

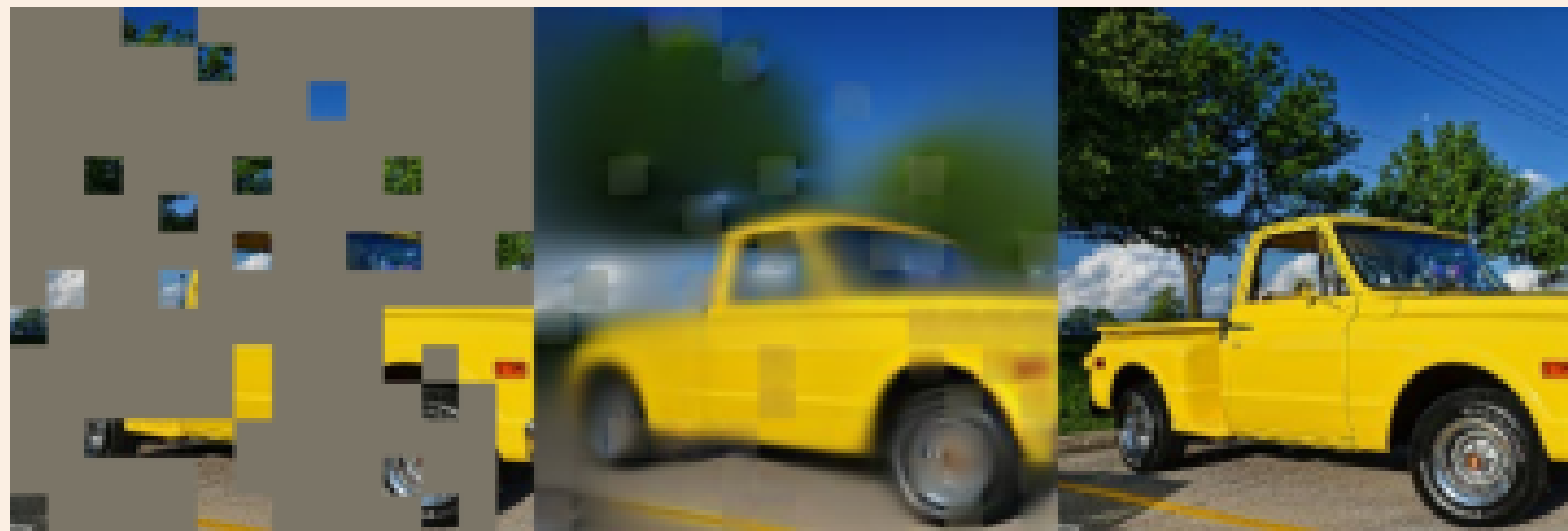
CIS 6280 · FALL 2026

WORLD MODELS

LECTURE 6

Representation Learning II

L5 Recap: Representations and MAE



MAE: visible patches · reconstruction · original

Course announcements: Last class ended at MAE. Today begins with AIM.

A representation: $z = f_{\theta}(o)$ — another way to represent an observation.

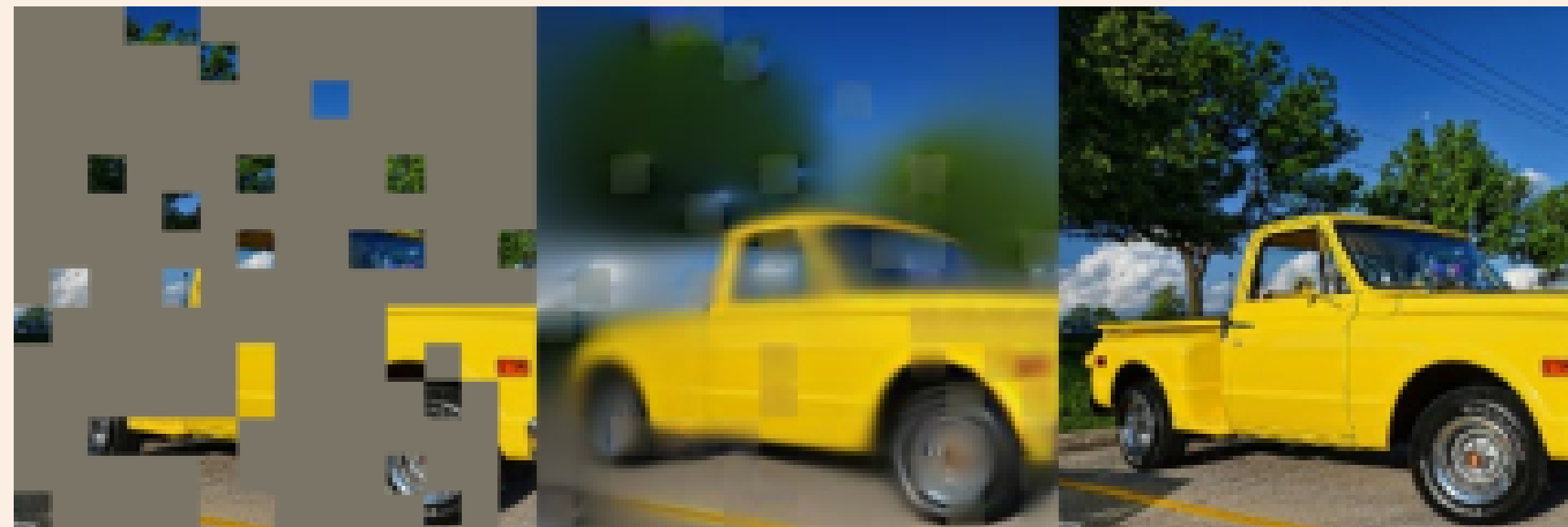
Learning: the training task rewards some information.

Testing: freeze the encoder; train a small readout.

Where we stopped: hide patches and predict their pixels.

Today: matching views and predicting learned feature targets.

Reconstruction-Based Learning



Corrupted input · reconstruction · original target

For MAE, ℓ measures error only at the masked patches.

$$\mathcal{L}_{\text{rec}} = \ell(D(E(T(x))), x)$$

T: corrupt the input, e.g. hide patches or add noise.

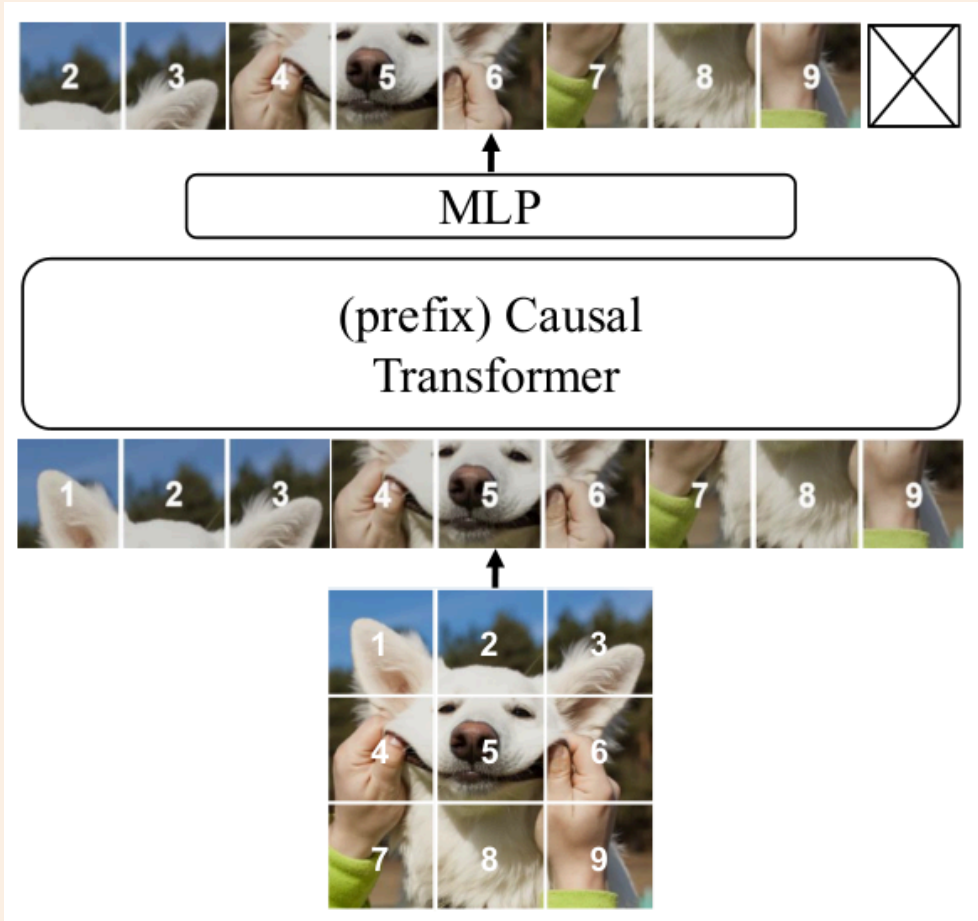
E: encode the corrupted input into features.

D: reconstruct the original data from those features.

ℓ : compare the reconstruction with the original x .

The encoder learns features that help recover information removed by T.

AIM



Original AIM architecture: an image is a sequence of patches.

$$\mathcal{L}_{\text{AIM}} = \sum_{i \in \text{scored}} \|\hat{x}_{i+1}(x_{\leq i}) - x_{i+1}\|^2$$

The image encoder must use visible context to predict the next patch.

AIM: Attention Mask and Targets

Output row sees →	x ₁	x ₂	x ₃	x ₄	x ₅	Next target
Input x ₁	●	●	—	—	—	x ₂ · excluded
Input x ₂	●	●	—	—	—	x ₃ · scored
Input x ₃	●	●	●	—	—	x ₄ · scored
Input x ₄	●	●	●	●	—	x ₅ · scored
Input x ₅	●	●	●	●	●	end

Prefix S = 2: x₁ and x₂ see each other. Later rows see only their prefix and past/current inputs.

The x₂ output predicts x₃ without seeing x₃. Prefix targets are excluded.

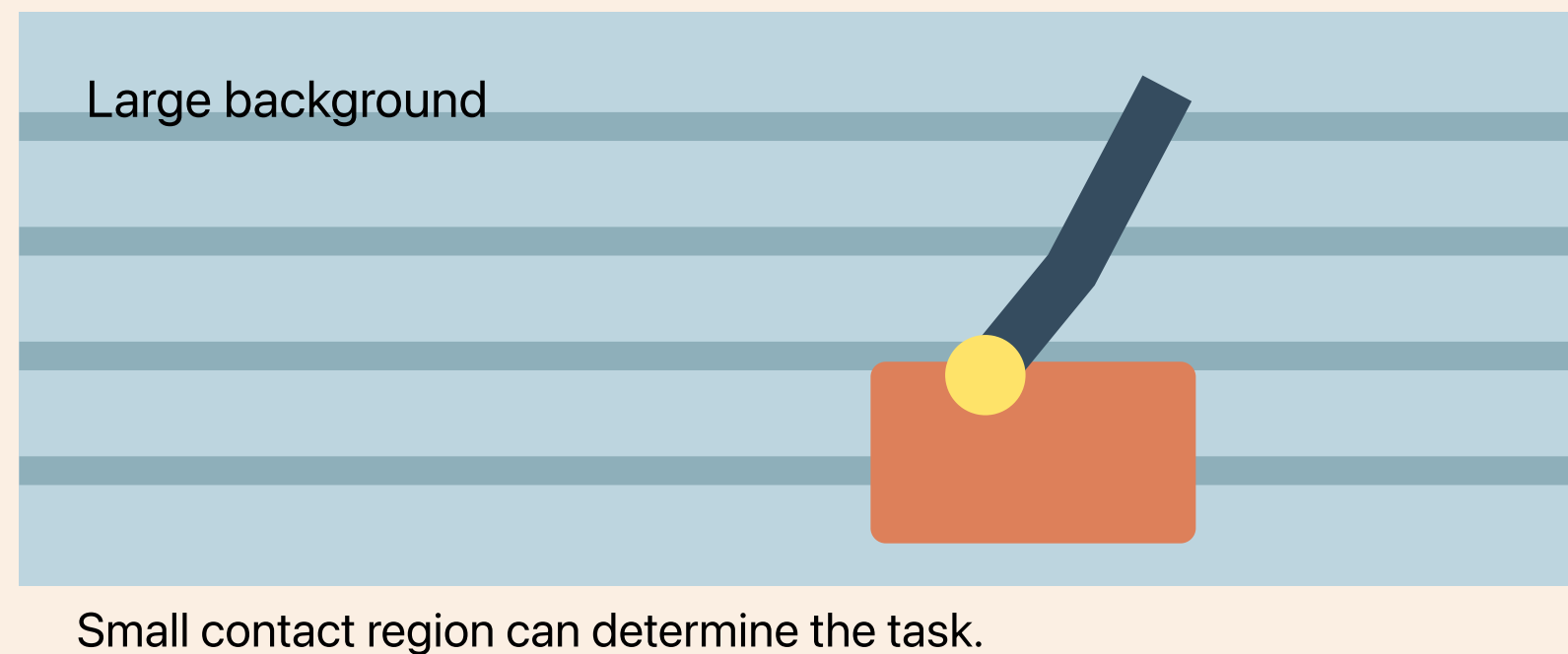
MAE vs. AIM

	MAE	AIM
Visible context	Randomly retained patches	Prefix and earlier patches
Prediction target	Masked patches	Next patch in a chosen scan
Output being scored	Original pixel values	Original pixel values
Transfer	Reuse the encoder	Reuse the encoder

Both can learn useful context. Neither objective directly names “contact,” “pose,” or “velocity.”

The target is similar; the available evidence is different.

Pixel MSE and Task Relevance



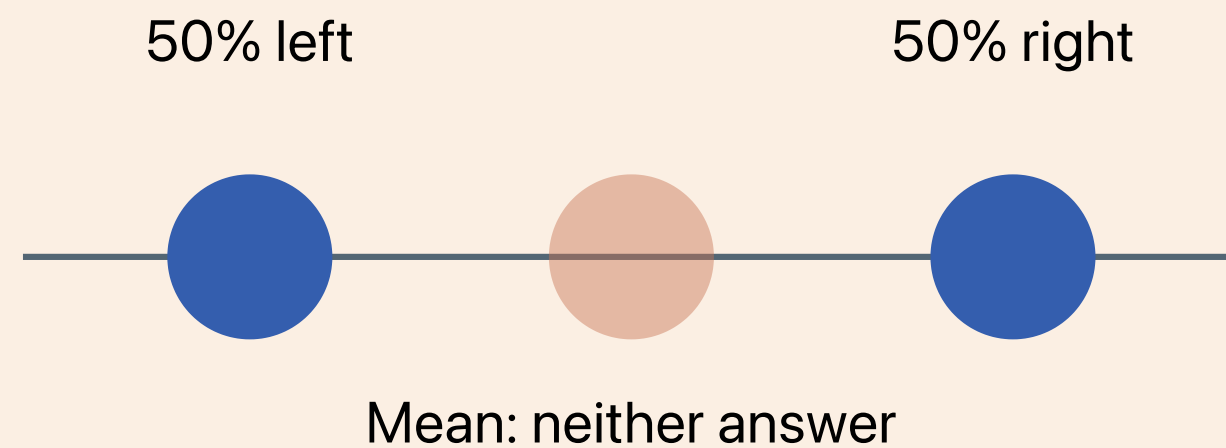
$$\mathcal{L}_{\text{MSE}} = \frac{1}{P} \sum_{p=1}^P (\hat{x}_p - x_p)^2$$

At equal per-pixel error: 9,000 background pixels contribute 90× more than 100 contact pixels.

With limited capacity, reducing a large predictable texture error can pay more than preserving a small task cue.

This is a mismatch of weights and task value, not a rule that MSE always ignores objects.

