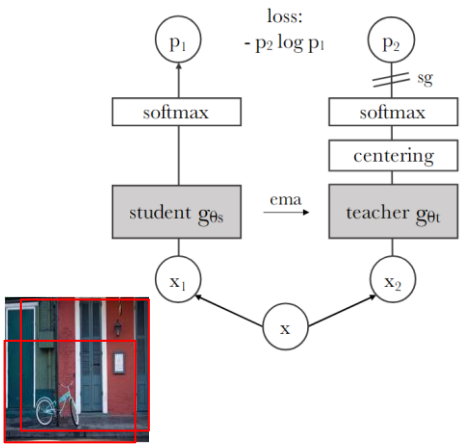
Siamese Network

- Signature Verification using a "Siamese" Time Delay Neural Network **(NIPS, 1993)**
  - 사람의 싸인 패턴을 Encoding 하여 원본 패턴과 추정 패턴 간의 거리를 계산하여 위변조를 측정하는 모델. Encoder 로는 Time Delay Neural Net (1D CNN) 를 사용.
  - "Siamese" 라는 단어에서 예측 가능하듯, 원본 Data 를 Encoding을 하는 모델과 예측 Data를 Encoding을 하는 모델을 동일하게 사용해서 특징벡터를 예측 하는 컨셉



DINO v1

- Emerging Properties in Self-Supervised Vision Transformers **(ICCV, 2021)**
  - DINO(Self- **Di**stillation with **NO** labels) 는 라벨이 없이 데이터를 스스로 학습시키는 Self-distillation 방식의 Self-supervise learning 알고리즘.
  - Teacher 모델과 Student 모델을 동시에 학습시키면서 성능을 높이는 구조. Student은 일반적인 backward 기반으로, teacher model은 ema 기반으로 업데이트를 하면서 성능을 높임.
  - 학습 목표는 하나의 이미지에 대해서 증강을 수행한 후, 각 증강된 데이터에 대해 동일한 내부표현이 되도록 Teacher와 Student를 학습



학습목표

$$P_s(x)^{(i)} = \frac{\exp(g_{\theta_s}(x)^{(i)}/\tau_s)}{\sum_{k=1}^{K}\exp(g_{\theta_s}(x)^{(k)}/\tau_s)}, \qquad (1)$$

$$\min_{\theta_s} H(P_t(x), P_s(x)), \qquad (2) \qquad H(a,b) = -a\log b.$$

$$\min_{\theta_s} \sum_{x\in\{x_1^g, x_2^g\}} \sum_{\substack{x'\in V \\ x'\neq x}} H(P_t(x), P_s(x')). \qquad (3)$$

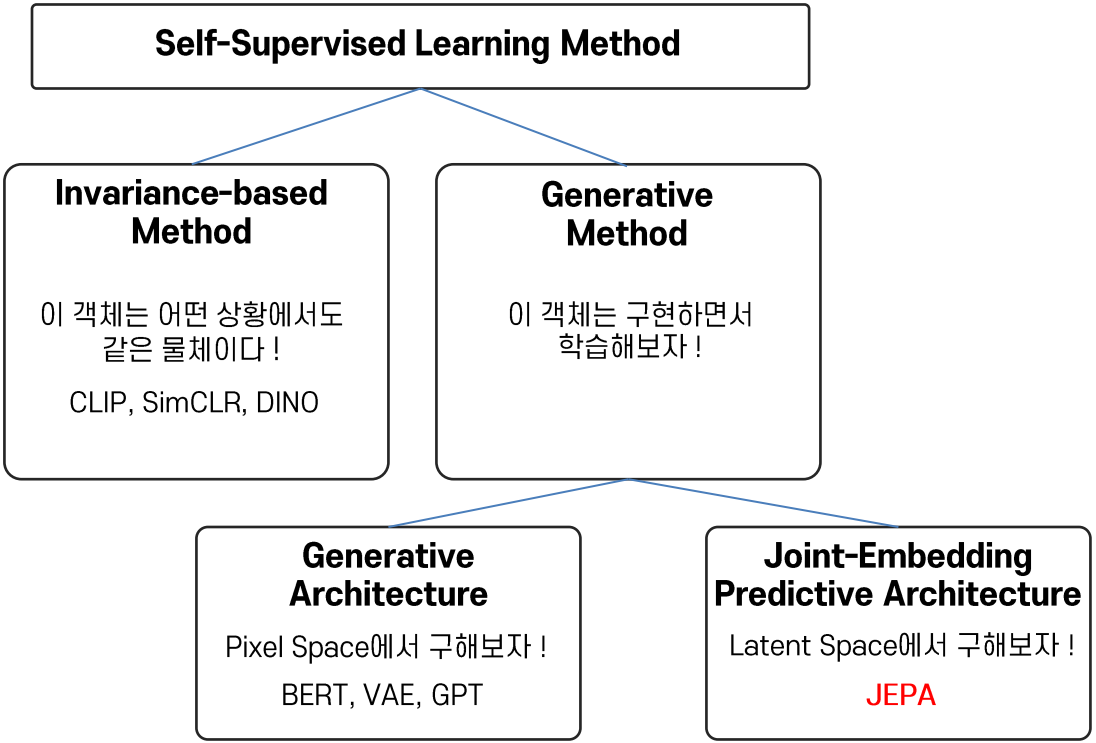
EMA

$$\theta_t \leftarrow \lambda\theta_t + (1-\lambda)\theta_s,$$

Centering

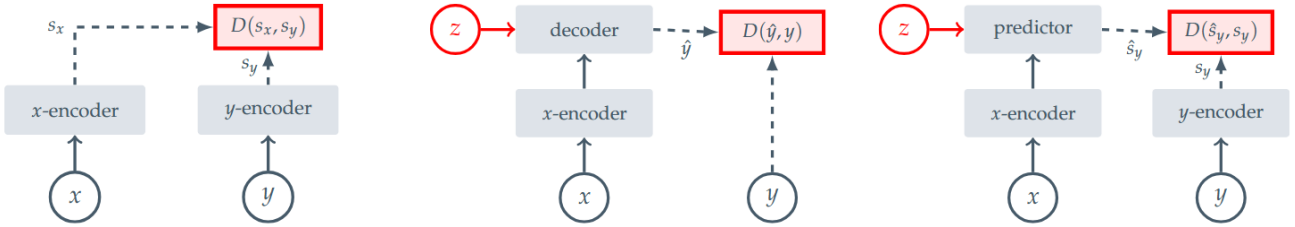
$$g_t(x) \leftarrow g_t(x) + c. \quad c \leftarrow mc + (1-m)\frac{1}{B}\sum_{i=1}^{B} g_{\theta_t}(x_i), \qquad (4)$$

What is JEPA?


Self-Supervised Learning Method
Invariance-based Method
이 객체는 어떤 상황에서도 같은 물체이다 !
CLIP, SimCLR, DINO
Generative Method
이 객체는 구현하면서 학습해보자 !
Generative Architecture
Pixel Space에서 구해보자 !
BERT, VAE, GPT
Joint-Embedding Predictive Architecture
Latent Space에서 구해보자 !
JEPA

What is JEPA?

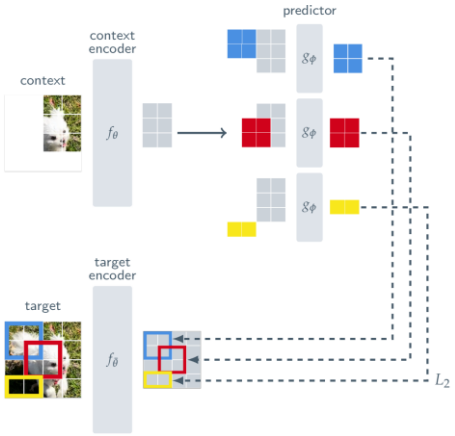
- JEPA 는 Predictor를 통해서 x-encoder의 Self-supervised Learning 시의 Representation Space를 정렬함.
- 일반적인 Biological Systems 에서 사용하는 Representation Learning을 사용함.
  (인간은 예측 기계이다.)