Point Predictions and Conditional Means



$$\hat{x}^*(c) = \mathbb{E}[x \mid c]$$

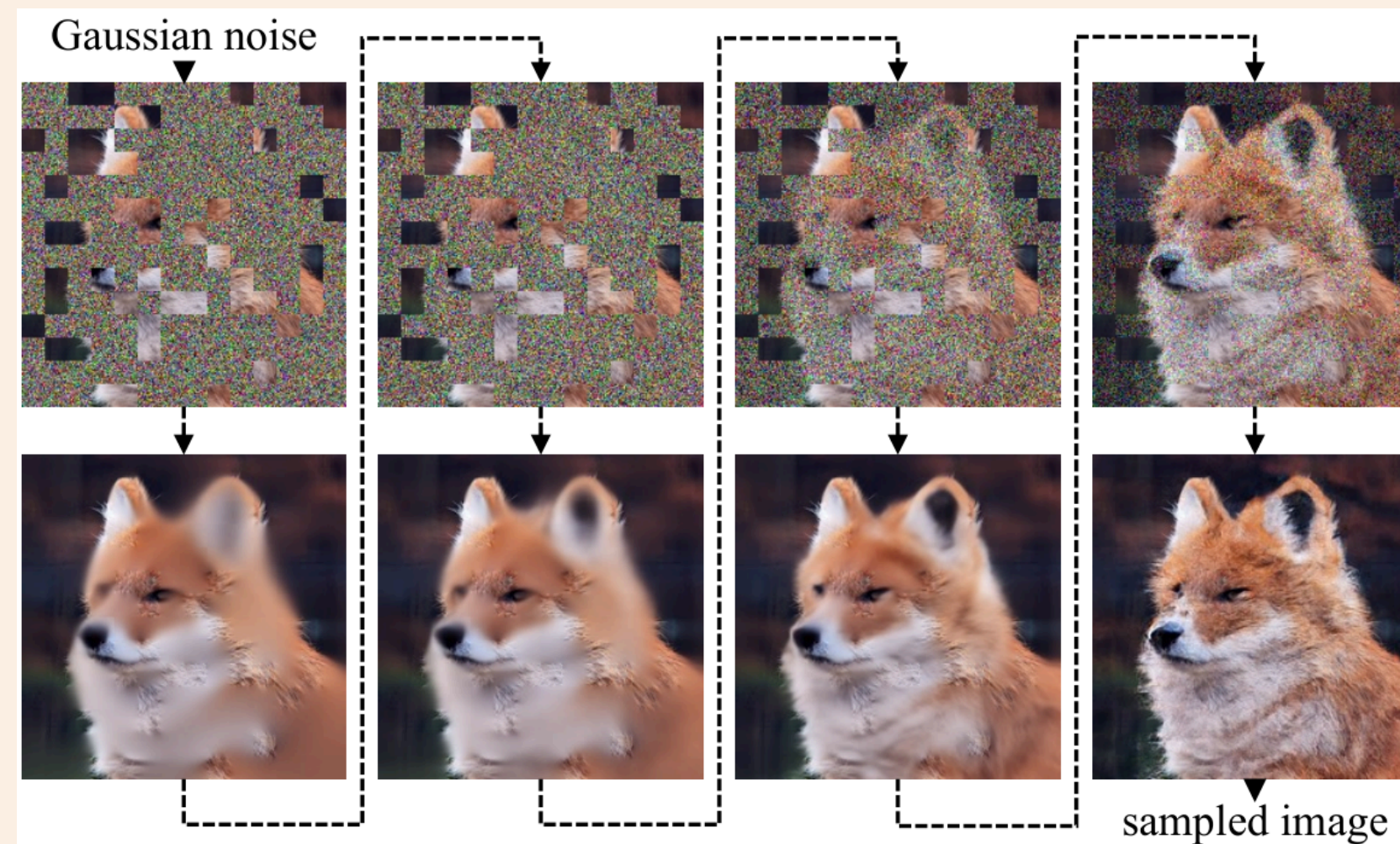
This is the optimal deterministic answer under squared error.

Generative target: $p_\phi(x_T \mid x_C)$

A distribution can assign probability to both possibilities.

Averaging answers and representing uncertainty are different learning problems.

DiffMAE



DiffMAE Figure 1 • successive denoising outputs

Visible patches condition a sequence of denoising steps.

A sampled completion can choose a coherent outcome.

The encoder is also evaluated through downstream recognition.

(We will talk about diffusion in later lectures.)

Generative reconstruction can learn useful features; realism alone does not certify them.

Positive Pairs



View 1



View 2

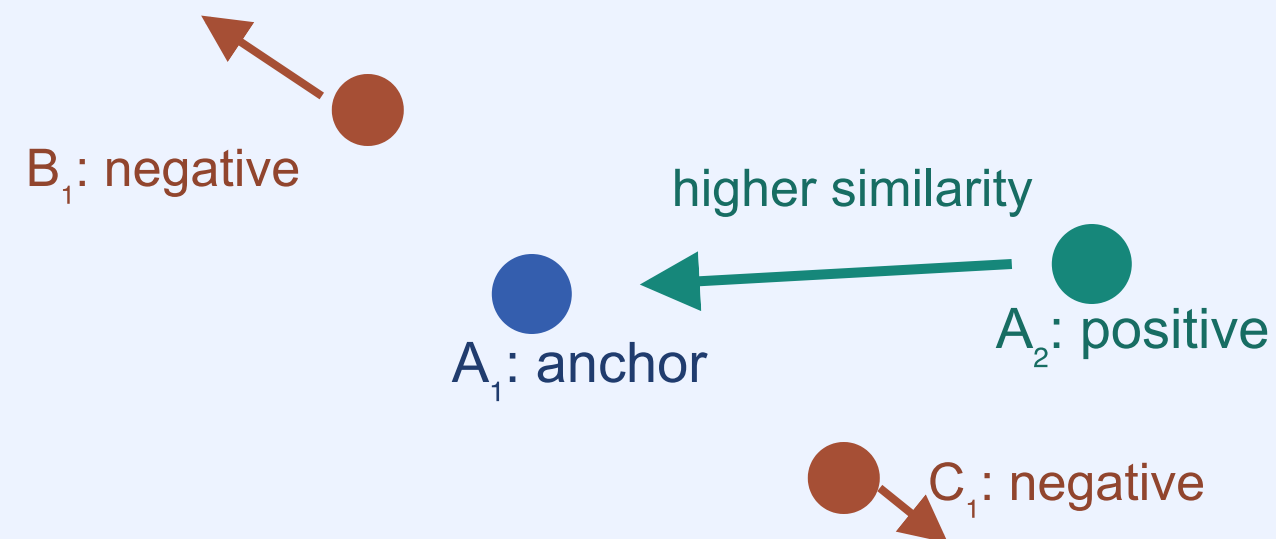
The answer can be a relationship between observations, rather than the missing pixels.

Contrastive learning asks: which views belong together?

Contrastive Learning

Create different views, then learn features that identify the matching view among candidates.

Embeddings of different views



A_1 and A_2 come from the same image A .

B_1 and C_1 come from other images. Positions are schematic.

1. Construct the pairing

Sample two views of one example.

2. Encode and compare

Map each view to an embedding; compute similarity.

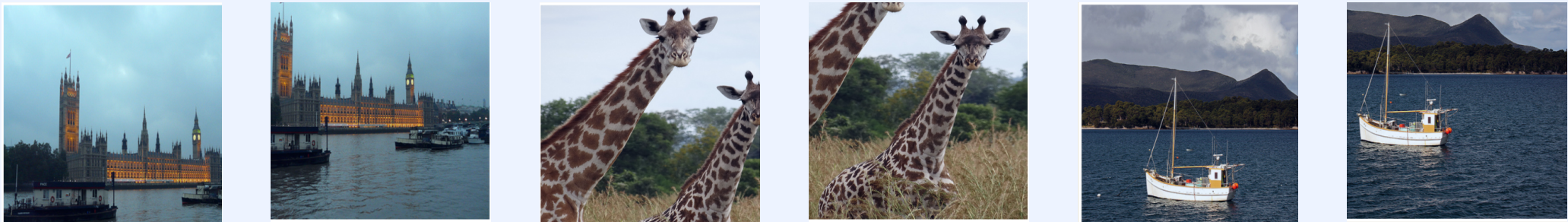
3. Favor the positive over negatives

Train the matched view to receive the higher score.

$$s(A_1, A_2) > s(A_1, B_1), s(A_1, C_1)$$

The target is which views match. The loss compares embeddings, not reconstructed pixels.

Views, Anchors, and Candidates



A₁

A₂

B₁

B₂

C₁

C₂

Anchor ↓ / candidate →	A ₁	A ₂	B ₁	B ₂	C ₁	C ₂
A ₁	—	positive	negative	negative	negative	negative
A ₂	positive	—	negative	negative	negative	negative

For A₁: exclude itself, identify A₂, compare with four other-image views.

InfoNCE

$$u_i = \text{normalize}(g_\phi(f_\theta(v_i))), \quad s_{ik} = u_i^\top u_k$$

$$p_{ij} = \frac{\exp(s_{ij}/\tau)}{\sum_{k \neq i} \exp(s_{ik}/\tau)}, \quad \ell_i = -\log p_{i,j(i)}$$

j(i) is the paired view chosen by the sampler.

τ controls how sharply the scores are compared.

The positive is included in the denominator.

The anchor itself is excluded.

The model wins by making the paired view stand out among the candidates.

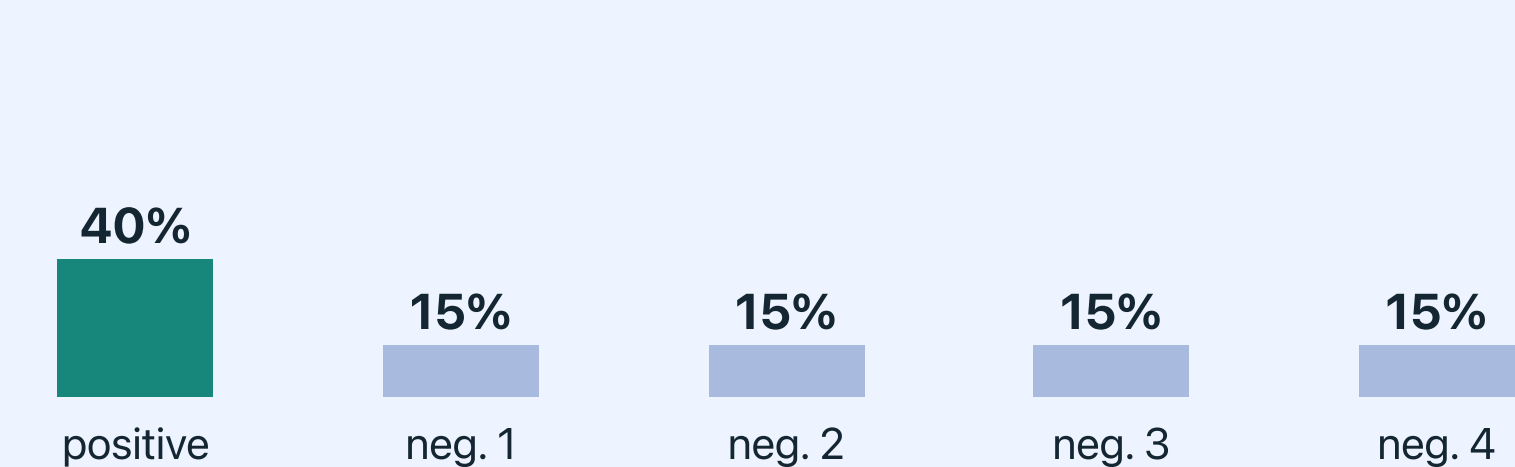
InfoNCE: Temperature

Same similarities: positive = 1; each negative = 0.
Scores ÷ $\tau \rightarrow$ softmax $\rightarrow$ probabilities

Larger τ

$\tau = 1.0$

Scores / τ : 1, 0, 0, 0, 0



A more even distribution

Smaller τ

$\tau = 0.2$

Scores / τ : 5, 0, 0, 0, 0



Positive probability: 97.4%

Smaller τ amplifies score differences. More probability goes to the highest-scoring candidate.

$$\frac{\partial \ell}{\partial s_k} = \frac{p_k - \mathbf{1}[k = +]}{\tau}$$

Positive similarity $\uparrow$; negative similarities $\downarrow$.
Encoder feature $\mathbf{e} \rightarrow$ projector $\rightarrow \mathbf{u}$ for the loss.
Downstream tasks usually reuse $\mathbf{e}$.

Representation Collapse



Image A



Image B

$$f(v) = c \quad \text{for every image and crop}$$

$$\mathcal{L}_+ = \|f(v_1) - f(v_2)\|^2 = 0$$

Every positive pair agrees, but all images look identical in feature space.

$$\ell_{\text{InfoNCE}} = \log(2N - 1) \quad \text{if all scores tie}$$

Negatives make the constant representation fail the matching task.

False Negatives

Example downstream task: cat vs. dog classification.

Training rule: only views from the **same photo** form a positive pair.

Anchor



Photo A · cat

Reference image

Positive



Another crop of photo A

Pull closer

Same source photo

Negative

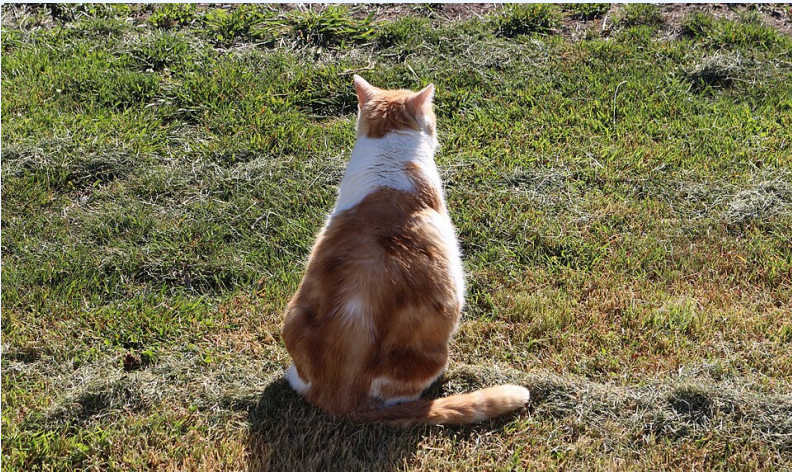


Photo B · another cat

Push apart

False negative:
still the same class!

Negative

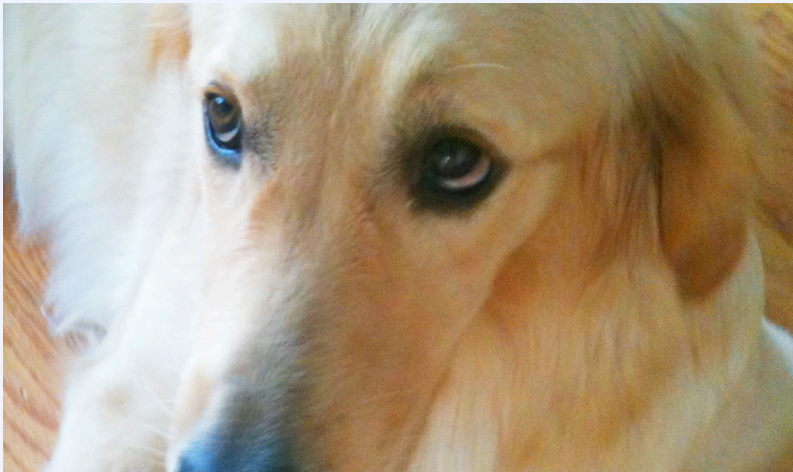


Photo C · dog

Push apart

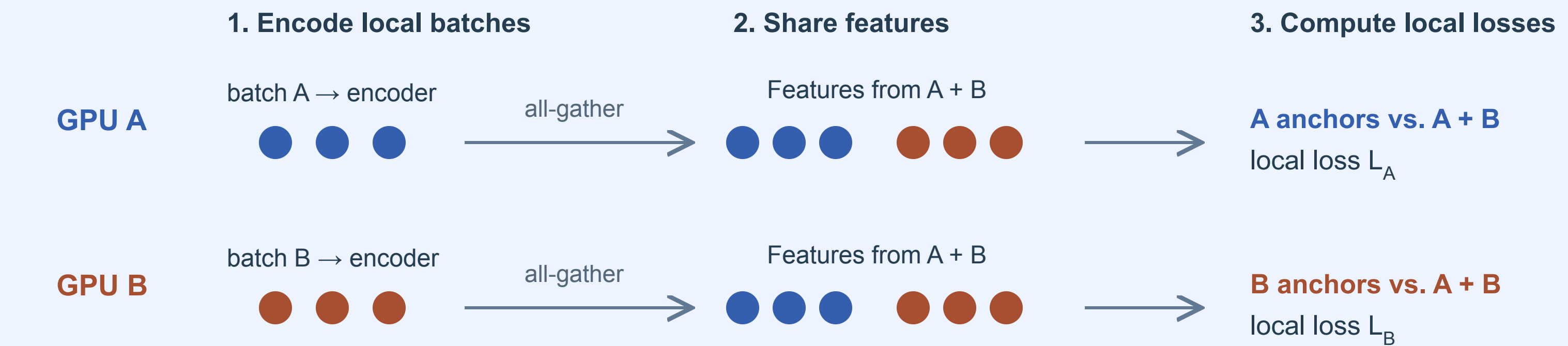
Different class

Another cat and a dog are **both negatives** under this training rule.

A negative pair can share the very class we want the representation to recognize.

Distributed Contrastive Learning

Each GPU keeps its images local and shares **feature vectors** for comparison.



Dots represent features. Each anchor has its own positive; other non-self candidates are negatives. Two GPUs shown for illustration.

4. Backward and update



Share candidates, split the loss computation, and combine gradients for the same model update.