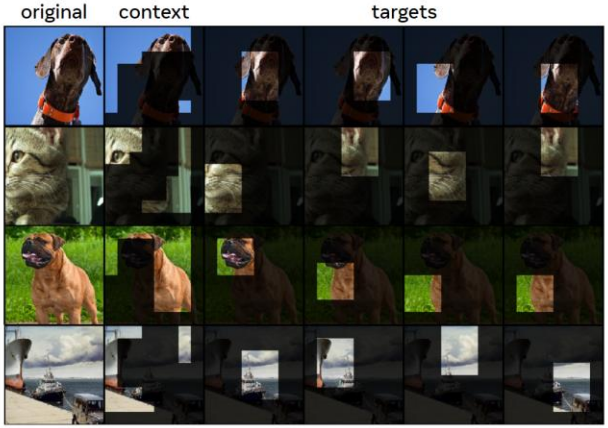


(a) **Joint-Embedding Architecture** (b) **Generative Architecture** (c) **Joint-Embedding Predictive Architecture**

I-JEPA

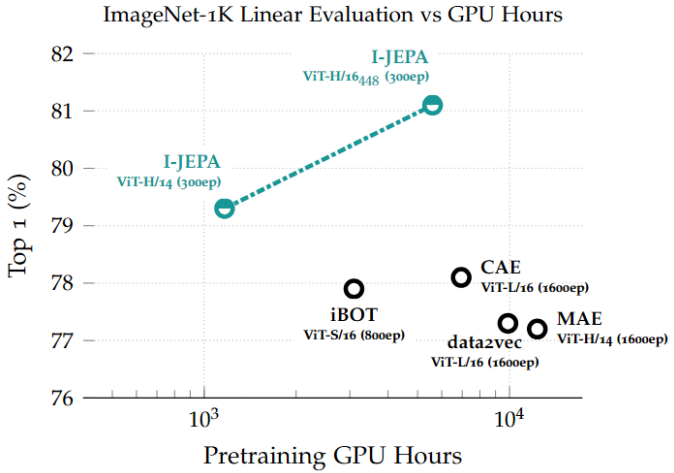
- Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture **(CVPR, 2023)**
  - 최초로 JEPA라는 단어를 사용한 논문.
  - 85%에 대한 이미지(Context)를 Encoding 하고, 마스킹 토큰 정보와 Encoding 된 정보를 predictor에 넣어, 마스킹 토큰이 위치해 있는 이미지(target)의 latent vector를 예측. Loss 는 MSE 만 사용
  - DINO 와 동일하게 stop gradient 와 EMA 기반의 update rule (target encoder) 사용. (Collapse 방지)





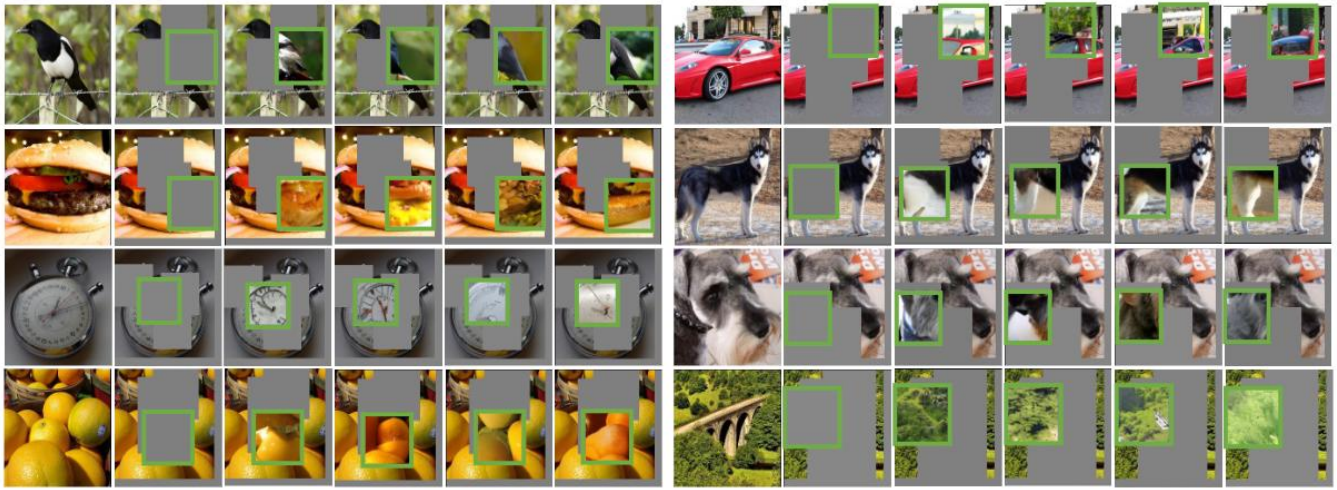
I-JEPA

- Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture **(CVPR, 2023)**
  - 실험은 학습한 Pretrained Model은 얼리고, Head만 추가해 학습 후 분류 Task를 얼마나 잘하는지 평가한 것.
  - 일반적인 다른 방법론 대비 큰 모델에서 잘 작동하는 것을 확인.
  - 빠른 수렴속도를 가지고 있는 것이 특징.
  - DINO 나 타 방법론에서 사용하는 Vision 전용 Augmentation 방법(Cropping, Rotation, Photometric Transform)을 쓰는데, 단순히 Masking 전략만을 사용해 타 데이터들에서도 일반화 해서 사용할 수 있는 장점이 존재.



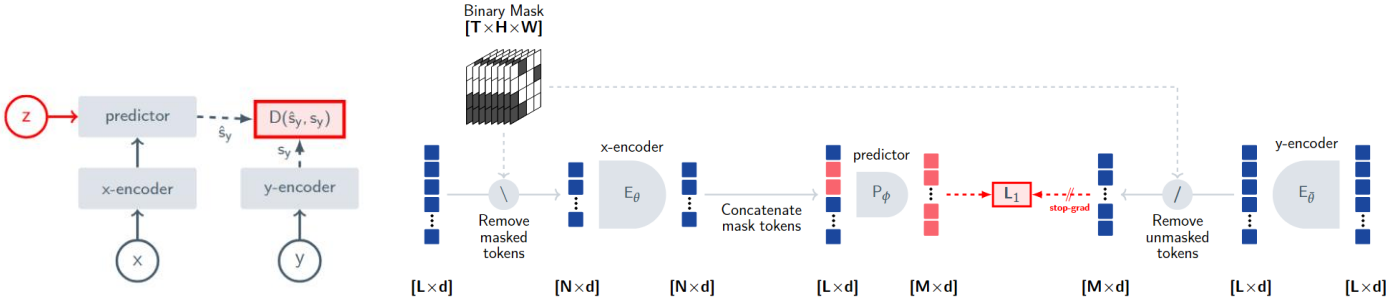
I-JEPA

- Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture **(CVPR, 2023)**
  - JEPA 학습이후, Encoder는 얼리고 Decoder(Diffusion Model)를 학습한 이후, Embedding data 기반 추론 결과.
  - 완벽하게 복원을 하지는 않지만, 의미론적으로 유사한 데이터들이 복원되는 것을 볼 수 있음.



V-JEPA

- Revisiting Feature Prediction for Learning Visual Representations from Video **(2024,TMLR)**
  - **I-JEPA** 와 동일한 아키텍처로 **16**개의 프레임의 비디오 정보를 가져와서 **Mask**로 가리고 **Context** 는 마스킹 된 비디오 정보, **Target**으로는 원본 데이터에서 마스킹 되어진 부분 인코딩.
  - 이 후, **Predictor**에 **Masking** 정보 + **Context**를 제공하여 **Target**를 예측하도록 함.
  - 추가적으로 **L1-Loss** 로 전환하면 성능이 좋아진다는 것을 확인함.
  - $\boldsymbol{minimze_{\theta,\phi}} \left\| P_{\Phi}(E_{\theta}(x), \Delta_y) - sg\left(E_{\bar{\theta}}(y)\right) \right\|_1, \bar{\theta} = \boldsymbol{EMA\ Update\ encoder}$



V-JEPA

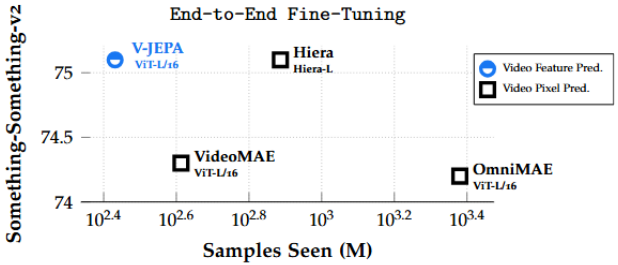
- Revisiting Feature Prediction for Learning Visual Representations from Video **(2024,TMLR)**
  - 기존 **SOTA** 모델**(DINOv2, OpenCLIP)** 대비 좋은 성능을 보이는 것을 확인. (상)
  - 또한, 기존 **Pixel Prediction** 모델 대비 좋은 성능을 보이는 것을 확인. (하)

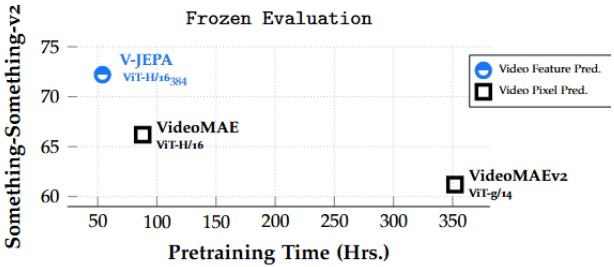
				Video Tasks			*Image Tasks*		
Method	**Arch.**	**Params.**	**Data**	**K400** (16×8×3)	**SSv2** (16×2×3)	**AVA**	**IN1K**	**Places205**	**iNat21**
Methods pretrained on Images									
I-JEPA	ViT-H/16_{512}	630M	IN22K	79.7	50.0	19.8	84.4	66.5	85.7
OpenCLIP	ViT-G/14	1800M	LAION	81.8	34.8	23.2	85.3	**70.2**	83.6
DINOv2	ViT-g/14	1100M	LVD-142M	**83.4**	50.6	24.3	**86.2**	68.4	**88.8**
Methods pretrained on Videos									
MVD	ViT-L/16	200M	IN1K+K400	79.4	66.5	19.7	73.3	59.4	65.7
OmniMAE	ViT-H/16	630M	IN1K+SSv2	71.4	65.4	16.0	76.3	60.6	72.4
VideoMAE	ViT-H/16	630M	K400	79.8	66.2	20.7	72.3	59.1	65.5
VideoMAEv2	ViT-g/14	1100M	Un.Hybrid	71.2	61.2	12.9	71.4	60.6	68.3
Hiera	Hiera-H	670M	K400	77.0	64.7	17.5	71.4	59.5	61.7
V-JEPA	ViT-L/16	200M	VideoMix2M	80.8	69.5	25.6	74.8	60.3	67.8
	ViT-H/16	630M		**82.0**	71.4	**25.8**	75.9	61.7	67.9
	ViT-H/16_{384}	630M		81.9	**72.2**	25.0	**77.4**	**62.8**	**72.6**

				Frozen Evaluation w/ Att. Pooling						*Fine-Tuning*	
Method	**Arch.**	**#Samples Seen**	**Iter.**	**K400** (16×8×3)	**SSv2** (16×2×3)	**AVA**	**IN1K**	**Places205**	**iNat21**	**K400-ft** (16×5×3)	**SSv2-ft** (16×2×3)
Methods pretrained using pixel prediction											
OmniMAE	ViT-L/16	2400M	1170K	65.6	60.6	14.4	**75.1**	59.8	66.1	84.0	74.2
VideoMAE	ViT-L/16	410M	400K	77.8	65.5	21.6	71.1	59.3	64.6	85.4	74.3
Hiera	Hiera-L	770M	1500K	75.5	64.2	15.8	68.9	58.5	56.9	**87.3**	**75.1**
V-JEPA	ViT-L/16	270M	90K	**80.8**	**69.5**	**25.6**	74.8	**60.3**	**67.8**	85.6	**75.1**

V-JEPA

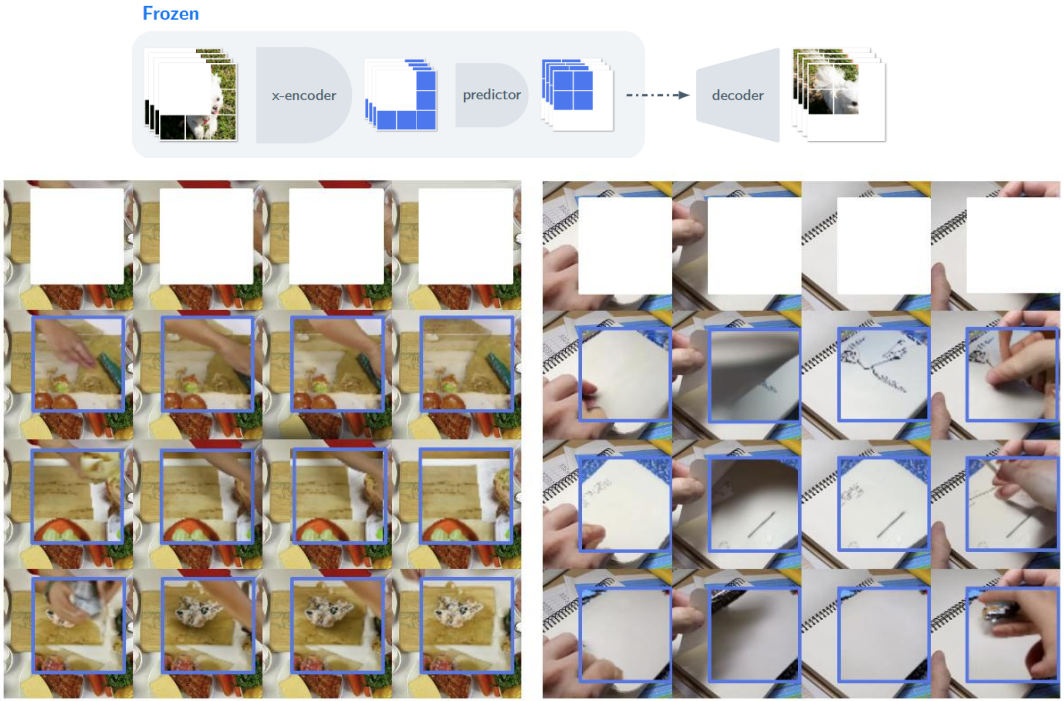
- Revisiting Feature Prediction for Learning Visual Representations from Video **(2024,TMLR)**
  - **Fine-Tuning** 시 적은 **Data**만 보여줘도, **V-JEPA** 는 좋은 성능을 보이는 것을 확인(좌)
  - 또한 **Pretraining Time** 또한 기존의 **Pixel Pred** 모델 대비 매우 적은 것을 알 수 있음. (우)





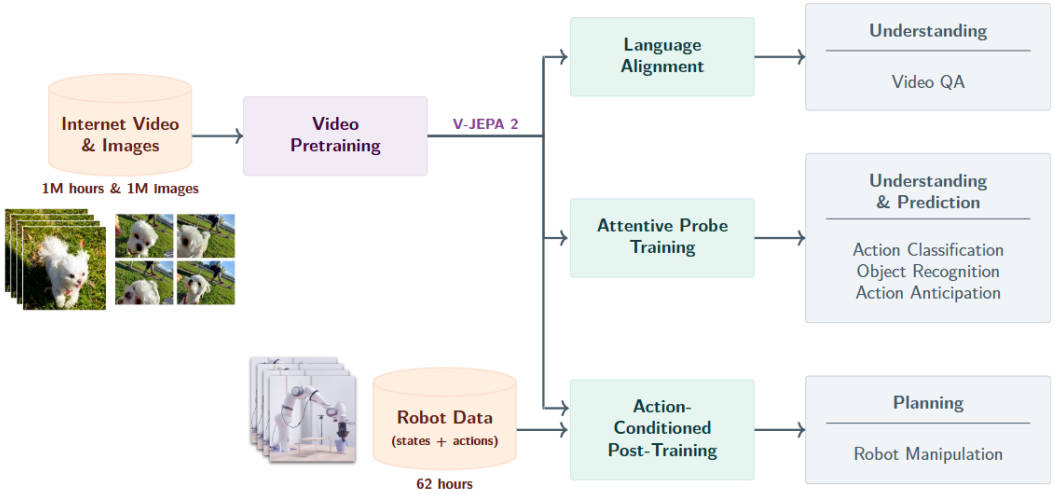
V-JEPA

- Revisiting Feature Prediction for Learning Visual Representations from Video **(2024,TMLR)**
  - **I-JEPA** 때와 동일하게, **encoder**를 얼리고 **decode**만 학습이 이 후, **Latent Vector**를 시각화 해봤을 때, 시간적 맥락과 정보를 잘 **Encoding**을 하고 있음을 확인함.



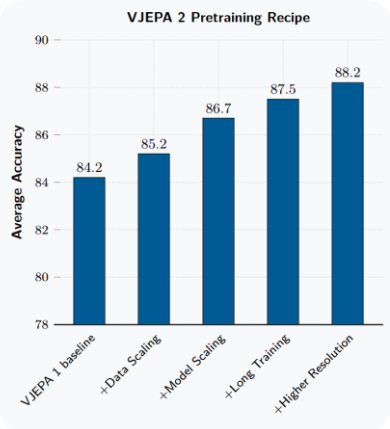
V-JEPA 2.0

- V-JEPA 2: Self-Supervised Video Models Enable Understanding, Prediction and Planning **(2025, CVPR CoRR)**
  - **V-JEPA 를 확장(VIT-H(630m) -> VIT-G(1.1B))하여 모델 규모를 확장. (+ 3D RoPE 추가)**
  - **단순 Representation Learning 에서 WorldModel로 확장. + LLM 과 Alignment 해서 Video QA도 가능하게 만듦.**



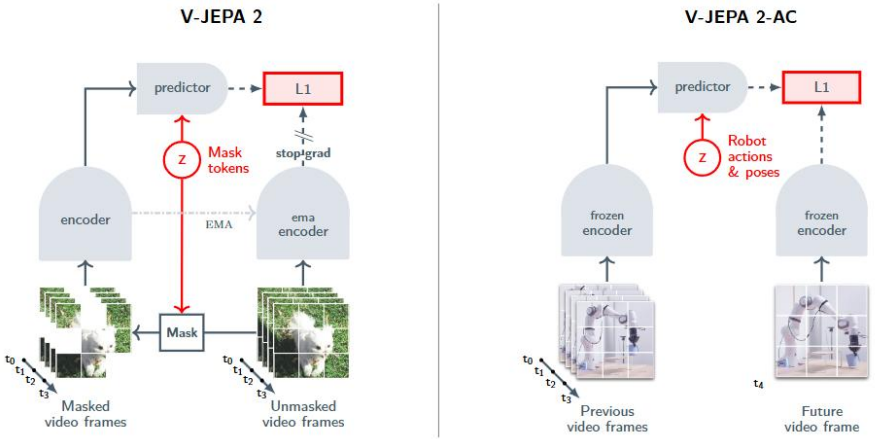
V-JEPA 2.0

- V-JEPA 2: Self-Supervised Video Models Enable Understanding, Prediction and Planning **(2025, CVPR CoRR)**
  - **Pretraining 단계에서 V-JEPA 2에 추가된 것들**
  - **(Data Scaling)** 기존 학습데이터를 200만개에서 2200만개로 늘리고, 클러스터링 기반의 큐레이션 기법으로 정보량 적은 영상 제거
  - **(Model Scaling)** 모델 사이즈를 **630M** 에서 **1.1B** 로 키워서 성능을 향상 **(Scaling Law** 적용 확인)
  - **(Long Training) Iteration** 은 **9**만번에서 **25.2**만번으로 확장
  - **(Higher Resolution) 16**프레임 데이터에서 **64** 프레임+고해상도 데이터로 학습을 해서 성능향상