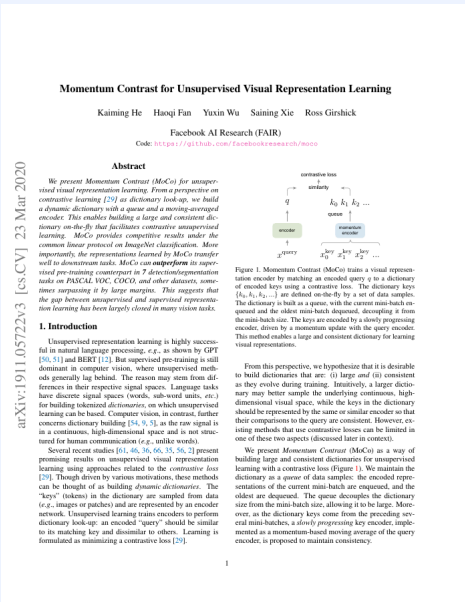
Recommended Reading: MoCo v1 / v2 / v3

MoCo = Momentum Contrast. Optional reading on negative dictionaries and training design.

MoCo v1 2019 / C/PR 2020

MoCo v2 2020

MoCo v3 ICCV 2021



A queue of recent features

A slowly updated key encoder keeps the dictionary consistent.

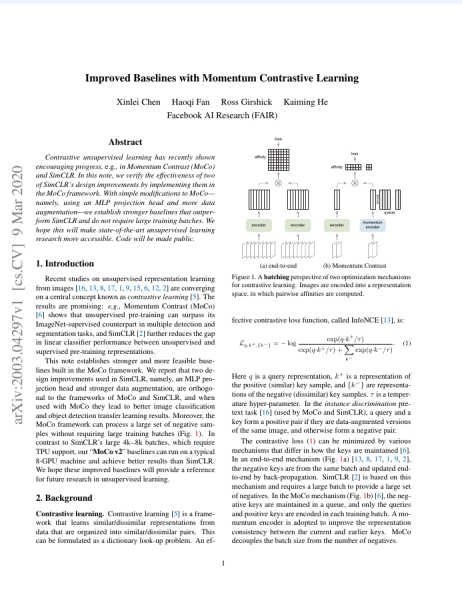
A stronger training recipe

MLP projection head and stronger data augmentation.

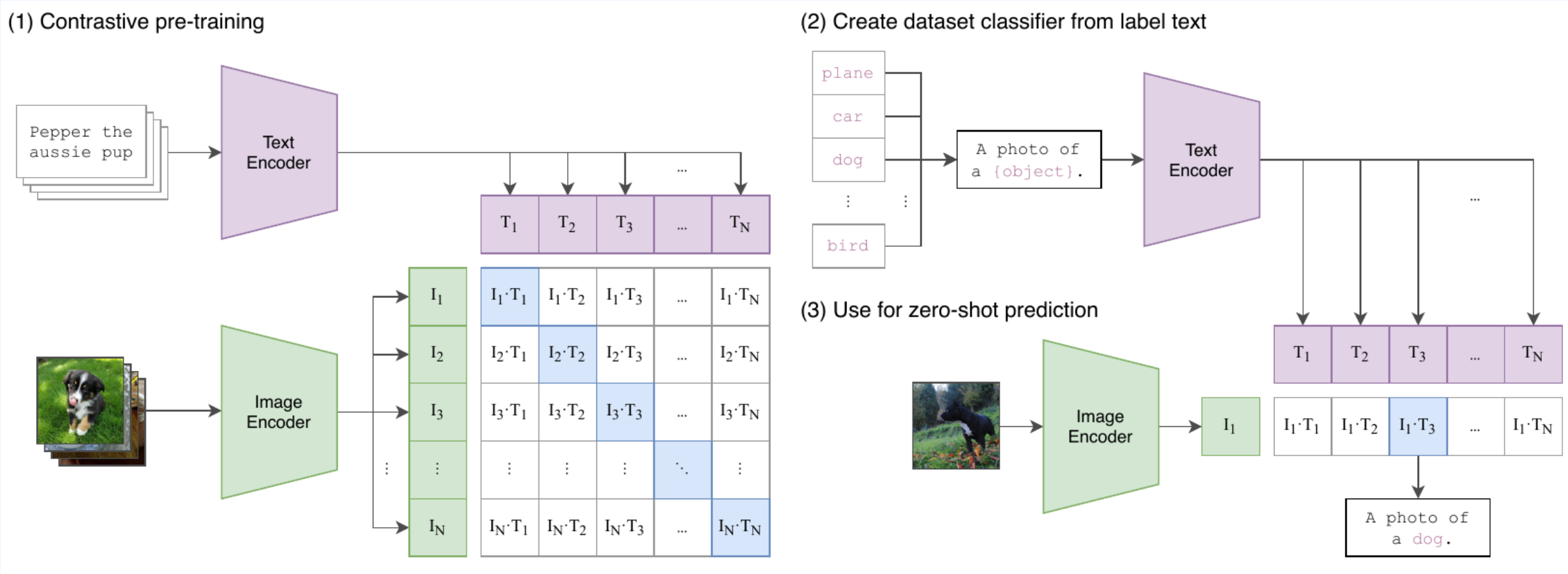
Self-supervised ViTs

Online predictor; current-batch negatives instead of a queue.

MoCo remains contrastive: a momentum encoder changes the targets, while negatives still provide separation.



CLIP



CLIP Figure 1 · image-text contrastive pretraining and transfer

The positive pair is an image and its paired caption. Both encoders receive gradients.

Language changes what is shared: captions emphasize some visual properties and omit others.

CLIP: Zero-Shot Classification



An image to classify

Encode “a photo of a giraffe,” “a photo of a boat,” and other candidate prompts.

$$\hat{k} = \arg \max_k \text{sim}(f_{\text{image}}(x), f_{\text{text}}(t_k))$$

The candidate descriptions specify the decision.

A useful readout is concrete: inputs, candidate labels, and a score.

SigLIP



One image, two candidate captions

Image-text pair	Training label	Sigmoid score
"A photo of a giraffe."	Match (+1)	0.88
"A photo of a boat."	Mismatch (−1)	0.27

Illustrative scores. They do not have to sum to 1.

CLIP: softmax over candidate captions.
SigLIP: a separate binary decision for each image-text pair.

Independent pair scores, with both positive and negative training pairs.

Contrastive Learning: Strengths and Limitations

Compared with reconstruction	Contrastive learning	Remaining limitation
Prediction target	Relations between views, rather than original pixels	The chosen views determine which information is shared
Feature emphasis	No requirement to reproduce every pixel detail	Useful color, pose or location cues may be suppressed by augmentation
Collapse control	Positive-versus-negative discrimination	False negatives and large candidate comparisons

A useful alternative to pixel targets, with strong assumptions about views and pair labels.

Positive-Only Learning

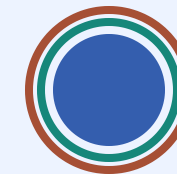
$$\mathcal{L}_+ = \|f(v_1) - f(v_2)\|^2$$

Agreement + image differences



Different colors = different images

Agreement + collapse



All inputs share one feature

$$f(v) = c \text{ for every input} \implies \mathcal{L}_+ = 0$$

Positive agreement alone does not rule out the constant solution.

Learned-Target Learning

How can we match or predict learned representations without negative pairs—and still avoid collapse?

Learning signal: a representation of another view, produced by a network that is also learning.

The shared problem	Two approaches we will study
Naive matching allows every input to share one output	Self-distillation: teacher targets and asymmetric updates. DINO: centering, sharpening and an EMA teacher.
Agreement must preserve differences between inputs	Distribution regularization: explicit constraints on feature variation. VICReg, SIGReg and LeJEPa.
The ingredients can be combined. Each method needs both a useful matching task and a way to resist collapse.	

For every method: what is the target, and what prevents a constant representation?

Joint-Embedding Predictive Architectures (JEPA)



Yann LeCun · photo: Alain Herzog / EPFL · CC BY 4.0

A common JEPA view of learned-target learning

Encode both observations, then match or predict one representation from the other.

| **DINO / BYOL:** direct output agreement / prediction across views.

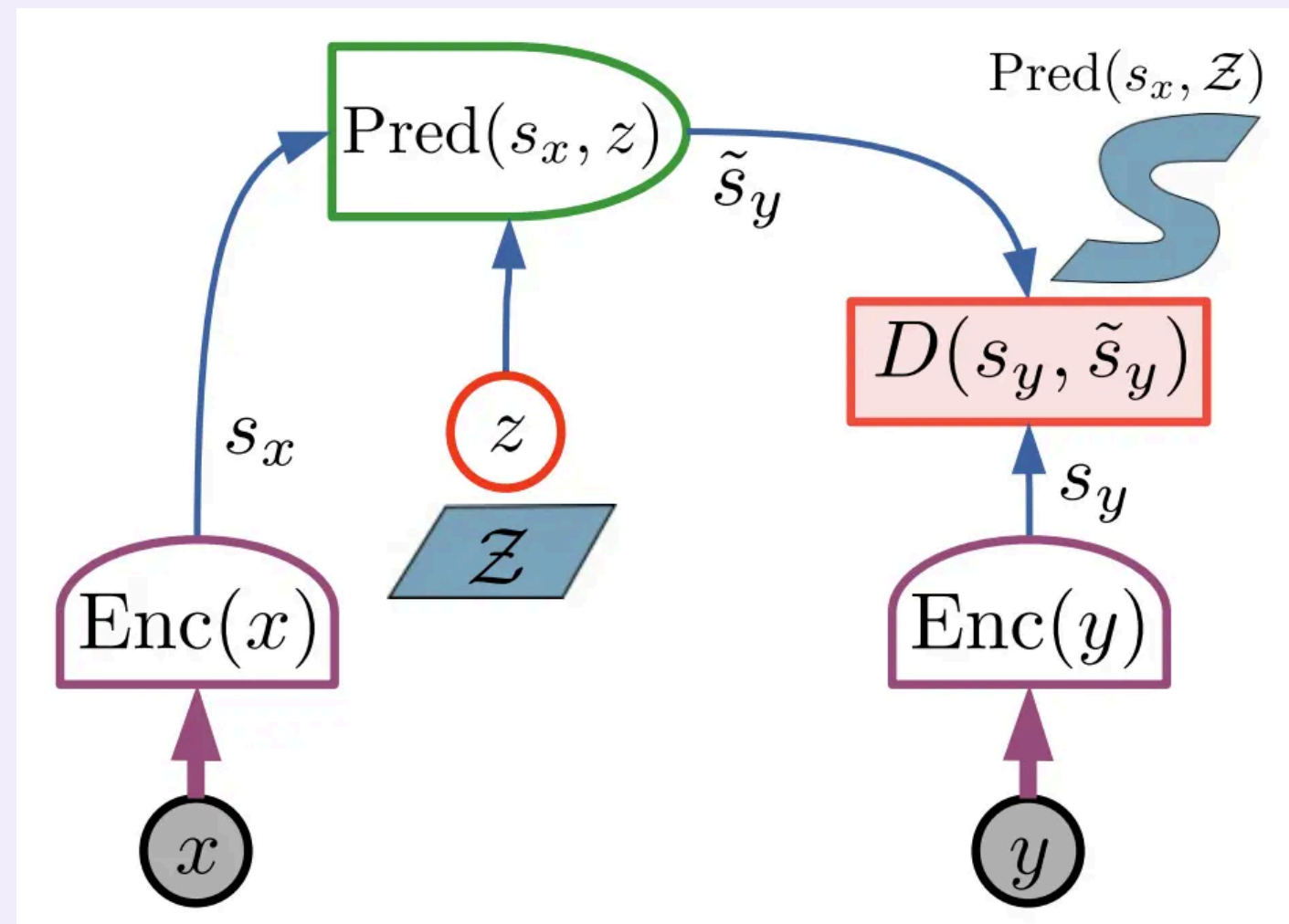
| **data2vec, I-JEPA, V-JEPA:** prediction of target features from partial context.

| **LeJEPA:** view agreement with distribution constraints.

Direct matching is the simplified case; an explicit predictor and latent z are optional.

One architectural perspective; different targets, prediction tasks and ways to prevent collapse.

JEPA: The Original Blueprint



LeCun (2022) · Figure 12 · original JEPA blueprint

$\mathbf{x}, \mathbf{y}$: two related observations.
 $\mathbf{s}_x, \mathbf{s}_y$: their learned representations.

Pred: predict the target representation.

$$\tilde{s}_y = \text{Pred}(s_x, z), \quad E = D(s_y, \tilde{s}_y)$$

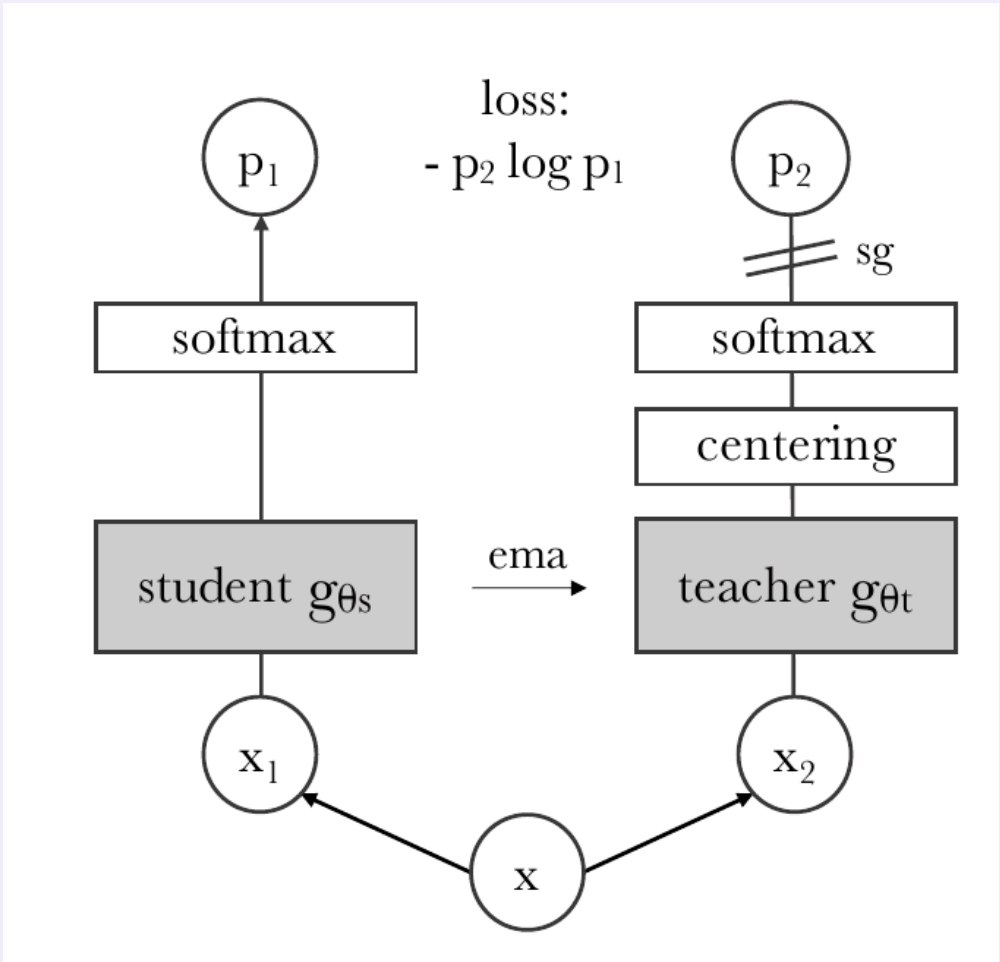
$\mathbf{z}$: a choice among possibilities that x alone cannot determine.

Example: a car approaching a fork can turn left or right. Different z values can represent those alternatives.

Vary z over $\mathcal{Z}$: the blue region is a set of possible predicted features.

Low prediction error is not enough: retain information in the features and limit what z can carry.

DINO: Teacher–Student Training



DINO Figure 2 · follow one teacher–student pair



Teacher: global crop



Student: another crop

Self-distillation: the student learns from targets produced by its own moving teacher.

The teacher starts as a copy of the student.