V-JEPA 2.0

- V-JEPA 2: Self-Supervised Video Models Enable Understanding, Prediction and Planning **(2025, CVPR CoRR)**
  - **Pretrained Model**을 학습하는 방법론 자체는 **V-JEPA, I-JEPA** 와 동일하게 **Mask**로 가리고, **Mask tokens + Context** 로 **Target**을 구하는 형식 **(L1 Loss)**
  - 이후 **V-JEPA** 에서 **Robot**의 **Action** 을 예측하는 **V-JEPA 2-AC**를 **Fine-Tuning** 시에는 **Mask-Token** 대신 **Robot actions + poses**를 해서 **Robot**의 다음 움직임을 예측



V-JEPA 2.0

- V-JEPA 2: Self-Supervised Video Models Enable Understanding, Prediction and Planning **(2025, CVPR CoRR)**
  - **V-JEPA 2-AC** 는 현재 상태(s_t), 비디오 프레임(x_t), 행동 데이터(a_t) 를 기반으로 다음 **Latent Space**를 예측 하는 모델.
  - **Loss**는 다음 z_{t+1}를 예측하는것에 대한 Loss 인 $\mathcal{L}_{teacher-forcing}$ 과 다음 z_t 에서 다음 행동 정보 $a_{t:t+k}$ 를 모두 사용 했을 때 나오는 결과에 대한 Loss인 $\mathcal{L}_{rollout}$ 을 결합해서 사용

$$\mathcal{L}_{\text{teacher-forcing}}(\phi) \coloneqq \frac{1}{T}\sum_{k=1}^{T}\|\hat{z}_{k+1} - z_{k+1}\|_1 = \frac{1}{T}\sum_{k=1}^{T}\left\|P_\phi\left((a_t, s_t, E(x_t))_{t\le k}\right) - E(x_{k+1})\right\|_1, \qquad (2)$$

$$\mathcal{L}_{\text{rollout}}(\phi) \coloneqq \|P_\phi(a_{1:T}, s_1, z_1) - z_{T+1}\|_1.$$

$$L(\phi) \coloneqq \mathcal{L}_{\text{teacher-forcing}}(\phi) + \mathcal{L}_{\text{rollout}}(\phi),$$

V-JEPA 2.0

- V-JEPA 2: Self-Supervised Video Models Enable Understanding, Prediction and Planning **(2025, CVPR CoRR)**
  - 실험 결과 **Zero-Shot** 에서 타 모델 대비 매우 좋은 성능을 보이는 것을 확인했음.
  - **Limitation** 으로 카메라 위치에 예민하다는 점, 그리고 지역 최적점에 갇힐 수 있음.

Table 2 Zero-Shot Robot Manipulation. All models are deployed zero-shot on two Franka arms with RobotiQ grippers located in different labs. Given image-goals for each considered task, all models run closed loop to infer a sequence of actions to achieve the goal. Success rates are reported out of 10 trials with various permutations to the task across trials (e.g., object location, starting pose, etc.).

			Grasp		Reach w/ Obj.		Pick-&-Place	
Method		**Reach**	Cup	Box	Cup	Box	Cup	Box
Octo (Octo Model Team et al., 2024)	Lab 1	100%	20%	0%	20%	70%	20%	10%
	Lab 2	100%	10%	0%	10%	70%	10%	10%
	Avg	*100%*	*15%*	*0%*	*15%*	*70%*	*15%*	*10%*
V-JEPA 2-AC (ours)	Lab 1	100%	70%	30%	90%	80%	80%	80%
	Lab 2	100%	60%	20%	60%	70%	80%	50%
	Avg	*100%*	*65%*	*25%*	*75%*	*75%*	*80%*	*65%*

V-JEPA 2.0

- V-JEPA 2: Self-Supervised Video Models Enable Understanding, Prediction and Planning **(2025, CVPR CoRR)**
  - 기본적인 **Video Understanding Task**에서 타 **SOTA** 모델 대비 우수한 성능이 나오는 것을 확인함.
  - 또한 **Size**가 커질수록 성능이 좋은 **Scaling LAW** 또한 확인

Method	Param.	Avg.	*Motion Understanding* SSv2	Diving-48	Jester	*Appearance Understanding* K400	COIN	IN1K
Results Reported in the Literature								
VideoMAEv2 (Wang et al., 2023)	1B	–	56.1	–	–	82.8	–	71.4
InternVideo2-1B (Wang et al., 2024b)	1B	–	67.3	–	–	87.9	–	–
InternVideo2-6B (Wang et al., 2024b)	6B	–	67.7	–	–	88.8	–	–
VideoPrism (Zhao et al., 2024)	1B	–	68.5	71.3	–	87.6	–	–
Image Encoders Evaluated Using the Same Protocol								
DINOv2 (Darcet et al., 2024)	1.1B	81.1	50.7	82.5	93.4	83.6	90.7	86.1
PE$_{\text{core}}$G (Bolya et al., 2025)	1.9B	82.3	55.4	76.9	90.0	88.5	**95.3**	87.6*
SigLIP2 (Tschannen et al., 2025)	1.2B	81.1	49.9	75.3	91.0	87.3	95.1	**88.0**
Video Encoders Evaluated Using the Same Protocol								
V-JEPA ViT-H (Bardes et al., 2024)	600M	85.2	74.3	87.9	97.7	84.5	87.1	80.0
InternVideo2$_{s2}$-1B (Wang et al., 2024b)	1B	87.0	69.7	86.4	97.0	**89.4**	93.8	85.8
V-JEPA 2 ViT-L	300M	86.0	73.7	89.0	97.6	85.1	86.8	83.5
V-JEPA 2 ViT-H	600M	86.4	74.0	89.8	97.7	85.3	87.9	83.8
V-JEPA 2 ViT-g	1B	87.5	75.3	90.1	97.7	86.6	90.7	84.6
V-JEPA 2 ViT-g$_{384}$	1B	**88.2**	**77.3**	**90.2**	**97.8**	87.3	91.1	85.1

V-JEPA 2.0

- V-JEPA 2: Self-Supervised Video Models Enable Understanding, Prediction and Planning **(2025, CVPR CoRR)**
  - Video Question Answering Task에 적용하기 위해 LLAVA Method 기반으로 Encoder형식으로 붙인 다음 Alignment 수행
  - 타 Video Encoder 들 대비 제일 좋은 성능이 나오는 것을 확인 (좌상)
  - Encoder 로서 Scaling Law를 만족하는것을 확인 (우상)
  - LLAMA 3.1 8B와 결합 후 Align 했을 때, 타 모델 방법론 대비해서 SOTA를 달성 (하단)

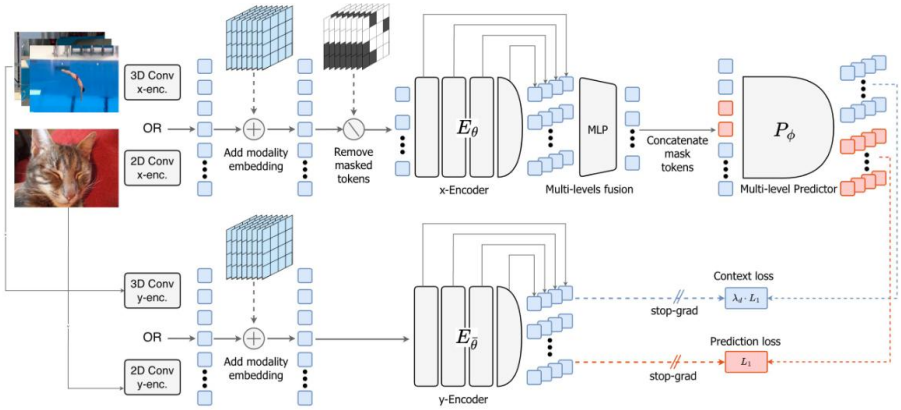
Method	Params Enc / LLM	Avg.	PerceptionTest SFT / Acc	MVP Paired-Acc	TempCompass multi-choice	TemporalBench (MBA-short QA)	TVBench Acc	TOMATO Acc	MVBench Acc
Off-the-shelf image encoders									
DINOv2 ViT-g$_{518}$	1.1B/7B	45.7	67.1	22.4	62.3	26.8	47.6	32.0	61.8
SigLIP2 ViT-g$_{384}$	1.1B/7B	48.1	**72.4**	26.2	66.8	25.7	48.7	33.2	64.0
PE ViT-G/14$_{448}$	1.9B/7B	49.1	72.3	26.7	67.0	27.5	51.6	34.0	64.7
V-JEPA 2 ViT-g$_{512}$	1B/7B	**52.3**	72.0	**31.1**	**69.2**	**33.3**	**55.9**	**37.0**	**67.7**

Method	Params Enc / LLM	Avg.	PerceptionTest SFT / Acc	MVP Paired-Acc	TempCompass multi-choice	TemporalBench (MBA-short QA)	TVBench Acc	TOMATO Acc	MVBench Acc
End-to-end Evaluation									
V-JEPA 2 ViT-L$_{256}$	300M/7B	51.7	74.6	32.3	70.1	30.2	50.9	36.5	67.1
V-JEPA 2 ViT-H$_{256}$	600M/7B	52.0	74.7	30.6	70.9	29.8	54.6	35.1	68.0
V-JEPA 2 ViT-g$_{256}$	1B/7B	52.3	75.5	31.9	70.7	28.3	54.2	37.3	68.3
V-JEPA 2 ViT-g$_{384}$	1B/7B	54.0	76.5	33.0	**71.7**	**33.1**	56.5	**39.0**	68.5
V-JEPA 2 ViT-g$_{512}$	1B/7B	**54.4**	**77.7**	**33.7**	71.6	32.3	**57.5**	38.5	**69.5**

Method	Params Enc / LLM	Avg.	PerceptionTest Test Acc	MVP Paired-Acc	TempCompass multi-choice	TemporalBench (MBA-short QA)	TOMATO Acc	TVBench Acc	MVBench Acc
$\leq 8B$ Video Language Models Results Reported in the Literature									
InternVL-2.5 (Chen et al., 2024)	300M/7B	52.1	68.9	39.9	68.3	24.3	29.4	61.6	72.6
Qwen2VL (Wang et al., 2024a)	675M/7B	47.0	66.9	29.2	67.9	20.4	31.5	46.0	67.0
Qwen2.5VL (Qwen Team et al., 2025)	1B/7B	49.7	70.5	36.7	71.7	24.5	24.6	50.5	69.6
PLM 8B (Cho et al., 2025)	1B/8B	56.7	82.7	39.7	72.7	28.3	33.2	**63.5**	**77.1**
V-JEPA 2 ViT-g$_{384}$ **LLama 3.1 8B**	1B/8B	**59.5**	**84.0**	**44.5**	**76.9**	**36.7**	**40.3**	60.6	73.5

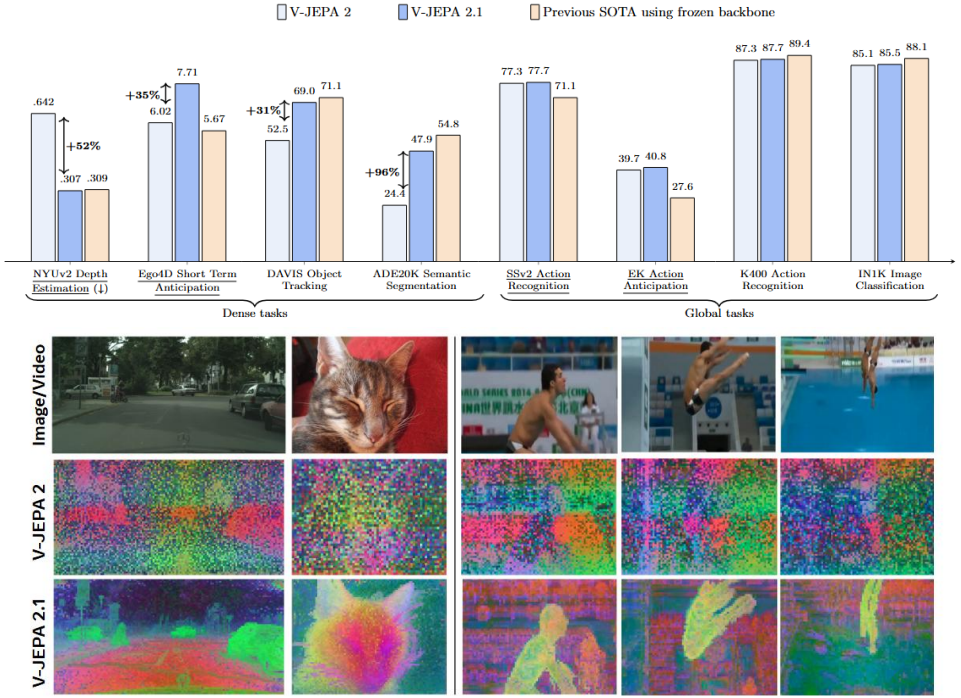
V-JEPA 2.1

- **V-JEPA 2.1: Unlocking Dense Features in Video Self-Supervised Learning (ECCV, 2026)**
  - V-JEPA 2.0 까지 동작 인식 맥락 이해는 좋았지만 정밀한(Dense) 정보들(깊이, 경계, 세부 정보)들을 추출하는데 한계가 존재했음. 이를 해결하기 위해
  - (**Dense Predictive Loss**) 기존 V-JEPA에서는 마스킹된 부분만 예측하게 했는데, 모든 보이는 부분을 예측하게 손실함수를 바꿈
  - (**Deep Self Supervision**) Middle Layer 의 정보를 Fusion 하여 표현력을 높임. (YOLO)
  - (**Multi-Modal Tokenizer**) V-JEPA 2.0에서는 이미지와 비디오 토크나이저를 동일하게 썼는데 개별적으로 하고 Architecture는 그대로 해서 동시학습 하도록 수정함.



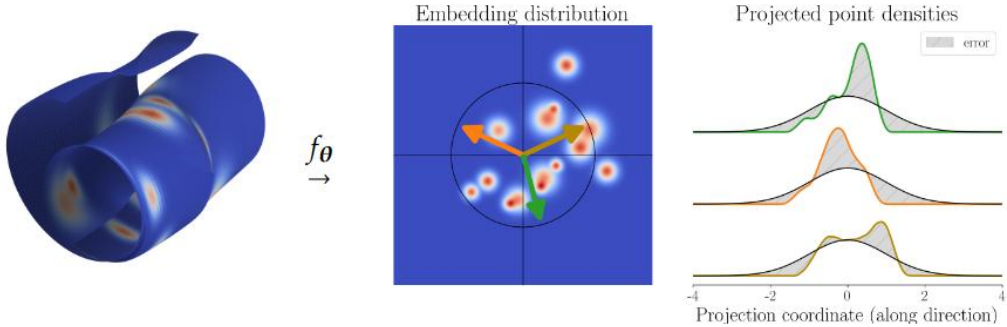
V-JEPA 2.1

- **V-JEPA 2.1: Unlocking Dense Features in Video Self-Supervised Learning (ECCV, 2026)**
  - 전반적으로 Dense Task에 대해 큰 성능향상이 있었음(상단)
  - 또한 PCA로 정보를 확인했을 때, 사물의 구조를 명확하게 이해하고 있는것을 볼 수 있음



LeJEPA

- **LeJEPA: Provable and Scalable Self-Supervised Learning without the Heuristics (2025, arXiv)**
  - 기존 논문은 **Representation Collapse** (Latent Space 의 값들이 모두 동일해지는 현상) 을 막기 위해 휴리스틱 방법론을 많이 사용했었음. (Stop-gradient, EMA, Asymmetric architectures(학습 때는 Teacher, Student 구분)). 그리고 이러한 휴리스틱 방법론은 또한 Transformer Architecture 에만 동작하는 것을 알 수 있음.
  - 이러한 Collapse 를 해결하기 위해서 Embedding 분포가 Isotropic Gaussian 여야 한다는 것을 보였고, SigReg 라는 Loss를 제안했고, 이 Loss 하나만으로 기존의 휴리스틱 방법론들 없이 학습된다는 것을 보였음.



LeJEPA

- **LeJEPA: Provable and Scalable Self-Supervised Learning without the Heuristics (2025, arXiv)**
  - SIGReg 는 JEPA에서 Collapse를 막기 위해 Latent Vector의 분포를 Isotropic Gaussian 으로 강제를 하는 Loss 임.
  - 이를 구현하기 위해서 Cramer-Wold 정리를 기반으로 고차원 데이터를 무작위 방향으로 Projection 을 한 이후 1차원 값들로 만든 뒤 이를 Epps-Pulley Test (데이터 내의 Characteristic Function를 계산한 뒤, 이를 표준 Gaussian Function 과 거리를 비교) 를 통해 Isotropic Gaussian 을 생성함.

Definition 2: SIGReg (PyTorch code in algorithm 1)

SIGReg sketches a statistical test T towards isotropic Gaussian

$$\text{SIGReg}_T(\mathbb{A}, \{f_\theta(x_n)\}_{n=1}^N) \triangleq \frac{1}{|\mathbb{A}|}\sum_{a\in\mathbb{A}} T(\{a^\top f_\theta(x_n)\}_{n=1}^N), \quad \text{(SIGReg)}$$

where we recommend the Epps-Pulley test (Section 4.2.3) for T.

Theorem 4: Stability of Epps-Pulley Test

(Epps–Pulley) satisfies for samples $z_1, \ldots, z_N$

$$\left|\frac{\partial EP(\mathbf{a})}{\partial z_i}\right| \le \frac{4\sigma^2}{N}, \quad \left|\frac{\partial^2 EP(\mathbf{a})}{\partial z_i^2}\right| \le \frac{C\sqrt{\pi}\sigma^3}{2N},$$

with constant C, and bandwidth σ. (Proof in Section B.12.)