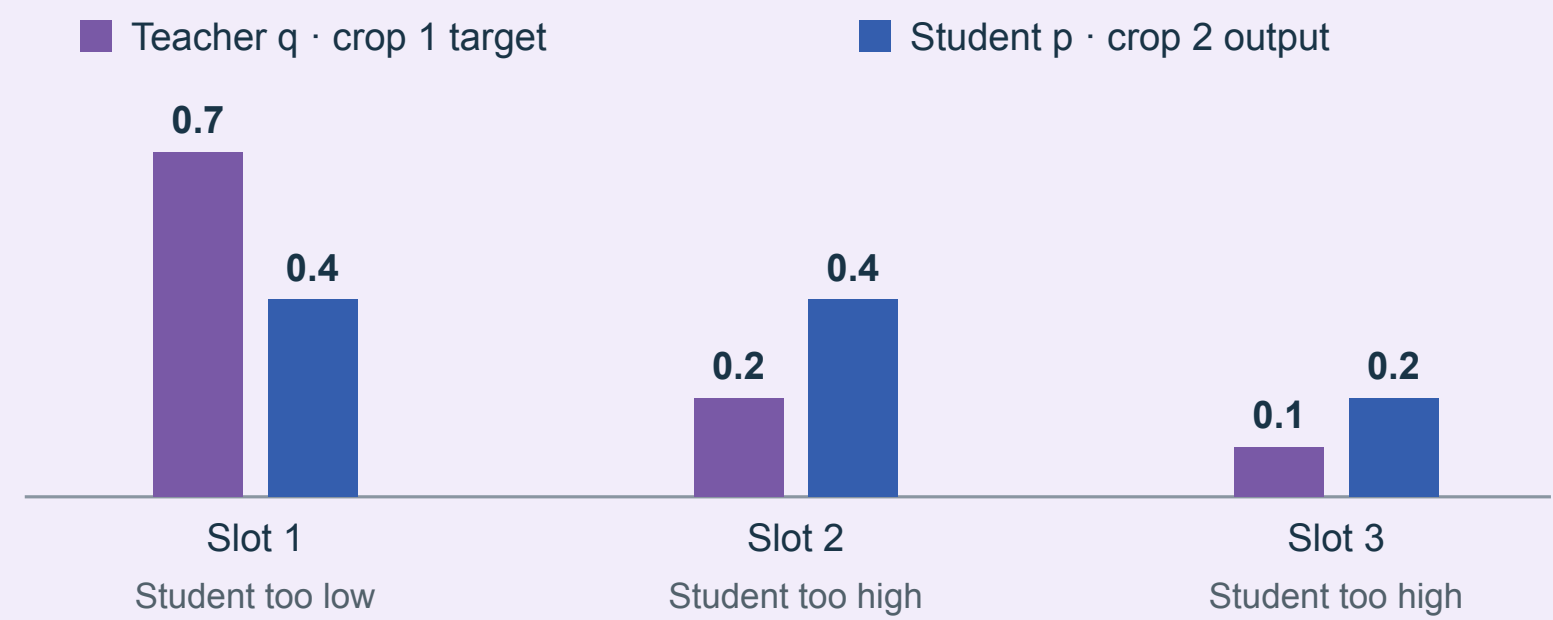
Teacher parameters move slowly with the student; the current loss updates only the student.

View pairing supplies the data relationship. A changing teacher supplies its current target.

DINO: The Output Vocabulary

The head has **K learned output slots**: a “visual vocabulary” with no preset word or class meanings.

q and p: teacher and student probabilities over these K slots; each vector sums to 1.



$$q = \text{softmax}\left(\frac{t - c}{\tau_T}\right) \qquad p = \text{softmax}\left(\frac{s}{\tau_S}\right)$$

t, s: head scores. **c, τ**: centering and temperature, explained next.

Randomly initialized, then learned. K = 3 here only for illustration.

Training goal: student p closer to teacher q. The teacher target is fixed for this update.

DINO: Cross-Entropy Loss

$$\mathcal{L} = - \sum_k q_k \log p_k$$

$$\log p_k = \frac{s_k}{\tau_S} - \log \sum_j e^{s_j / \tau_S}$$

$$\mathcal{L} = - \frac{1}{\tau_S} \sum_k q_k s_k + \log \sum_j e^{s_j / \tau_S}$$

The last step uses $\sum_k q_k = 1$. Teacher q is fixed during this update.

A large teacher probability asks the student to allocate probability to that coordinate.

DINO: Logit Gradients

$$\frac{\partial \mathcal{L}}{\partial s_k} = \frac{p_k - q_k}{\tau_S}$$

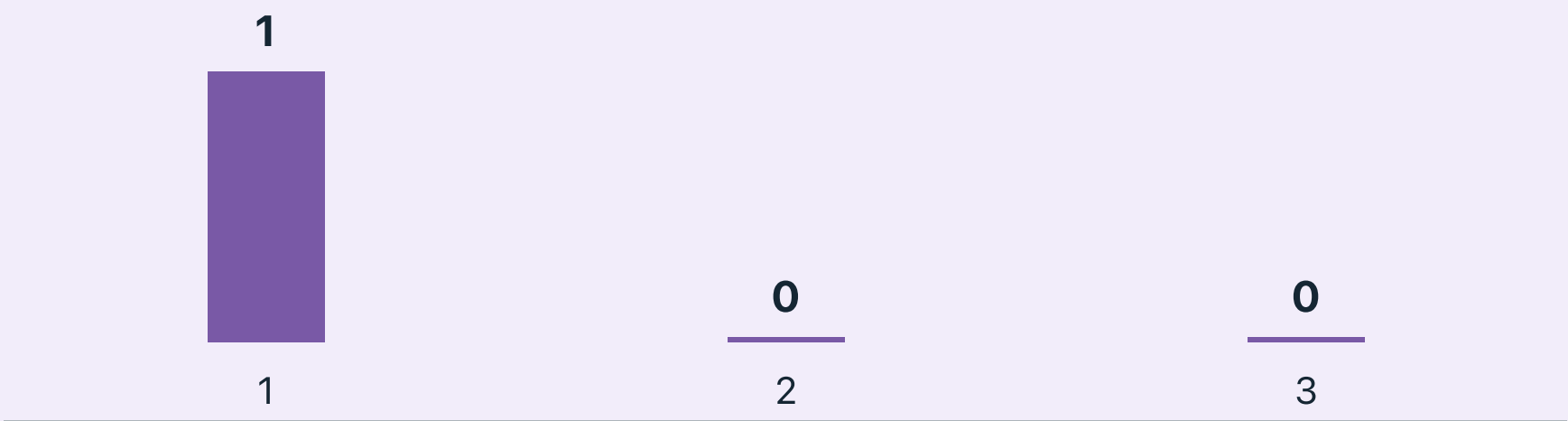
Coordinate	Teacher q	Student p	Gradient descent asks for
1	0.70	0.40	Increase logit 1
2	0.20	0.40	Decrease logit 2
3	0.10	0.20	Decrease logit 3

The derivative of the log-sum-exp term is p_k/τ_S .

The learning signal is a discrepancy between two views' probability vectors.

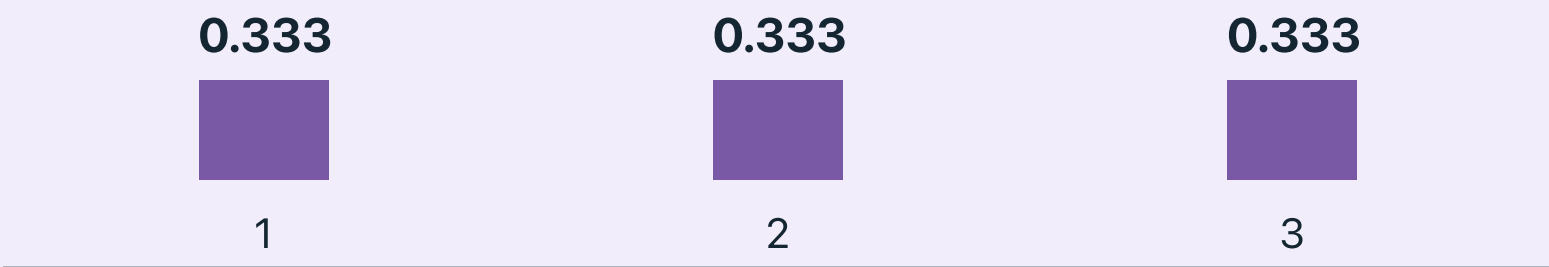
DINO: Collapse Modes

Every image → the same one-hot vector



Confident, yet no distinction between images.

Every image → the same uniform vector



Uncertain, and still no distinction between images.

Matching the teacher is insufficient if the teacher target stops depending on the image.

DINO: Centering

Collapse risk: slot 1 gets the highest teacher score for every image.

Teacher scores t

Input	Slot 1	Slot 2	Slot 3
Image A	8	1	0
Image B	7	3	1
Image C	6	2	2
Column mean	7	2	1



Centered scores $t - c$

Input	Slot 1	Slot 2	Slot 3
Image A	1	-1	-1
Image B	0	1	0
Image C	-1	0	1
Same center c	$c = (7, 2, 1)$		

Illustration: the running center has reached these column means. The highest-scoring slot can now differ by image.

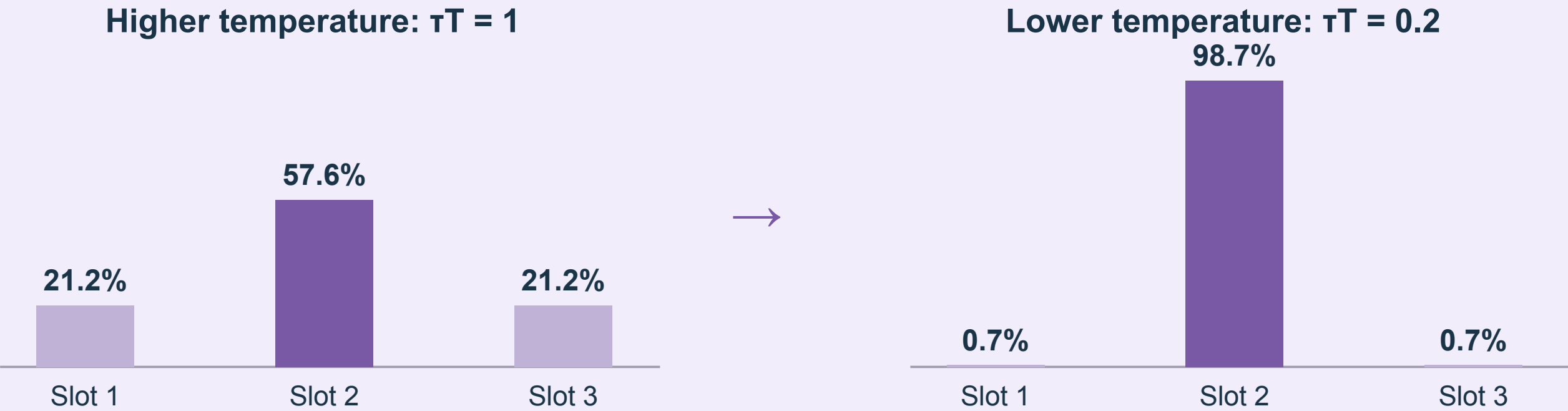
$$c \leftarrow m_c c + (1 - m_c) \underbrace{\frac{1}{B} \sum_b t_b}_{\text{mean across images}}, \quad q = \text{softmax}\left(\frac{t - c}{\tau_T}\right)$$

- Across images:** remove each slot’s persistent bias.
- Within an image:** sharpening (next) makes remaining differences more decisive.

Centering counters a permanently favored slot. On its own, it can still yield uniform collapse.

DINO: Sharpening

Same image B as before: centered teacher scores $t - c = (0, 1, 0)$.



$$q = \text{softmax}((t - c)/\tau_T), \quad \frac{q_2}{q_1} = e^{1/\tau_T}$$

Same preferred slot; a stronger preference at lower temperature.

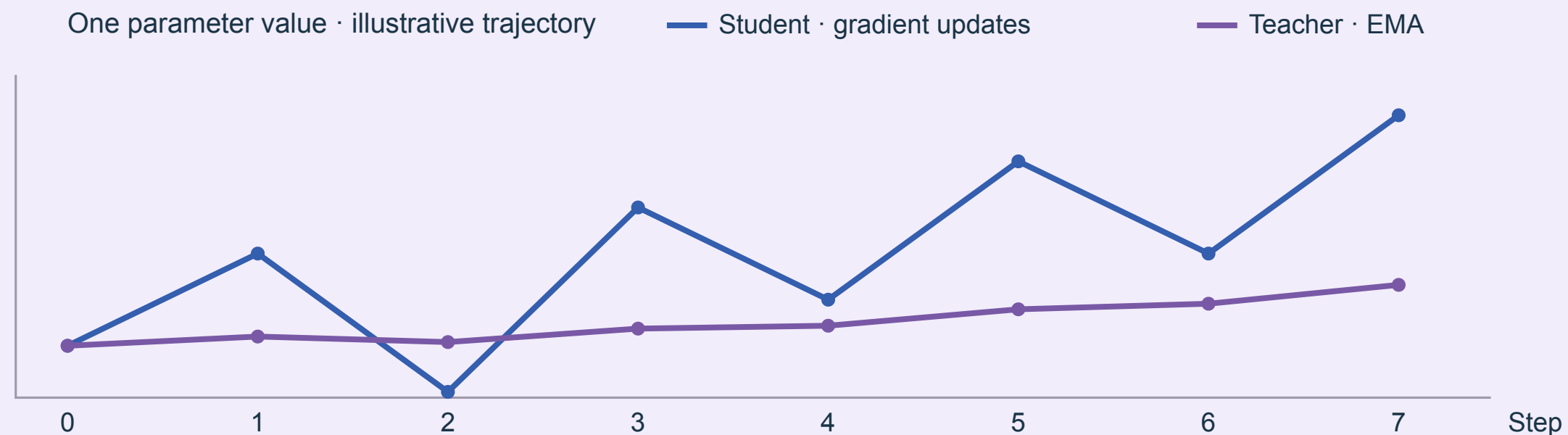
Illustrative temperatures. Exactly tied scores would stay uniform at any temperature.

Centering removes a shared bias across images. Sharpening strengthens each image's remaining preference.

DINO: EMA Teacher

EMA = Exponential Moving Average.

The teacher tracks a smoothed history of the student's parameters.



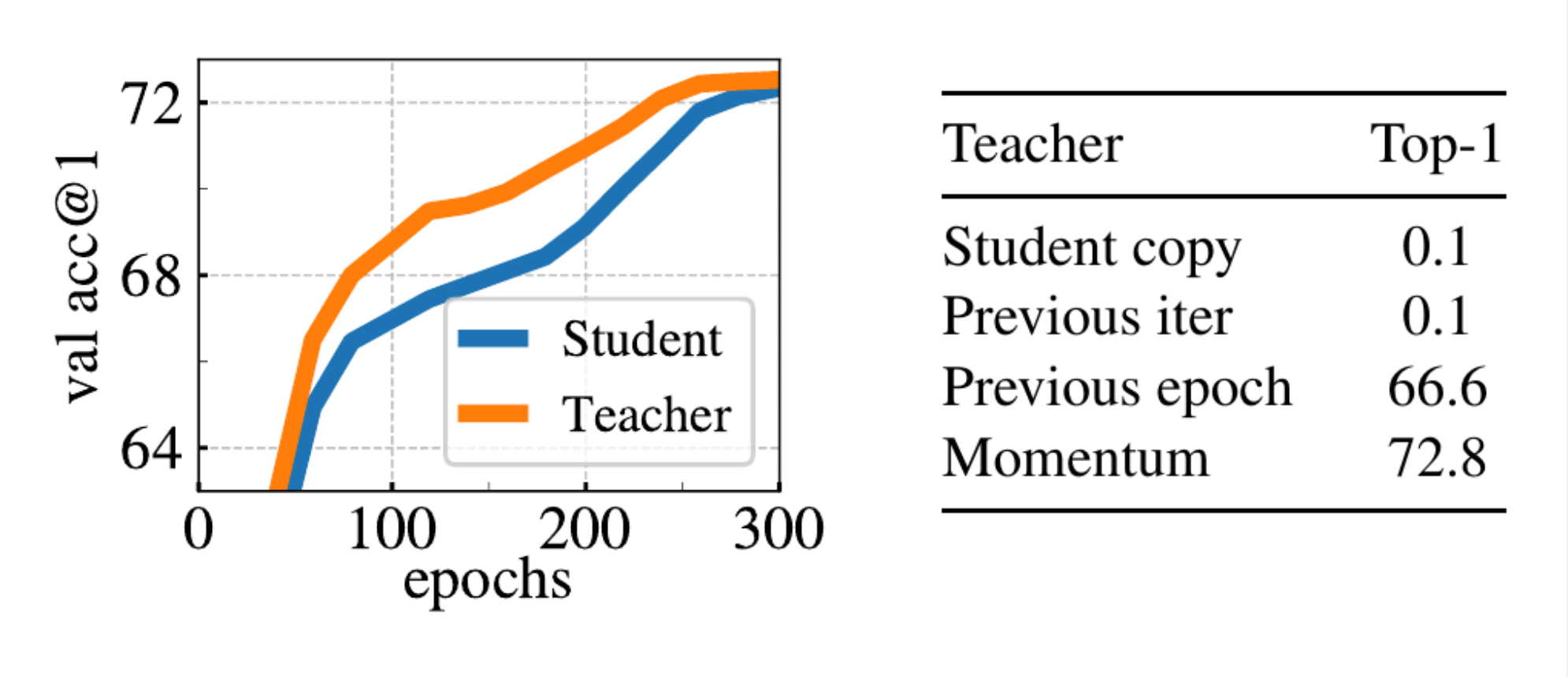
$$\theta_T \leftarrow m\theta_T + (1 - m)\theta_S$$

For $m = 0.9$: 90% previous teacher + 10% updated student.