Lemma 3: Hyperspherical Cramér-Wold

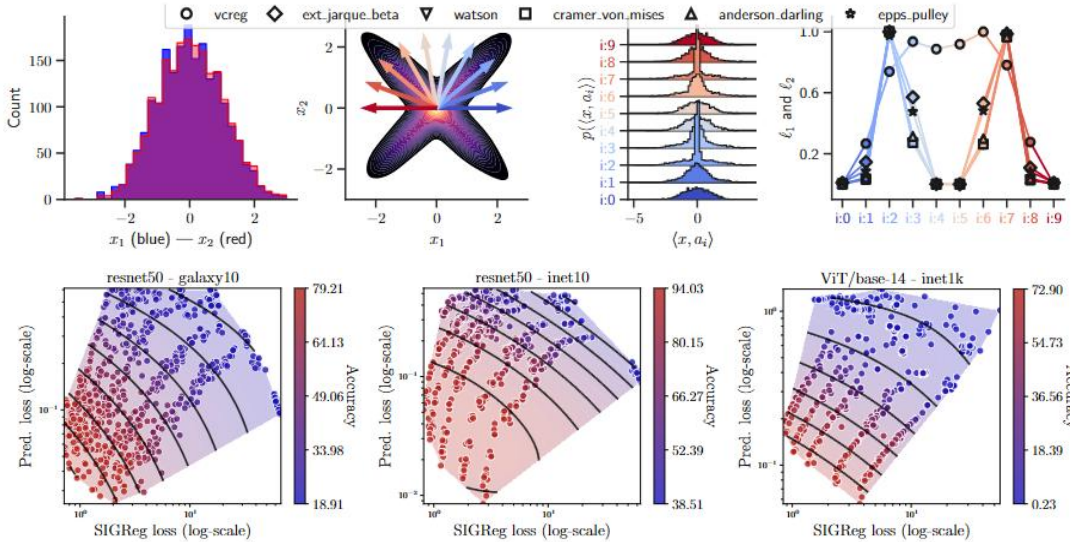
Let X, Y be $\mathbb{R}^d$-valued random vectors, then

$$\langle \boldsymbol{u}, X\rangle \overset{d}{=} \langle \boldsymbol{u}, Y\rangle, \forall \boldsymbol{u} \in \mathbb{S}^{d-1} \iff X \overset{d}{=} Y.$$

Convergence in distribution also holds. (Proof in Section B.8.)

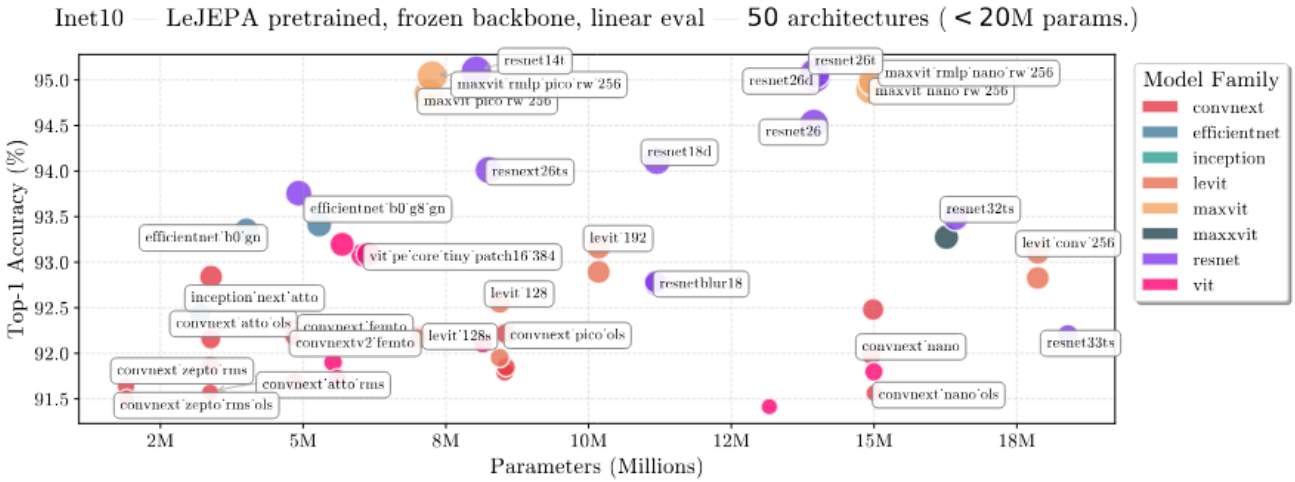
LeJEPA

- **LeJEPA: Provable and Scalable Self-Supervised Learning without the Heuristics (2025, arXiv)**
  - X 자 형태의 분포에 해서도 SigReg로 학습을 하게 되면 Isotropic Gaussian이 아닌것을 확인할 수 있고(상), SIGReg Loss 가 낮아짐에 따라 Pred. loss 가 감소하는 것을 확인할 수 있음.(하)



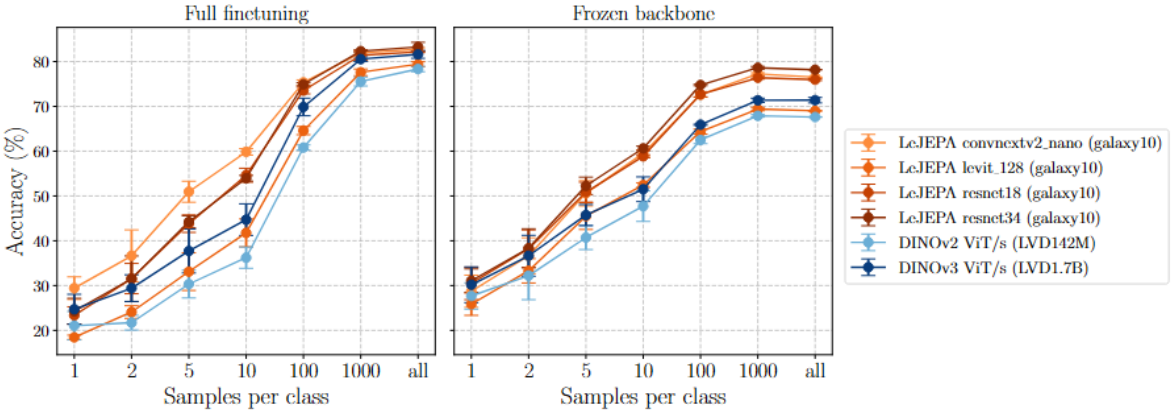
LeJEPA

- **LeJEPA: Provable and Scalable Self-Supervised Learning without the Heuristics (2025, arXiv)**
  - ViT 를 제외한 다른 Backbone 들에서도 매우 좋은 성능을 보이는 것을 확인할 수 있음.



LeJEPA

- **LeJEPA: Provable and Scalable Self-Supervised Learning without the Heuristics (2025, arXiv)**
  - DINO 와 같은 대형 Backbone 모델들을 Transfer Learning 하는 것 보다도 LeJEPA 로 도메인에 맞춰서 학습하는 것이 SOTA 성능이 나온다는 것을 확인할 수 있음. (도메인 특화)



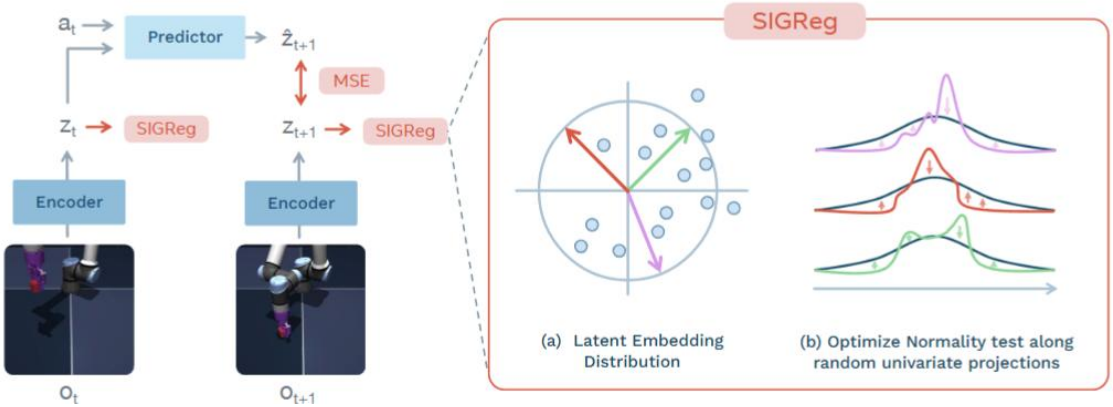
LeWorldModel

- **LeWorldModel: Stable End-to-End Joint-Embedding Predictive Architecture from Pixels (arXiv, 2026)**
  - **LeJEPA** 아키텍처를 기반으로 동일 **Encoder**를 사용을 하고, $o_{0:t}$ 까지의 데이터를 **Encoder**에 입력하고 , a_t 입력으로 넣고 **Predictor**를 사용하여 Z_{t+1} 을 예측하는 구조

$$\mathcal{L}_{\text{pred}} \triangleq \|\hat{z}_{t+1} - z_{t+1}\|_2^2, \qquad \hat{z}_{t+1} = \text{pred}_\phi(z_t, a_t). \tag{1}$$

$$\text{SIGReg}(Z) \triangleq \frac{1}{M}\sum_{m=1}^{M} T(h^{(m)}). \tag{2}$$

$$\mathcal{L}_{\text{LeWM}} \triangleq \mathcal{L}_{\text{pred}} + \lambda\,\text{SIGReg}(Z). \tag{3}$$

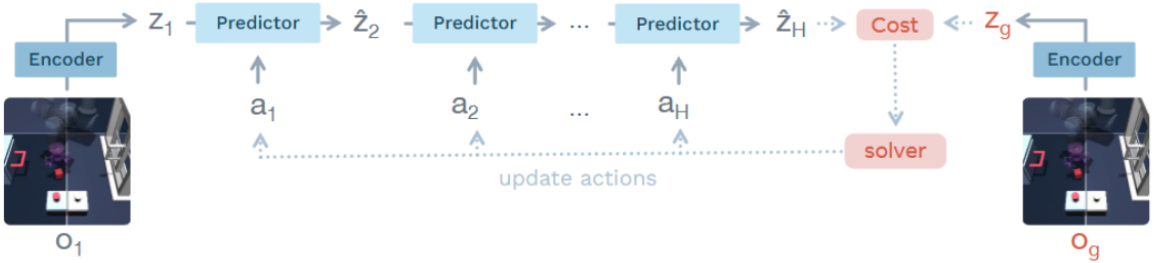


LeWorldModel

- **LeWorldModel: Stable End-to-End Joint-Embedding Predictive Architecture from Pixels (arXiv, 2026)**
  - **LeJEPA V-JEPA 2.0** 과 동일하게 목표 z_g와 가장 가까운 거리의 값을 찾고, 이를 기반으로 행동 분포를 업데이트를 하는 형태로 수행
  - 그리고 **Model Predictive Control** 전략을 사용해서, **K**개의 행동만 실행하고 이후에 다시 계속 관측값을 받아서 계획을 새우는 형태로 만듬
  - 이를 통해 **DINO-WM** 대비 **48**배 빠른 계획 속도를 보여줌.

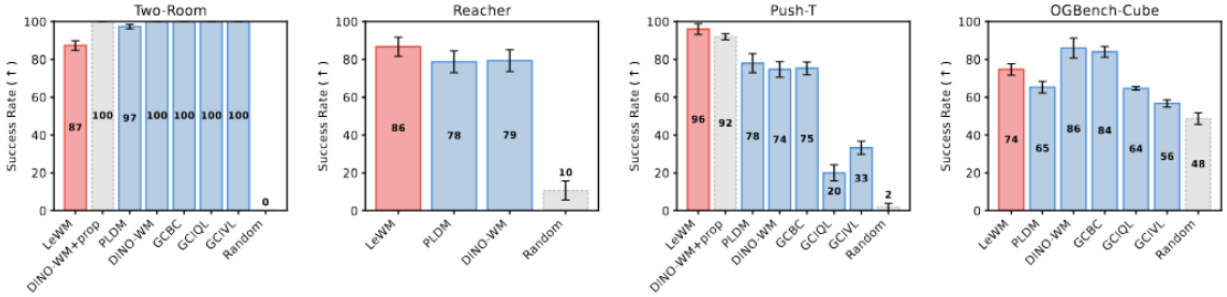
$$\mathcal{C}(\hat{z}_H) = \|\hat{z}_H - z_g\|_2^2, \quad z_g = \mathrm{enc}_\theta(o_g), \tag{4}$$

$$a^*_{1:H} = \arg\min_{a_{1:H}} \mathcal{C}(\hat{z}_H), \tag{5}$$



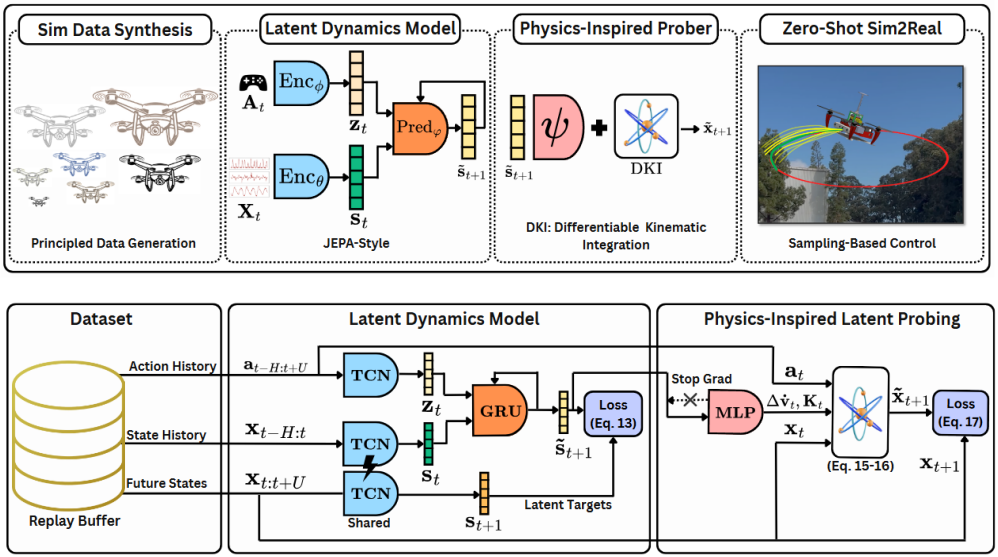
LeWorldModel

- **LeWorldModel: Stable End-to-End Joint-Embedding Predictive Architecture from Pixels (arXiv, 2026)**
  - **Planning task**에서 대부분의 모델 대비 좋은 성능을 보임.
  - **LeJEPA** 은 짧은 미래를 예측하는데 최적화 되어 있으나, 장기 계획에는 약한 지점이 있음.



Application of JEPA

- SkyJEPA: Learning Long-Horizon World Models for Zero-Shot Sim-to-Real Control of Quadrotors (arXiv, 2026.06.23, UC Berkeley ,**Randall Balestriero**)
  - 100Hz 환경에서 명령을 내려야 하는 쿼드로터의 비행제어를 LeWM 기반으로 수행한 모델.
  - 고속환경과 실제 환경에 맞게 Encoder를 TCN, Predictor를 GRU를 사용했고, Physics 를 기반으로 동작하게 하는 시스템(DKI (병진 운동과 회전 운동))을 구축



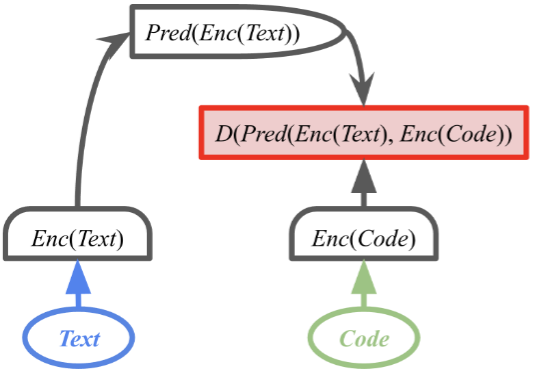
Application of JEPA

- SkyJEPA: Learning Long-Horizon World Models for Zero-Shot Sim-to-Real Control of Quadrotors (arXiv, 2026.06.23, UC Berkeley ,**Randall Balestriero**)
  - https://pratyaksh10.github.io/skyjepa-project-page/

Application of JEPA

- LLM-JEPA: LARGE LANGUAGE MODELS MEET JOINT EMBEDDING PREDICTIVE ARCHITECTURES **(ICLR 2026, Hai Huang, Randall Balestriero)**
  - I-JEPA base로 Text 와 Code 간의 Hidden State 에서 JEPA 를 사용해 성능을 높인 논문.

$$\mathcal{L}_{\text{LLM-JEPA}} = \underbrace{\sum_{\ell=2}^{L} \mathcal{L}_{\text{LLM}}(\text{Text}_{1:\ell-1}, \text{Text}_{\ell})}_{\text{generative capabilities (LLM)}} + \lambda \times \underbrace{d(\text{Pred}(\text{Enc}(\text{Text})), \text{Enc}(\text{Code}))}_{\text{abstraction capabilities (JEPA)}}, \quad (2)$$



Natural Language to Regular Expression

```
source: "lines containing the string      Text
        'dog', 2 or more times"
```

```
target: ".*(dog){2,}.*"                   Code
```

Natural Language to SQL

```
db_id:    "department_management"         Text
question: "How many heads of the
           departments are older than 56 ?"
```

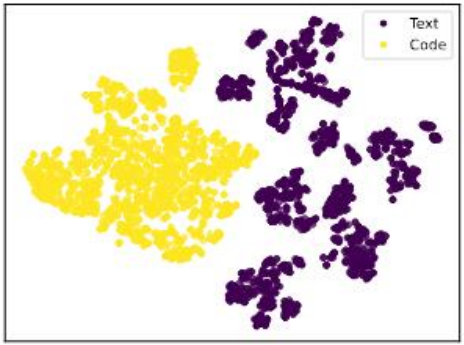
```
query:    "SELECT count(*)                Code
           FROM head WHERE age > 56"
```

Application of JEPA

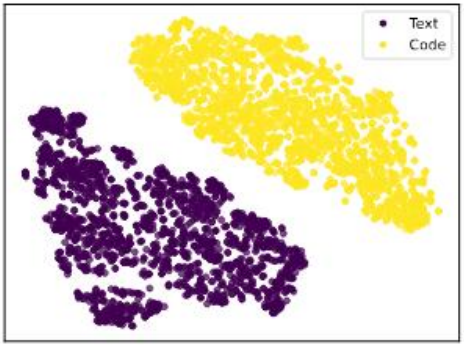
- LLM-JEPA: LARGE LANGUAGE MODELS MEET JOINT EMBEDDING PREDICTIVE ARCHITECTURES **(ICLR 2026, Hai Huang, Randall Balestriero)**
  - 실제 성능도 올라갈 뿐만 아니라, T-SNE 로 시각화시 Hidden State 가 잘 Align 된 것을 알 수 있다.