The current loss updates the student by gradient descent. The teacher changes only through EMA.

A slowly moving target and different update rules for the two branches. EMA alone is not a collapse guarantee.

DINO: Teacher Ablation

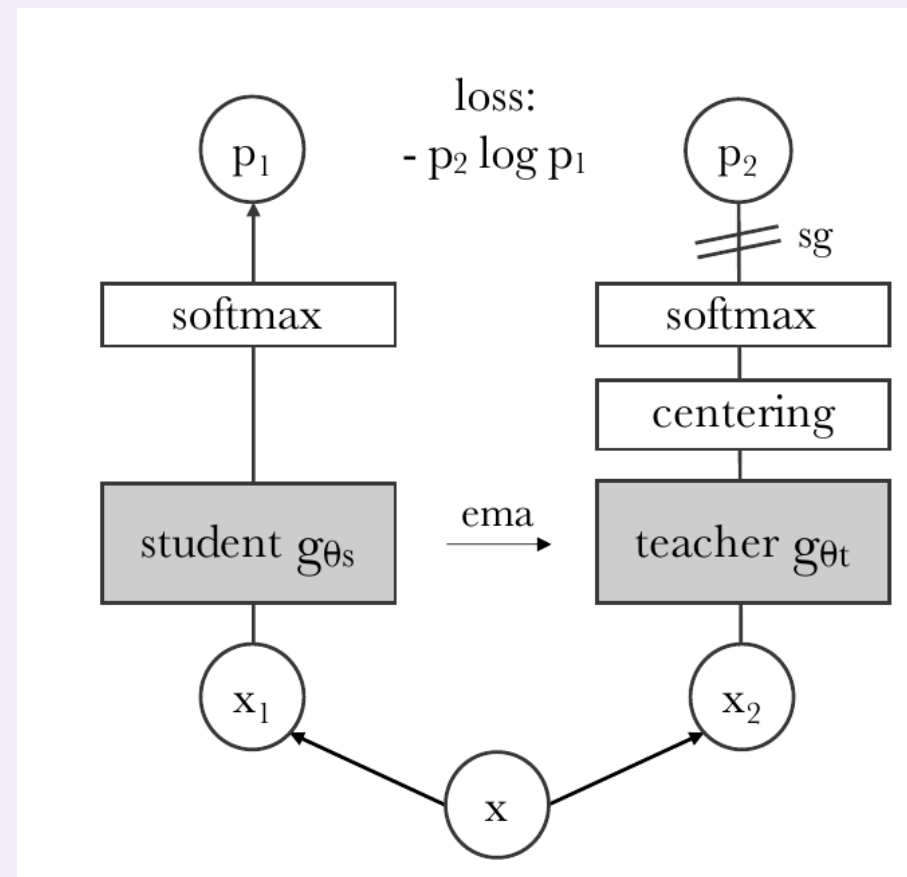


DINO Figure 6 · teacher/student features and teacher-update comparison

Read the evaluation metric: k-NN classification on learned features.

Under this recipe, the EMA teacher provides stronger features than the contemporaneous student.

DINO: Training Step



Locate each operation in the original diagram.

```

1  q = teacher(global_view).center().sharpen()
2  p = student(other_view).softmax()
3  loss = cross_entropy(q.detach(), p)
4  update student by backpropagation
5  update teacher parameters by EMA
6  update center from teacher logits

```

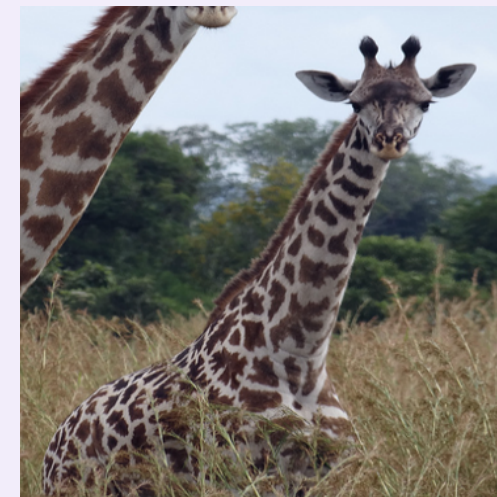
Conceptual pseudocode: the real center update uses raw teacher logits.

Student gradients, teacher EMA, and center statistics are different operations.

DINO: Local-to-Global Matching



Global view

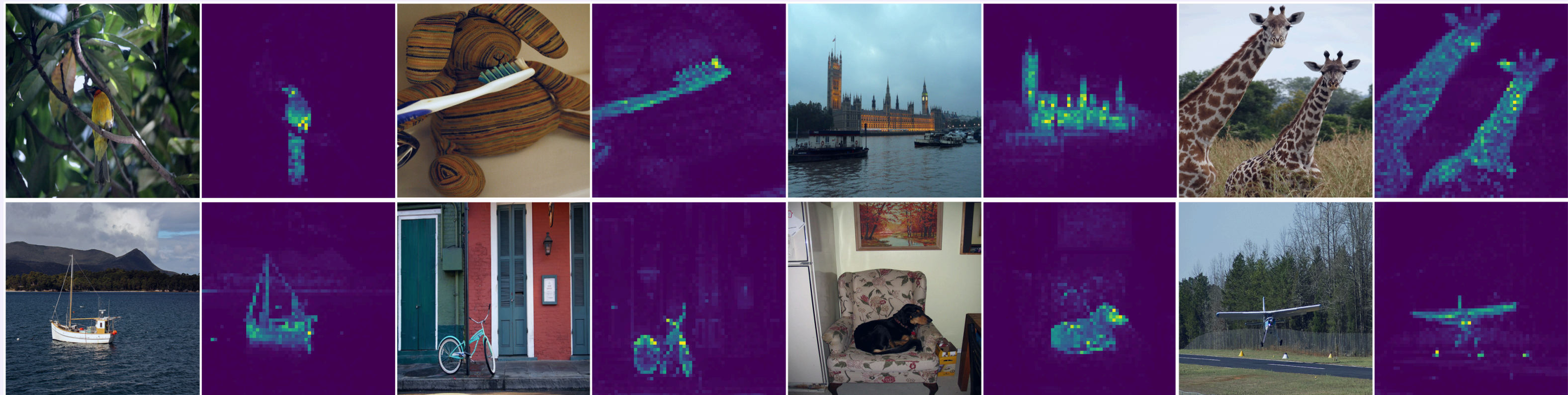


Local view

Local-to-global matching rewards information that predicts the teacher's larger-view target.

For a world model: test whether pose and contact survive, even when object identity is stable.

DINO: Qualitative Results



Original DINO examples · image / attention-map pairs

Each map shows attention from the image-level token to image patches. Brighter regions have higher attention weights.

Training uses no segmentation masks. These maps show qualitative structure; downstream probes test usefulness.

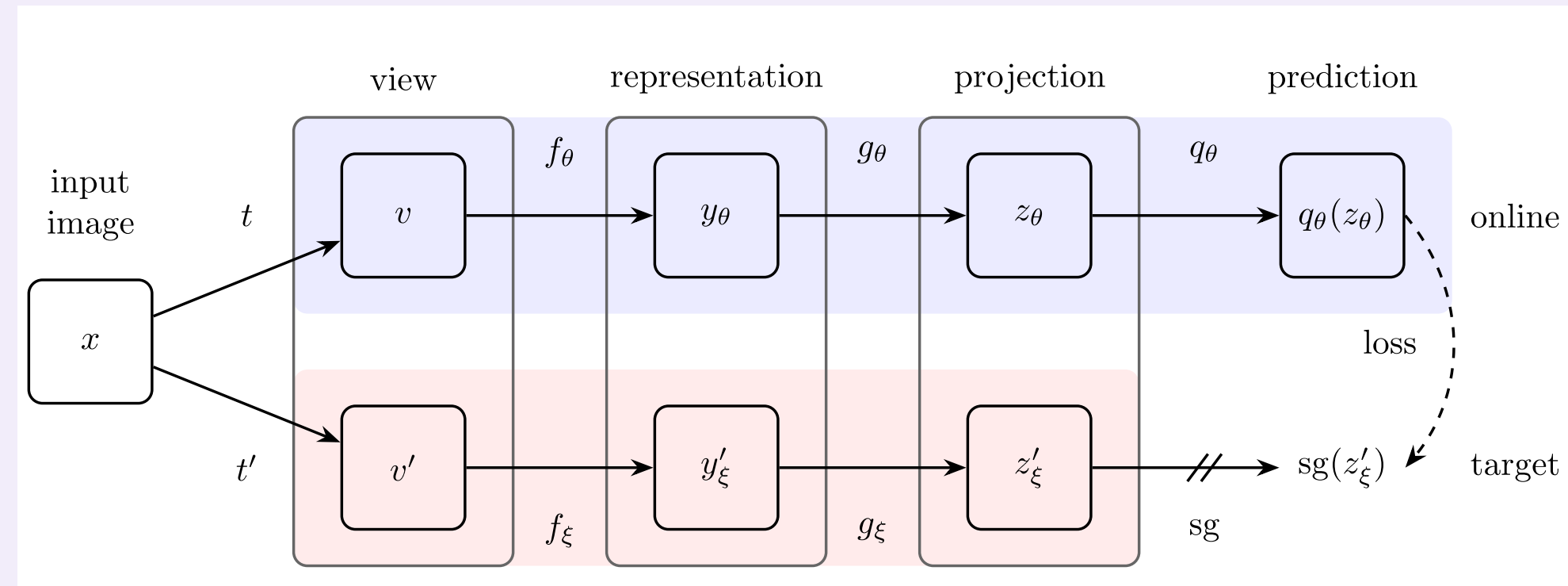
DINO: Attention Maps



Official DINO demonstration · play a short excerpt; static frame retained in PDF.

Coherent attention is evidence to inspect; a downstream test determines what can be read out.

BYOL: Asymmetric Prediction



Grill et al. (2020) · Figure 2 · q_θ is the predictor; z denotes projected features

Online branch: an extra predictor q_θ .

Target branch: EMA weights + stop-gradient.

DINO: match output probabilities.

BYOL: predict the other view's continuous features.

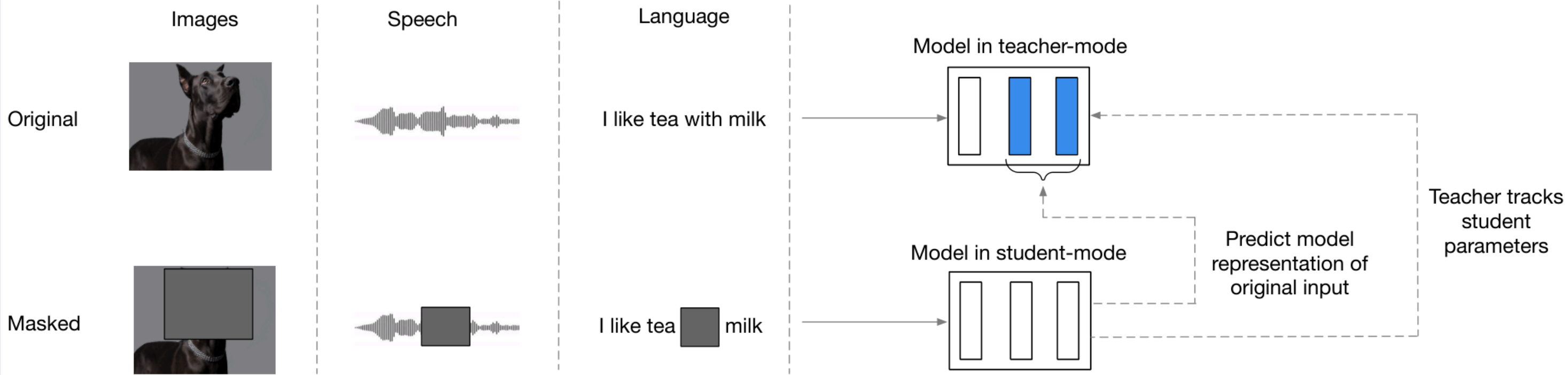
$$\mathcal{L} = \left\| \frac{q_\theta(z_\theta)}{\|q_\theta(z_\theta)\|_2} - \text{sg} \left(\frac{z'_\xi}{\|z'_\xi\|_2} \right) \right\|_2^2$$

Normalize both vectors, then compare.

Also train with the two views swapped.

Predictor on one branch; a stopped, slowly updated target on the other. No negative pairs.

data2vec



data2vec · Figure 1 from the original paper

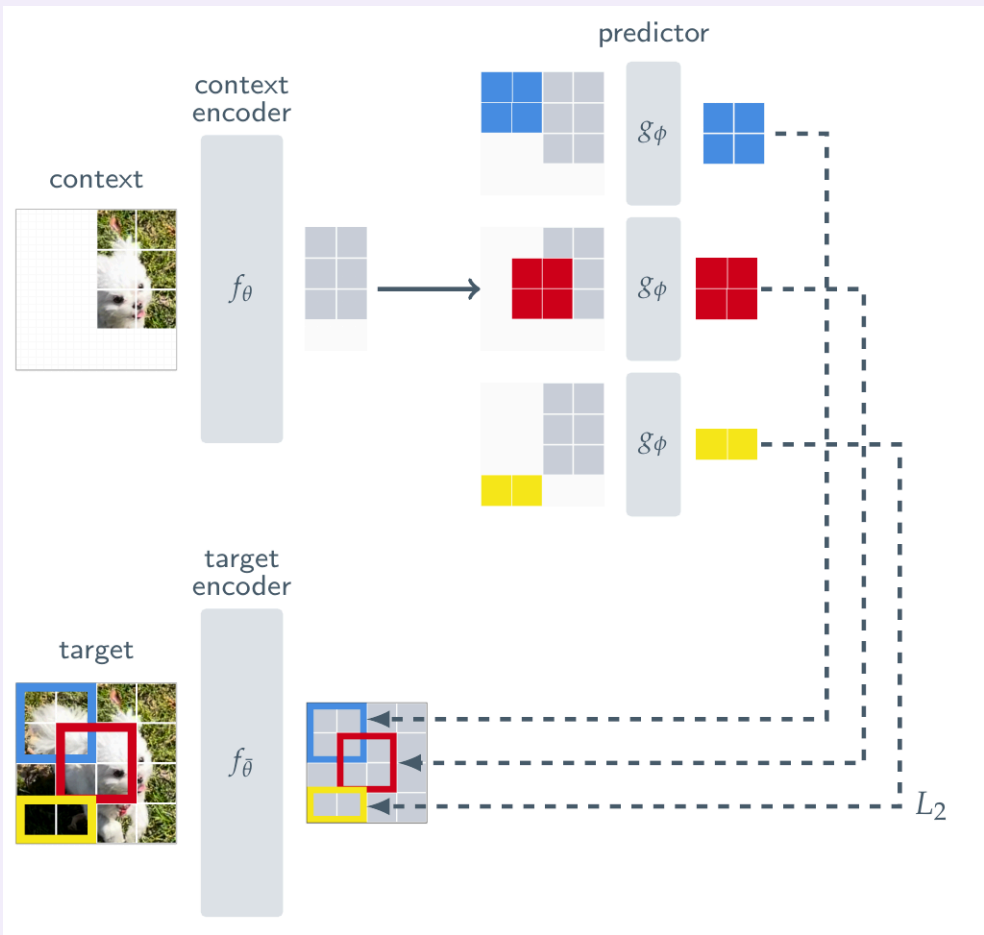
Earlier recipe → masked prediction	DINO	data2vec
Input relationship	Different crops of one image	Masked student input; unmasked teacher input
Training target and loss	Softmax output distribution Cross-entropy matching	Continuous teacher features Regression at masked positions

Teacher target: average normalized features from several upper layers. Each position includes context from the full input.

Like BYOL, continuous teacher targets; now predict masked positions in the input.

I-JEPA

Shared with data2vec: masked feature prediction, full-input teacher targets, and an EMA teacher.



I-JEPA · Figure 3 · follow one colored target block

1. Encoder + predictor

data2vec: masked sequence through encoder + linear output head.
I-JEPA: visible-only context encoder + separate position-conditioned Transformer predictor.

2. Context and target blocks

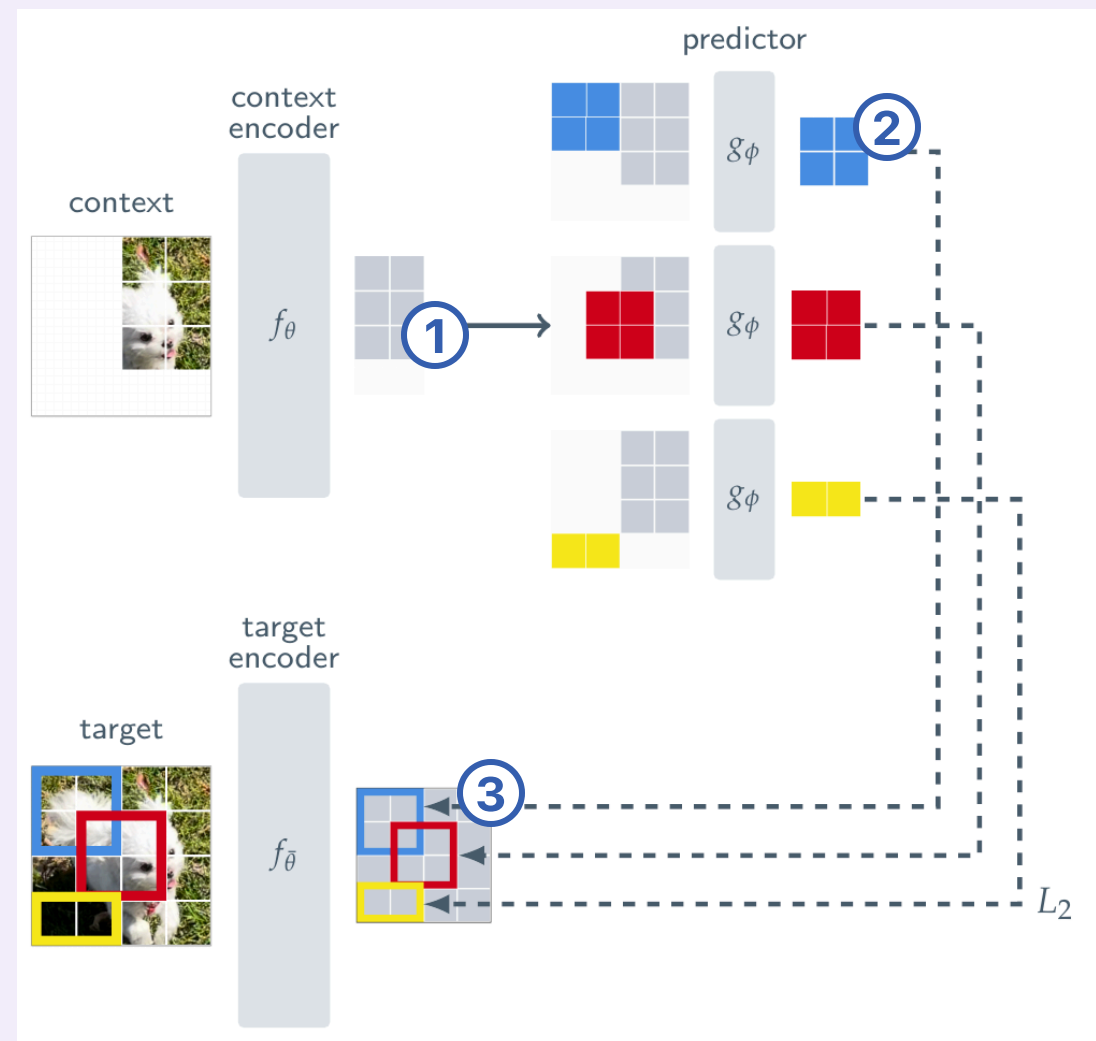
data2vec already uses block masking for images.
I-JEPA explicitly pairs informative context with several large target regions.

3. Teacher features

data2vec: average several normalized upper layers.
I-JEPA: target-encoder output at selected positions.

A closely related learning principle, with a different prediction architecture and masking design.

I-JEPA: Prediction Objective



Original Figure 3 · follow the blue target block

① Visible context

$$h_C = f_\theta(x_C)$$

C: the image patches the student can see.

② Prediction at the blue positions

$$\hat{y}_T = g_\phi(h_C, \text{pos}(T))$$

T: the hidden positions we ask about.

The colored tokens specify where to predict.

③ Teacher features at the same positions

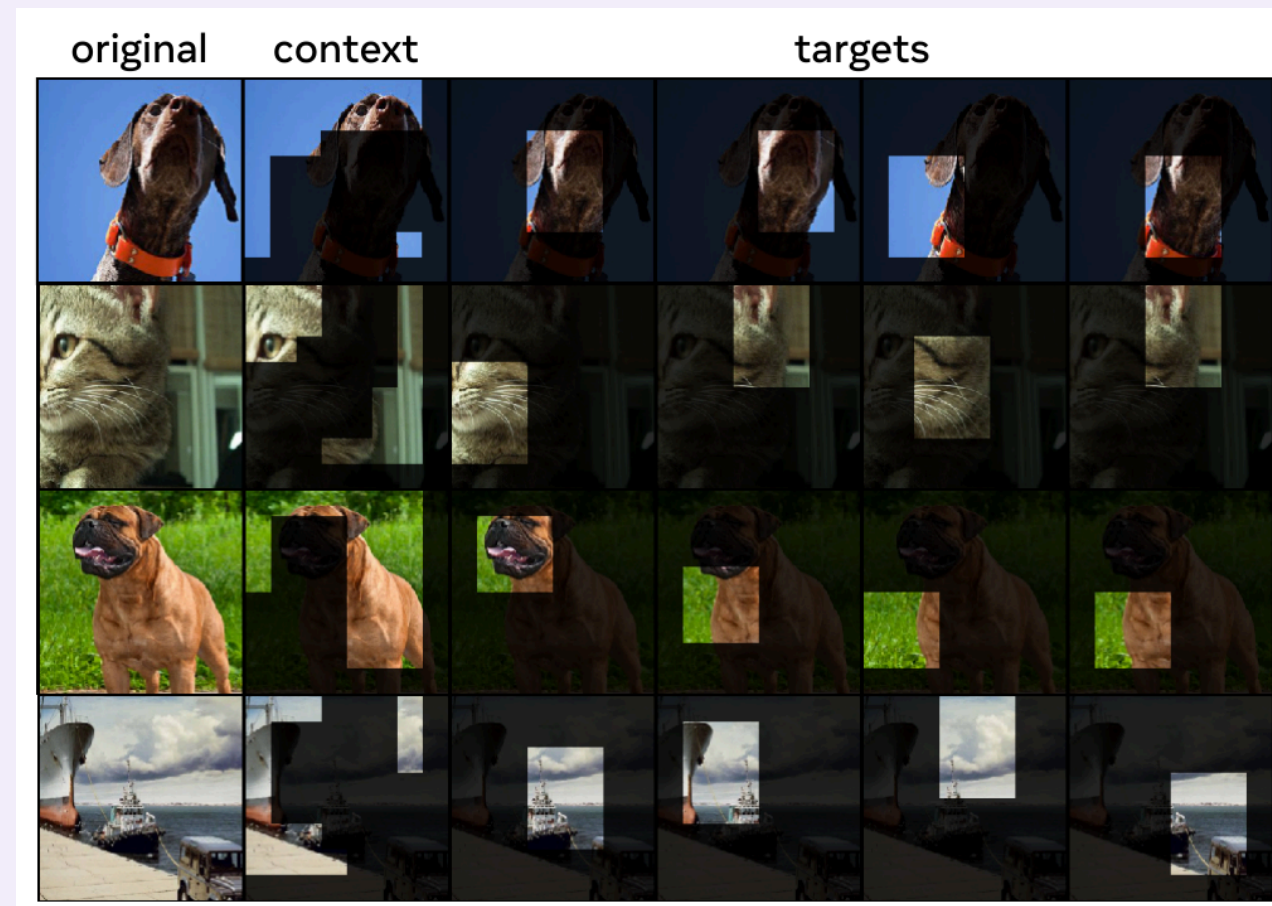
$$y_T = \text{Norm}([f_{\bar{\theta}}(x)]_T)$$

Encode the full image, then select T.

$$\mathcal{L} = \frac{1}{|T|} \sum_{k \in T} \left\| \underbrace{\hat{y}_k}_{\text{prediction (2)}} - \underbrace{\text{sg}(y_k)}_{\text{teacher (3)}} \right\|_2^2$$

Average feature error at the requested positions.
sg: the teacher receives no gradient from this loss.

I-JEPA: Mask Design



I-JEPA Figure 4 · actual context and target masks

- A tiny scattered hole may be filled with local texture clues.
- A larger coherent target can require information about the surrounding object.
- Enough visible context must remain to make prediction informative.

Difficulty alone is not the goal: the missing content must be inferable from useful context.

I-JEPA: Masking Ablation

Mask	Targets		Context		Avg. Ratio*	Top-1
	Type	Freq.	Type			
multi-block	Block(0.15, 0.2)	4	Block(0.85, 1.0) × Complement		0.25	54.2
rasterized	Quadrant	3	Complement		0.25	15.5
block	Block(0.6)	1	Complement		0.4	20.2
random	Random(0.6)	1	Complement		0.4	17.6

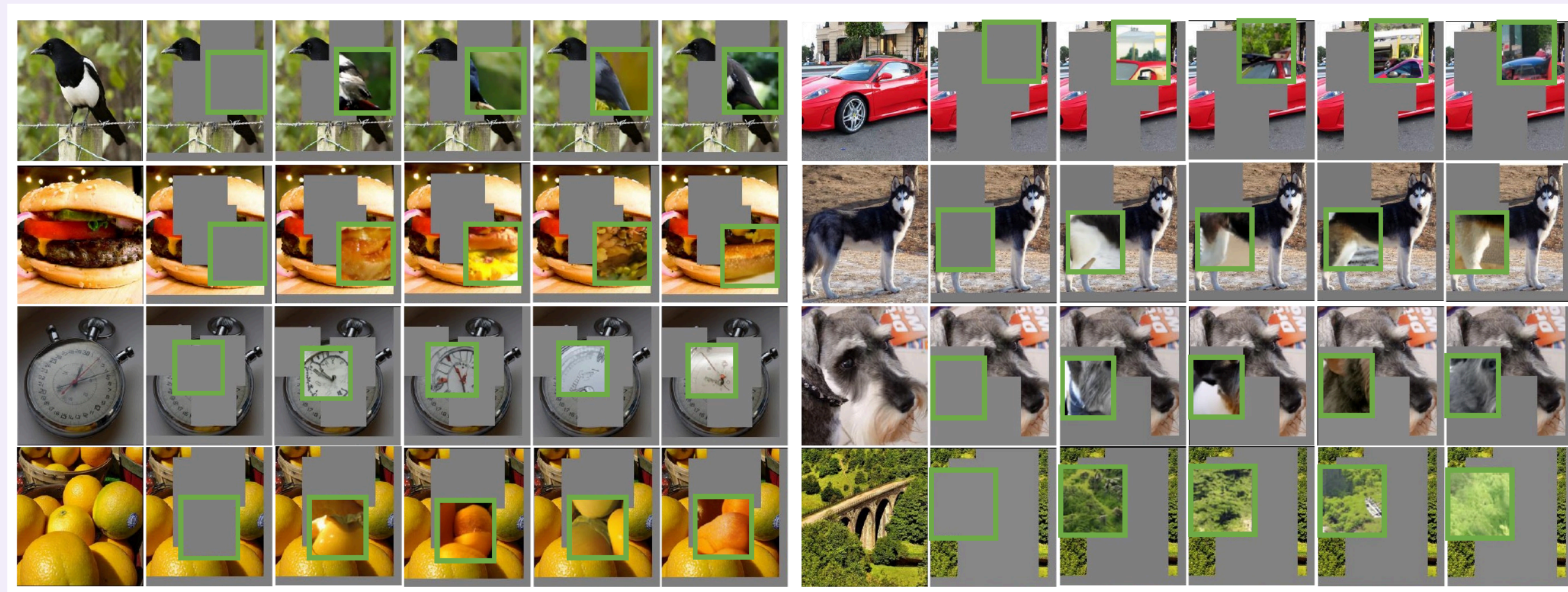
*Avg. Ratio is the average number of patches in the context block relative to the total number of patches in the image.

I-JEPA Table 6 · preserve the target and context settings when comparing rows

The paper compares mask configurations through downstream recognition performance.

Use the table to assess a masking recipe, not to infer that one geometric change explains everything.

I-JEPA: Feature Visualization



I-JEPA Figure 6 · feature visualizations produced by an additional generative decoder

These pixels come from a separately trained decoder; I-JEPA itself predicts features.

Temporal Information in Representations

Same final image. Different motion directions.

Clip A
moving right →

Clip B
← moving left

Earlier

Later

Now · identical images

One frame

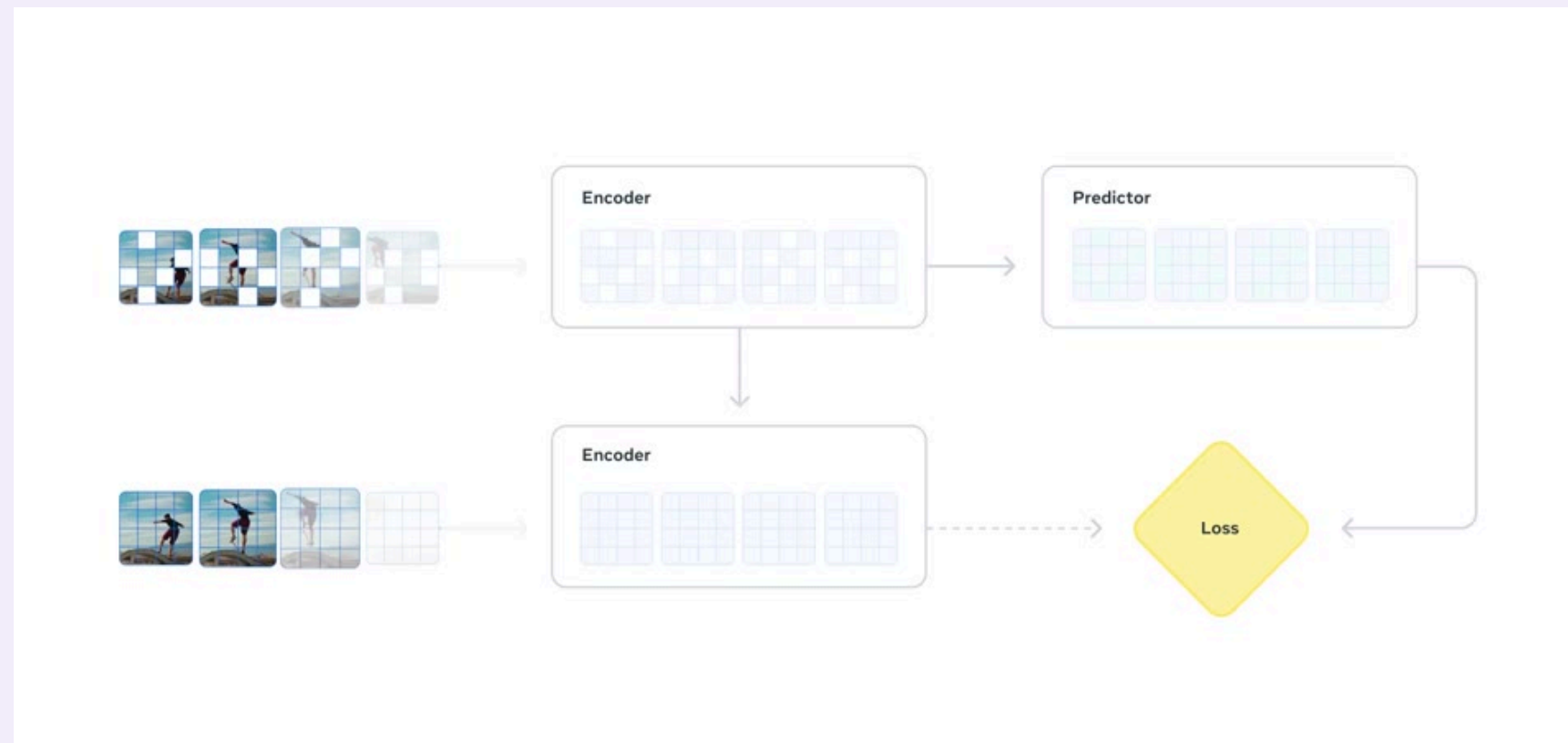
The puck is at the center.
Its previous direction is ambiguous.

Ordered frames

A: left → center. B: right → center.
The sequence reveals the direction.

Goal for video representations: retain motion direction from the ordered frames.

V-JEPA



Student sees a masked clip; teacher sees the full clip.

$$\mathcal{L} = \frac{1}{|T|} \sum_{k \in T} \|\hat{y}_k - \text{sg}(y_k)\|_1$$

Original V-JEPA uses an L1 feature distance. The target contains spatiotemporal context.

V-JEPA 2 · original self-supervised video illustration

Masking video creates a feature-completion task; it does not automatically create a causal forecaster.