Table 3: Fine-tuning accuracy on NL-RX-SYNTH under different design choices. Reported are average accuracy and standard deviation across runs. Learning rate $lr = 2e{-}5$, $\lambda = 1.0$, and $k = 1$.

Accuracy (%) ↑						
Baseline	LLM-JEPA	ℓ_2-norm	MSE	Prepend	Code → Text	InfoNCE loss
57.29 ± 5.32	71.46 ± 1.34	2.22 ± 0.07	70.64 ± 2.05	68.07 ± 2.57	65.70 ± 2.63	34.40 ± 6.10



(a) Baseline: Fine-tuned by NTP loss



(b) LLM-JEPA (Ours) $k = 0$

Application of JEPA

- Lens-JEPA: Physics Informed Joint Embedding Predictive Architecture for Gravitational Lensing (NeurIPS 2026 Workshop: ML and the Physical sciences, IIT)
  - I-JEPA based 로 중력렌즈 이미지를 분석하는 Application 모델.
  - 중력렌즈 기반의 데이터를 예측 할 때, 기존의 중력렌즈 Equation과 결합하여 일관성과 물리법칙을 감지할 수 있도록 만듬.

$$\vec{S} = \vec{I} - \nabla\Psi(\vec{I}), \tag{1}$$

$$\Psi(x_i, y_i) = k(x_i, y_i) \cdot \Psi_{\text{SIS}}(x_i, y_i), \tag{2}$$

$$\Psi_{\text{SIS}}(x_i, y_i) = \sqrt{x_i^2 + y_i^2}, \tag{3}$$

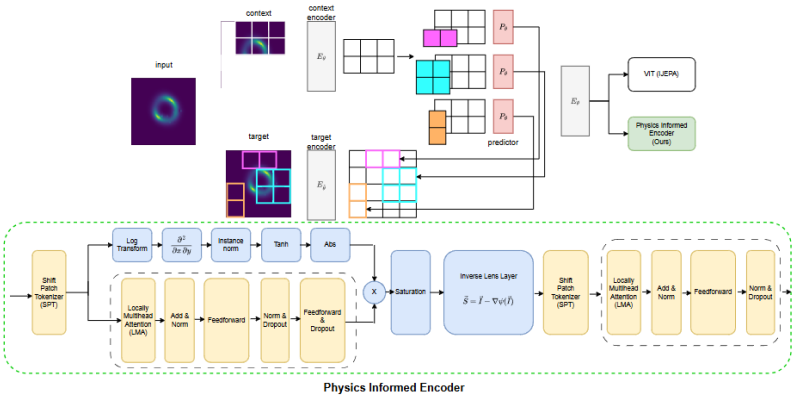


Table 1: Performance Metrics of Different Models

Model	Model B			
	Accuracy	Roc AUC Score		
		Axion	CDM	No Subs
Resnet18	0.8207	0.94	0.88	0.97
ViT	0.8931	0.96	0.94	0.99
CaiT	0.8065	0.94	0.85	0.95
ViTSD	0.8838	0.96	0.93	0.99
Lensformer	0.8969	0.94	0.92	0.98
I-JEPA	0.9017	0.96	0.95	0.98
Lens-JEPA	**0.9120**	**0.97**	**0.96**	**1.00**

Application of JEPA

- REPRESENTATION LEARNING FOR SPATIOTEMPORAL PHYSICAL SYSTEMS (ICLR 2026 Workshop: AI & PDE, Flatiron Institute (미국 기초과학 컴퓨팅 연구소), **Yann Lecun**)
  - 물리량이 포함된 Multi-channel SpatioTemporal Data를 입력으로 받아 물리량(활성물질(active matter), Shear Flow(전단류), 열대류)을 예측하는 모델
  - 픽셀을 복원 하는 모델 대비 좋은 성능을 보임.

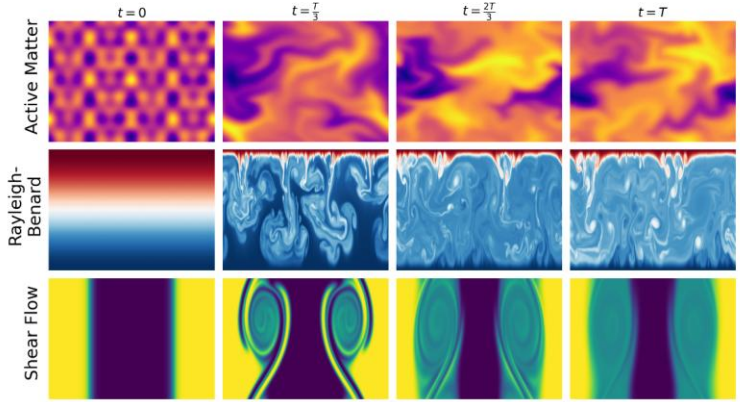


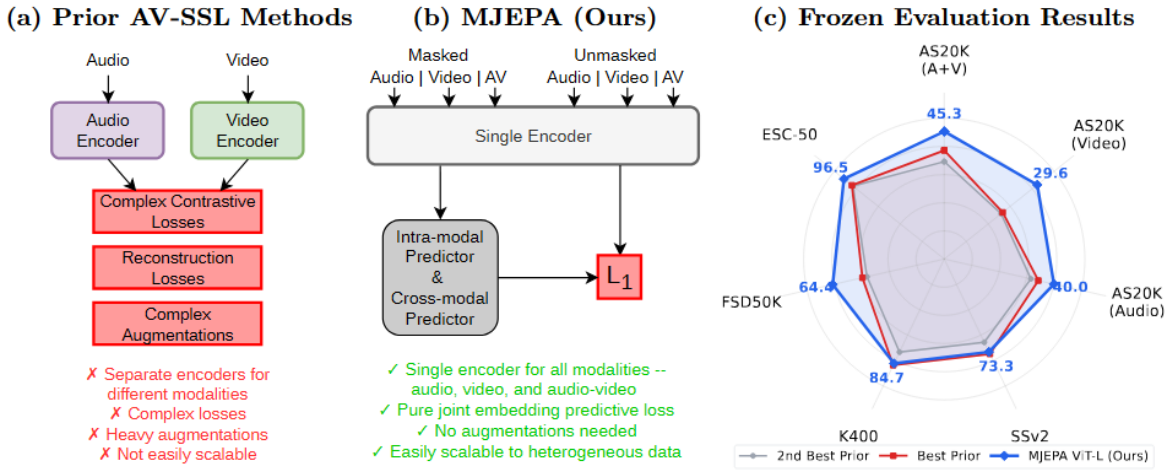
Figure 1: Example trajectories from the physical systems in our evaluation.

	MSE (↓)		
	active matter	shear flow	Rayleigh-Bénard convection
JEPA	**0.079**	**0.38**	**0.13**
VideoMAE	0.160	0.67	0.18
DISCO	**0.057**	**0.13**	**0.01**
MPP (full finetuning)	0.230	0.59	0.08

	fine-tuning dataset fraction		
	10%	50%	100%
JEPA	**0.57**	**0.40**	**0.38**
VideoMAE	0.98	0.75	0.67

Application of JEPA

- MJEPA: A Simple and Scalable Joint-Embedding Predictive Architecture for Audio-Visual Learning (arXiv 26.06, META, NYU)
  - V-JEPA에서 확장되어서 Audio 와 Video를 Single Encoder 에 넣고 통합으로 학습하는 JEPA



Application of JEPA

- CF-JEPA: Mask-free forward prediction with asymmetric encoder utilization for time-series representation learning (arXiv 2026.05, 심성현 교수님)
  - IJEPA 형태를 기반으로 해서, Masking 기반의 방법론 대신 Forward Prediction + Random Cropping 을 하여 시계열 흐름이 깨지지 않는 상태의 Time Series Representation 을 학습하여 Classification, Anomaly Detection, Forecasting을 수행하는 논문

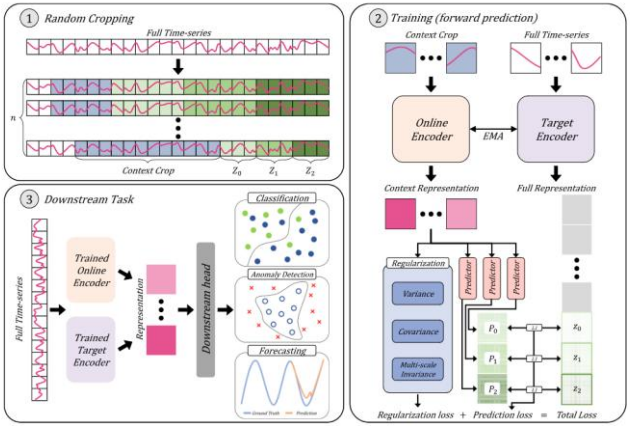


Fig 1. Training pipeline and downstream deployment of CF-JEPA.

Q&A

Q&A