Masked Completion vs. Causal Forecasting

Task	What the predictor can observe	What it predicts
Masked completion	Visible regions before and after a hole	Features at missing locations
Causal forecasting	History up to time t , plus supplied actions	Features after time t

A clip can contain future context even when some frames or regions are hidden.

Inspect the attention mask and conditioning, not just the word “predict.”

V-JEPA 2.1: Global and Dense Features

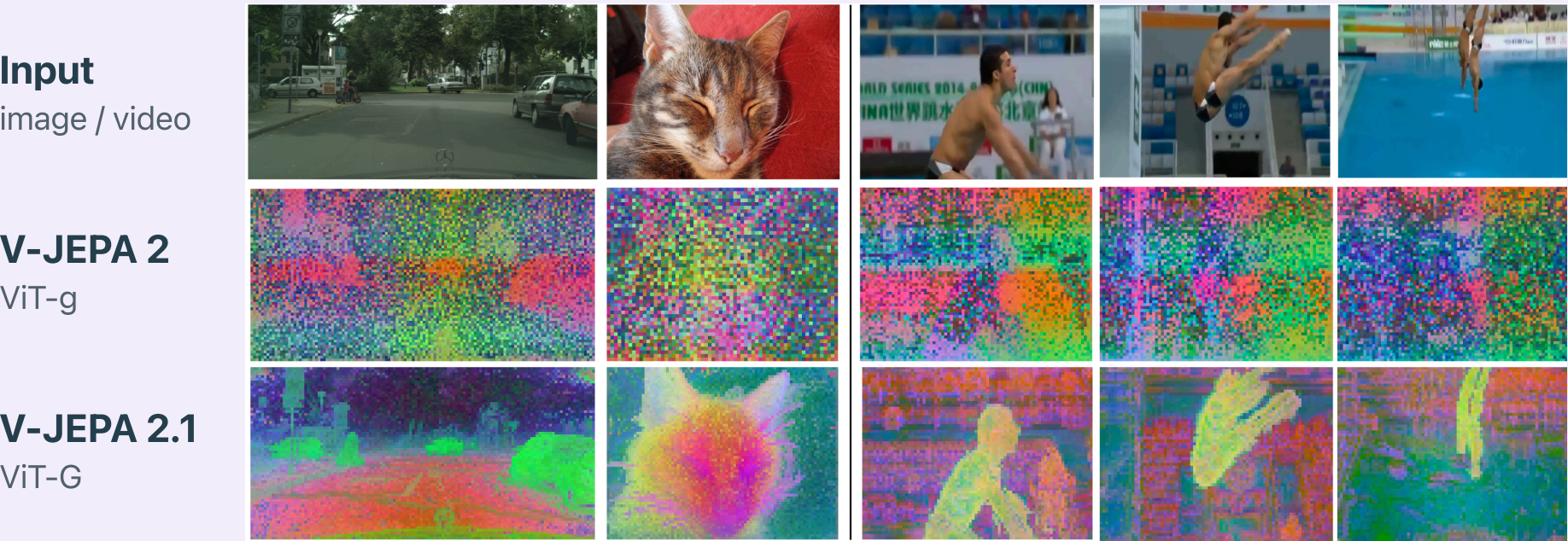


Figure 1 · patch-feature PCA comparison

Color key: patch feature → 3 PCA components → RGB.
Colors are feature coordinates, not class labels.

SSv2 measures action classification. ADE20K measures image segmentation. Neither is a control-success score.

The target locations and layers affect which information is easy to read out.

ViT-L ablation	SSv2 accuracy ↑	ADE20K mIoU ↑
V-JEPA 2 baseline	72.8	22.2
+ context loss	62.5	33.8
+ multi-level prediction	72.1	38.6

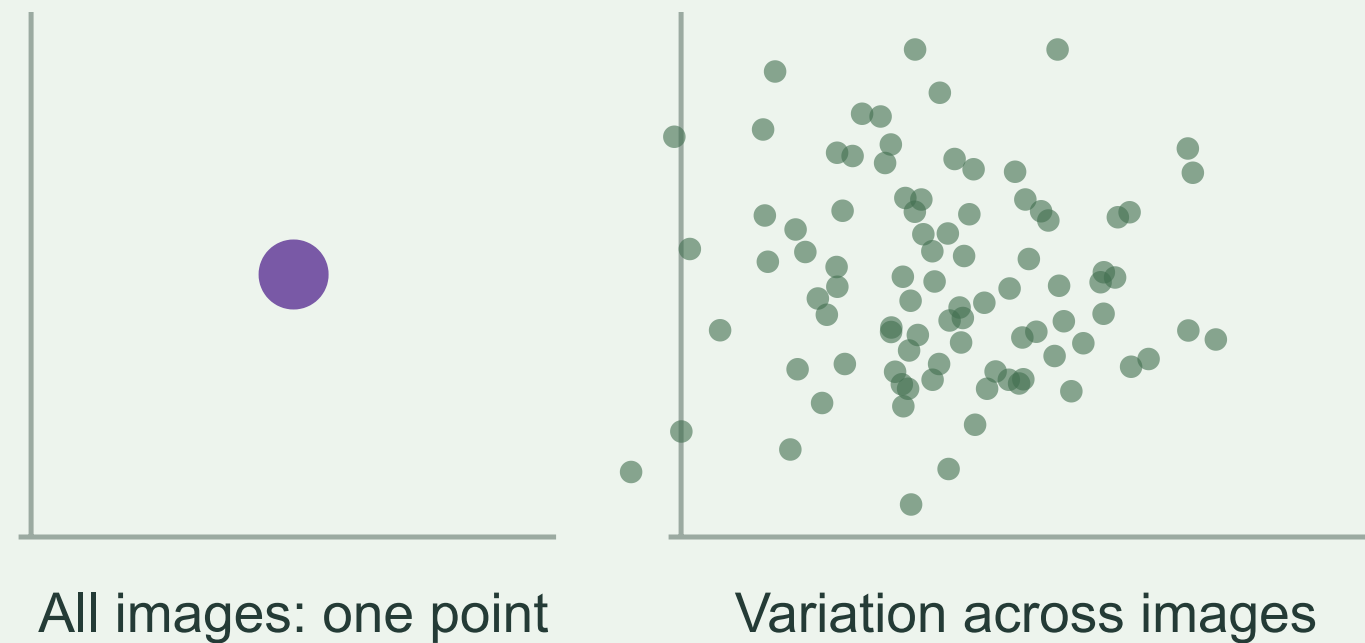
Collapse Control: Methods So Far

Models	Main ingredients	Role in the training recipe
MAE / AIM	Targets from the original data	A useful prediction must depend on the input.
SimCLR / CLIP / SigLIP	Positive and negative comparisons	Identical outputs cannot separate matching from non-matching pairs.
DINO	EMA + stop-gradient Centering + sharpening	Slow targets; balanced output usage and confident targets.
BYOL	EMA + stop-gradient Online-only predictor	Asymmetric branches and updates for feature agreement.
data2vec	EMA + stop-gradient Normalized teacher features	Stabilized, scale-controlled regression targets.
I-JEPA / V-JEPA	EMA + stop-gradient Context-to-target predictor	Asymmetric feature prediction, with recipe-specific normalization.

EMA: slow teacher updates. **Stop-gradient:** hold the target fixed during the current backward pass.

Learned-target methods depend on the complete training recipe. No listed ingredient is a universal guarantee.

Explicit Constraints on Feature Variation



Each dot represents an image embedding; its views can agree in either picture.
Schematic distributions.

Agreement alone

Every image can share one representation.

An explicit constraint

Penalize a batch of features with no variation.

$$\mathcal{L} = \mathcal{L}_{\text{agreement}} + \lambda R(Z)$$

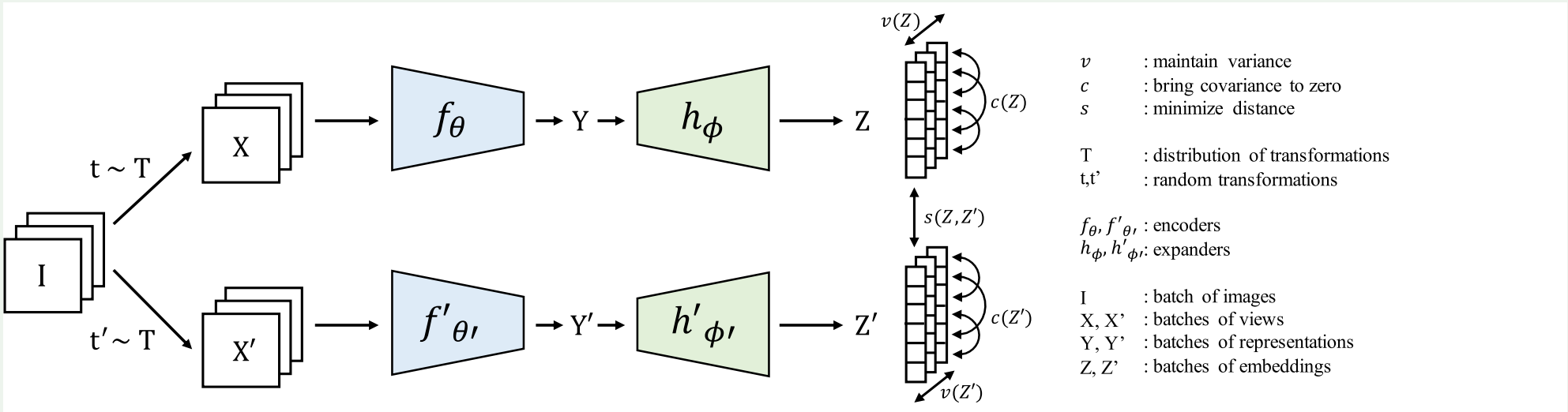
Z: embeddings from different images.

A constant Z must pay a nonzero regularization cost.

The next question is what to constrain—and which collapsed solutions the constraint rules out.

VICReg

Variance–Invariance–Covariance Regularization



VICReg · original Figure 1 · encoder f , expander h , training embeddings Z and Z'

V · Variance

Across images:
each feature coordinate should vary.

I · Invariance

Within one image:
its two views should agree.

C · Covariance

Across coordinates:
discourage redundant variation.

Related reading: [Barlow Twins \(2021\)](#). Cross-correlation between views $\rightarrow$ identity matrix.

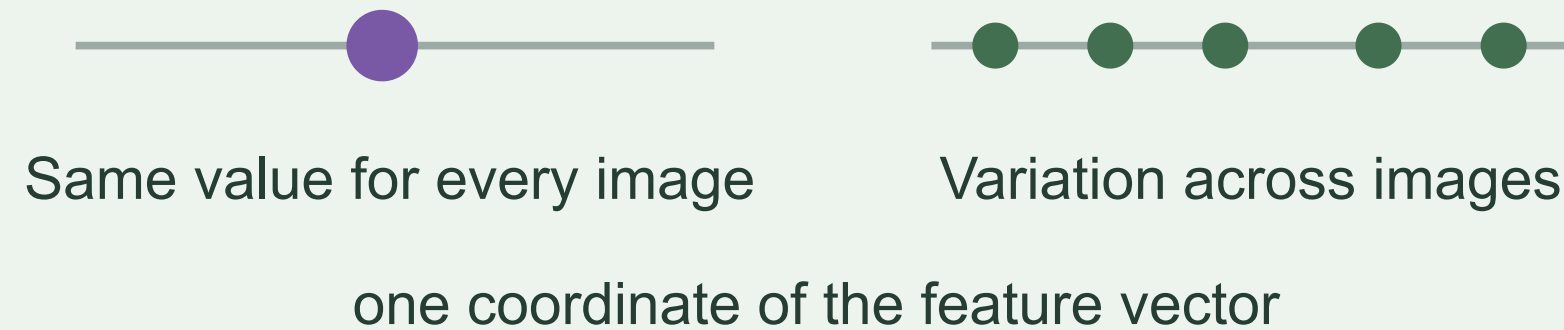
VCReg: the V + C regularization strategy. VICReg also includes I, the view-agreement term.

VCReg: Variance and Covariance

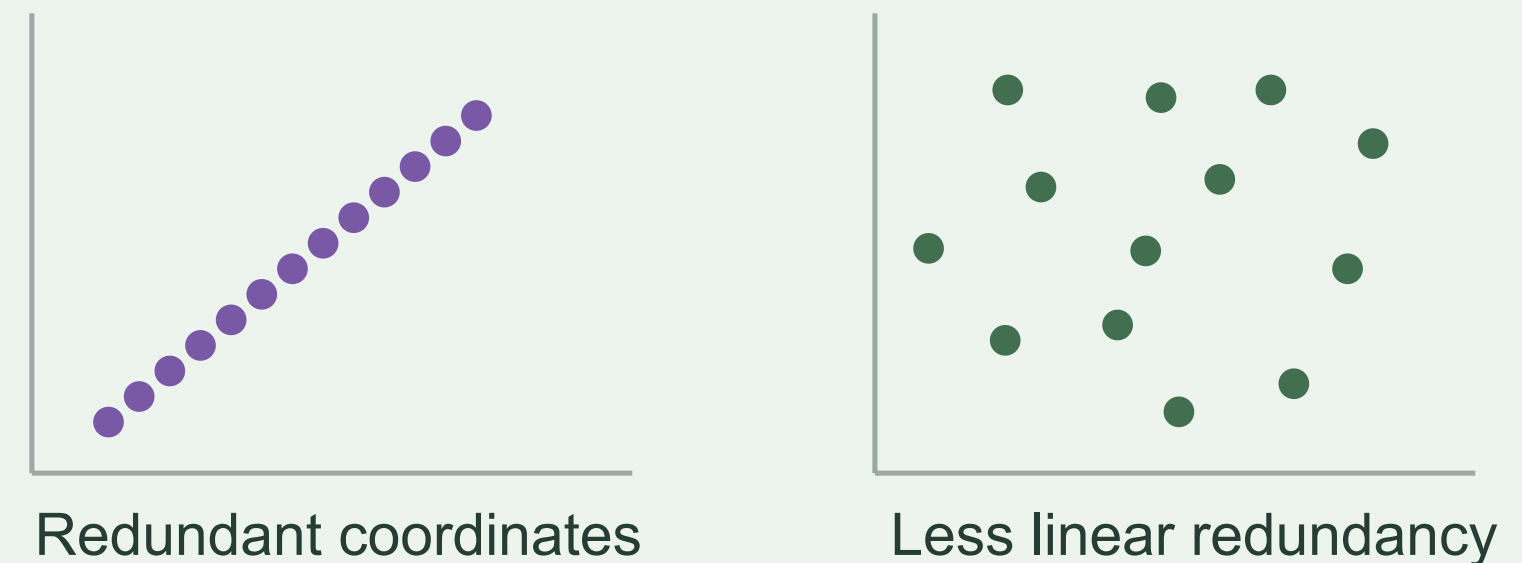
VCReg = Variance-Covariance Regularization. Here: the V + C terms used in VICReg.

V: a floor on coordinate variation

C: a penalty on coordinate redundancy



$$R_V = \frac{1}{d} \sum_j \max(0, \gamma - \sqrt{\text{Var}(z_j) + \epsilon})$$

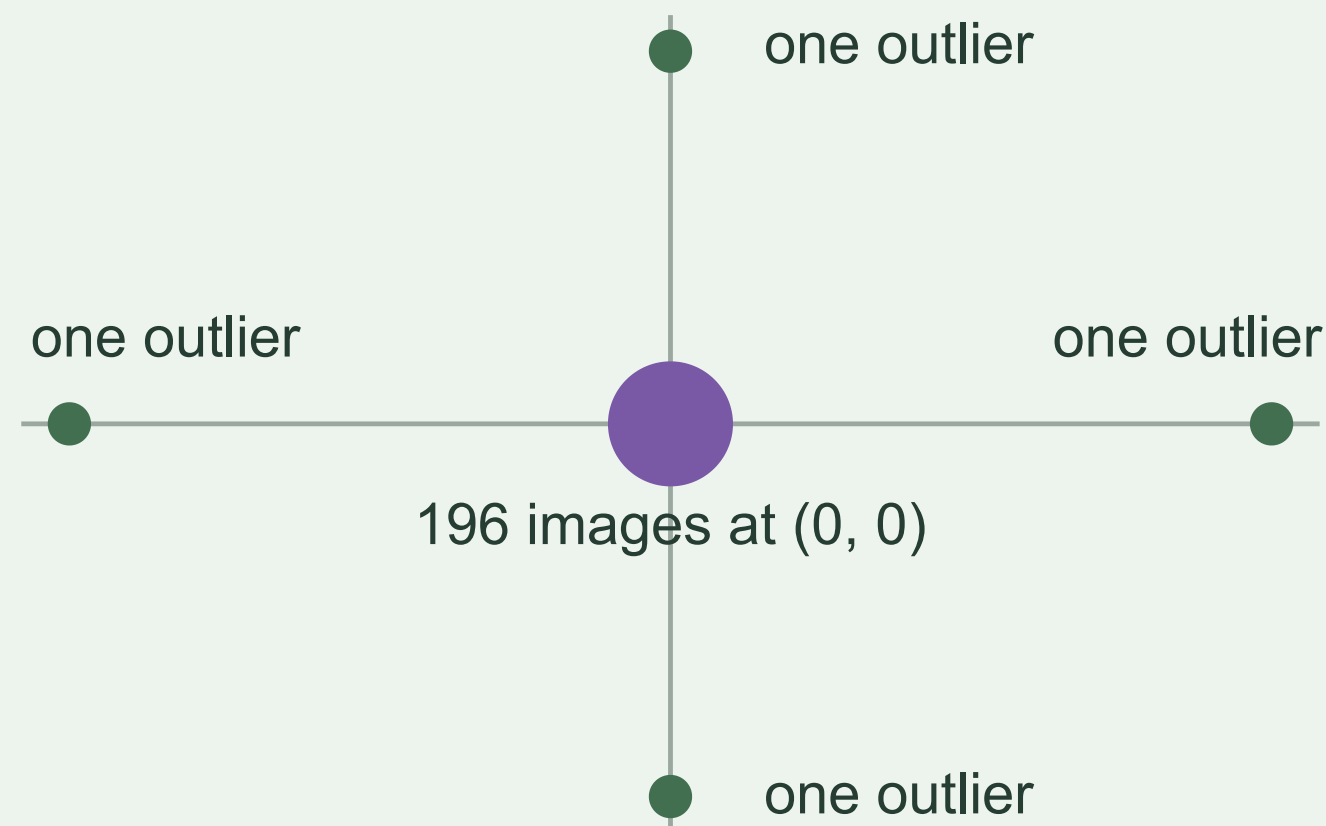


$$R_C = \frac{1}{d} \sum_{i \neq j} \text{Cov}(z)_{ij}^2$$

Statistics are computed across images. d: feature dimension; γ : minimum standard deviation; ϵ : numerical stabilizer.

V + C constrain second-order statistics. They do not specify the full distribution.

VCReg: What Moments Miss



A constructed 2D example · dot size indicates overlap.

Passing variance and covariance checks is not enough. We also need to inspect distribution shape.

200 image embeddings

196 are identical; four are far away.

$$z_{\text{outlier}} \in \{(\pm a, 0), (0, \pm a)\}, \quad a = \sqrt{199/2}$$

The four outliers supply all the variance.

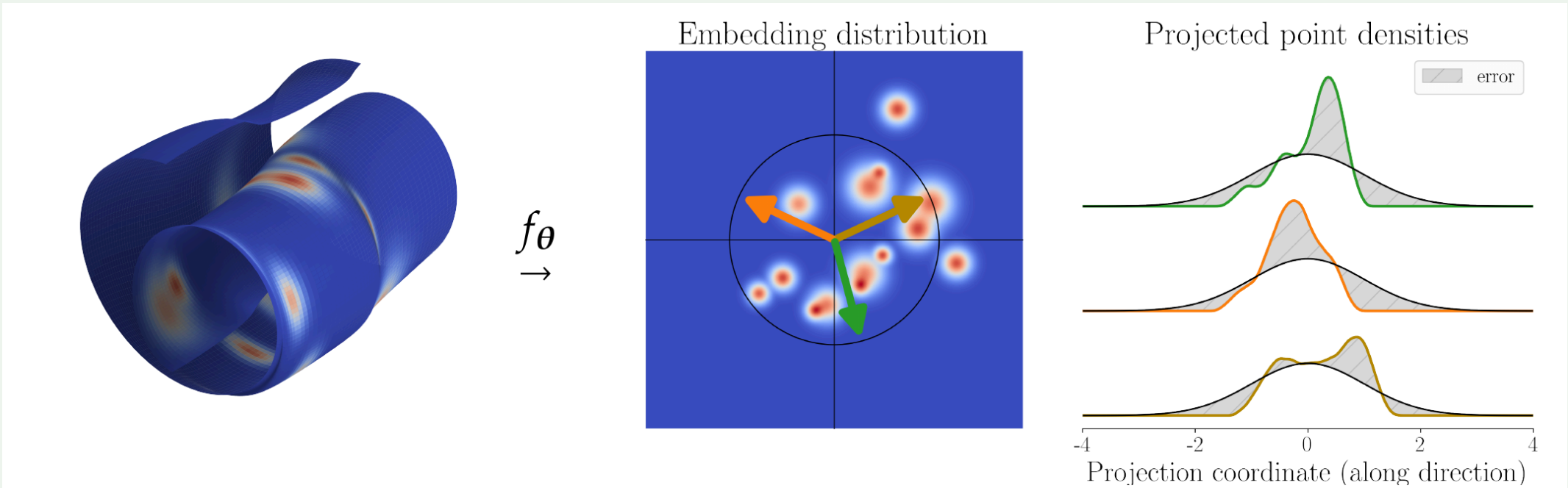
$$\text{Cov}_{\text{sample}}(Z) = I_2 \quad \Rightarrow \quad R_V = R_C = 0$$

Variance floor $\gamma = 1$; usual small ϵ .

98% of the images have the same features.

SIGReg

Sketched Isotropic Gaussian Regularization



LeJEPA Figure 2 · project embeddings and compare with a Gaussian reference

Sketched

Use sampled one-dimensional projections.

Isotropic Gaussian

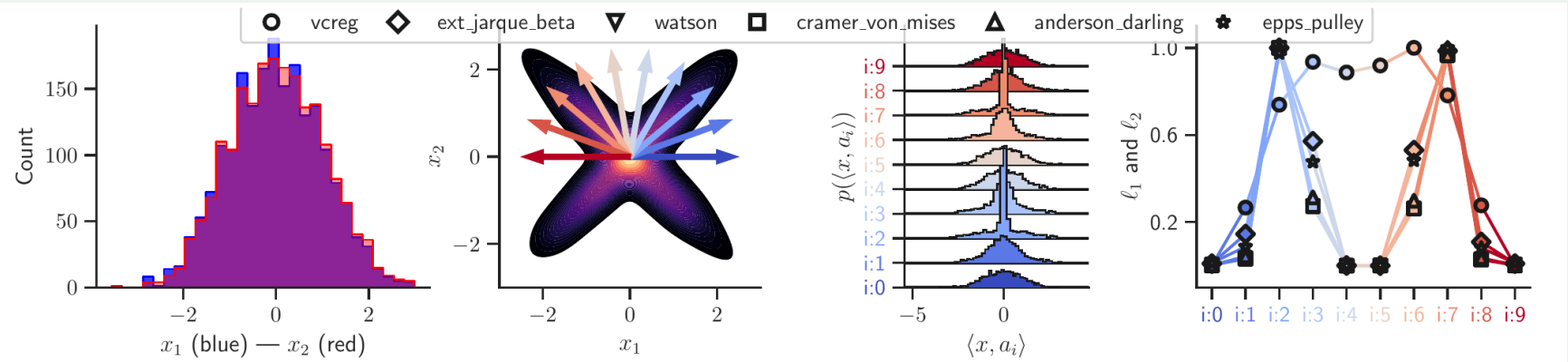
A bell-shaped reference with the same spread in every direction.

Regularization

Add a loss for mismatch with that reference.

A difficult high-dimensional comparison becomes many simple one-dimensional comparisons.

Gaussian Marginals and Joint Distributions



LeJEPA Figure 5 · X-shaped distribution

Even Gaussian coordinate marginals plus identity covariance do not determine the joint distribution.

$$Z = (X, SX), \quad X \sim \mathcal{N}(0, 1), \quad S \in \{-1, +1\}$$

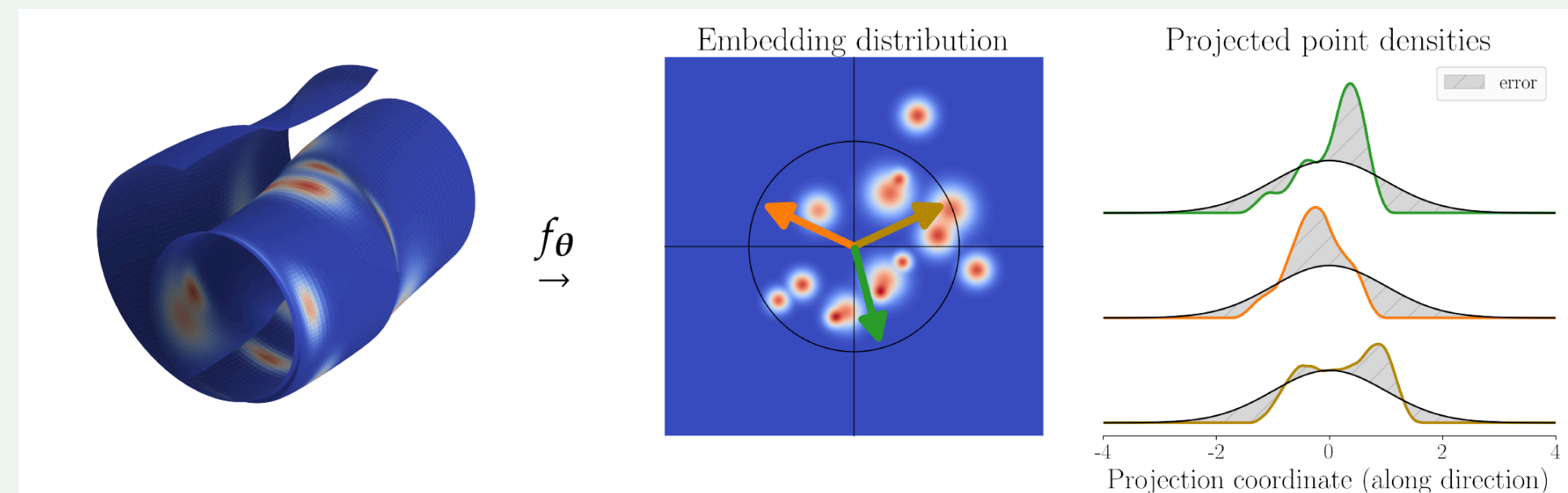
Let S be an independent fair sign. Each coordinate is normal; covariance is I.

$$\frac{Z_1 + Z_2}{\sqrt{2}} = \begin{cases} \sqrt{2}X & S = +1 \\ 0 & S = -1 \end{cases}$$

A diagonal projection exposes a point mass at zero.

Random Projections and the Cramér–Wold Theorem

$$u_{n,m} = a_m^\top z_n, \quad \|a_m\| = 1$$



A random direction selects a scalar statistic of every embedding.

Cramér–Wold: equality of every one-dimensional projection distribution implies equality of the joint distribution.

Finite random directions and finite frequencies give an approximate constraint, not an exact certificate.

Characteristic Functions

$$\hat{\varphi}_m(t) = \frac{1}{N} \sum_n e^{itu_{n,m}} = C_m(t) + iS_m(t)$$

$$C_m(t) = \frac{1}{N} \sum_n \cos(tu_{n,m}), \quad S_m(t) = \frac{1}{N} \sum_n \sin(tu_{n,m})$$

$$\varphi_{\mathcal{N}(0,1)}(t) = e^{-t^2/2}$$

For each frequency t , compute two ordinary averages. Together they describe the projected distribution.

The calculation uses all sample values through oscillations, not only a mean and variance.

SIGReg: Objective

$$|\hat{\varphi}_m(t) - e^{-t^2/2}|^2 = (C_m(t) - e^{-t^2/2})^2 + S_m(t)^2$$

$$R_{\text{SIGReg}} = \frac{N}{M} \sum_{m=1}^M \int \left[(C_m(t) - e^{-t^2/2})^2 + S_m(t)^2 \right] w(t) dt$$

In code: project → average cosine/sine → subtract the reference → square → integrate.

$$z_n = c \ \forall n \quad \Rightarrow \quad |\hat{\varphi}_m(t)| = 1 > e^{-t^2/2} \quad (t \neq 0)$$

Identical embeddings cannot match the Gaussian reference: collapse has an explicit regularization cost.

LeJEPA

Le = Latent-Euclidean · JEPA = Joint-Embedding Predictive Architecture



View 1



View 2

Within one image: bring its view embeddings together.

Across different images: keep a specified distribution of embeddings.

The minimal recipe learns both sides directly, without an EMA teacher.

VICReg: view agreement + VC penalties. LeJEPA: view agreement + SIGReg.

LeJEPA: View Matching and Regularization

$$\mu_n = \frac{1}{V} \sum_v z_{n,v}, \quad \mathcal{L}_{\text{match}} = \frac{1}{BVd} \sum_{n,v} \|z_{n,v} - \mu_n\|^2$$

Operation	Which embeddings are compared?	What a constant encoder does
View matching	Different views of one image n	Makes the loss zero
Distribution constraint	Different images n at each view index v	Fails to resemble a spread-out Gaussian

$$\mathcal{L} = (1 - \lambda) \mathcal{L}_{\text{match}} + \frac{\lambda}{V} \sum_v R_{\text{SIGReg}}(z_{1:B,v})$$

The two terms act along different axes of the training batch.

SIGReg: Distributed Computation

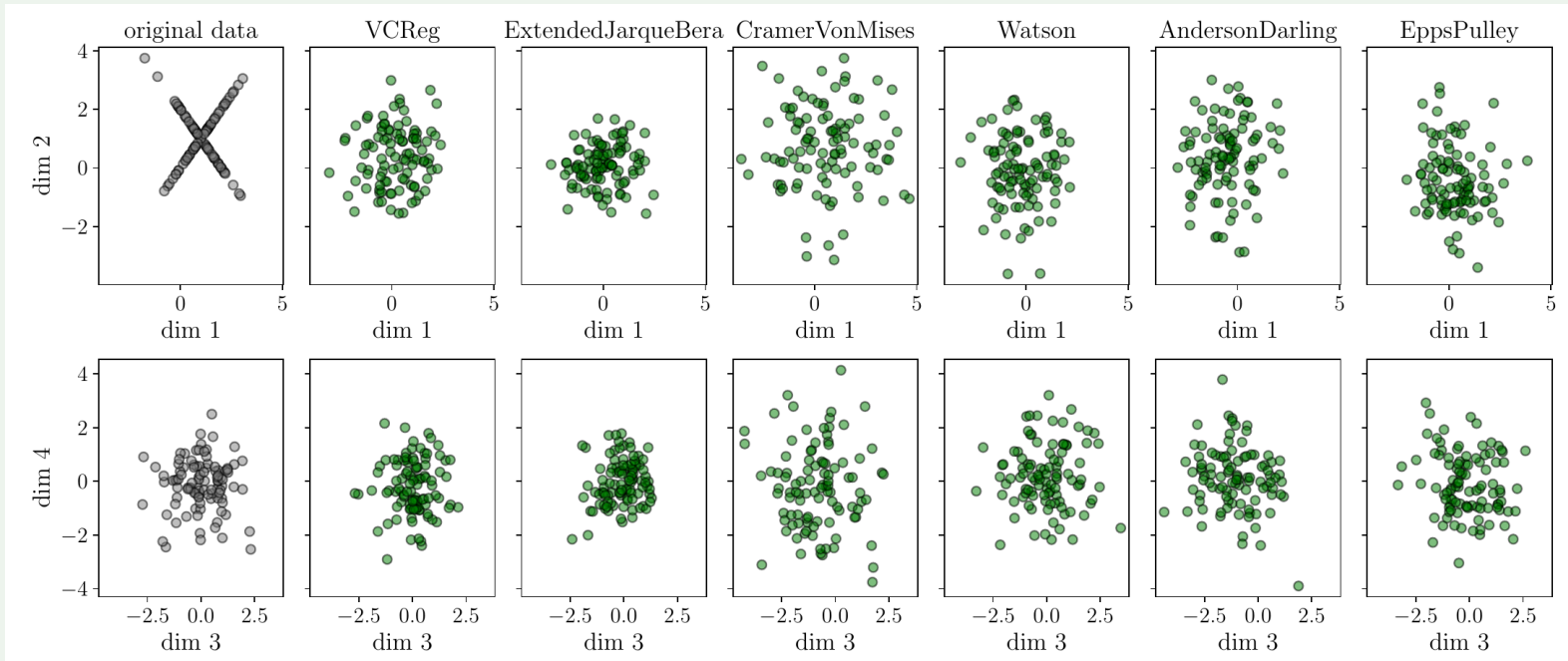
$$C_m(t) = \frac{\sum_r \sum_{n \in r} \cos(t a_m^\top z_n)}{\sum_r N_r}, \quad S_m(t) = \frac{\sum_r \sum_{n \in r} \sin(t a_m^\top z_n)}{\sum_r N_r}$$

```
A = shared_unit_directions()
u = z_local @ A
cos_sum = cos(u[..., None] * t).sum(0)
sin_sum = sin(u[..., None] * t).sum(0)
C = differentiable_all_reduce(cos_sum) / N
S = differentiable_all_reduce(sin_sum) / N
```

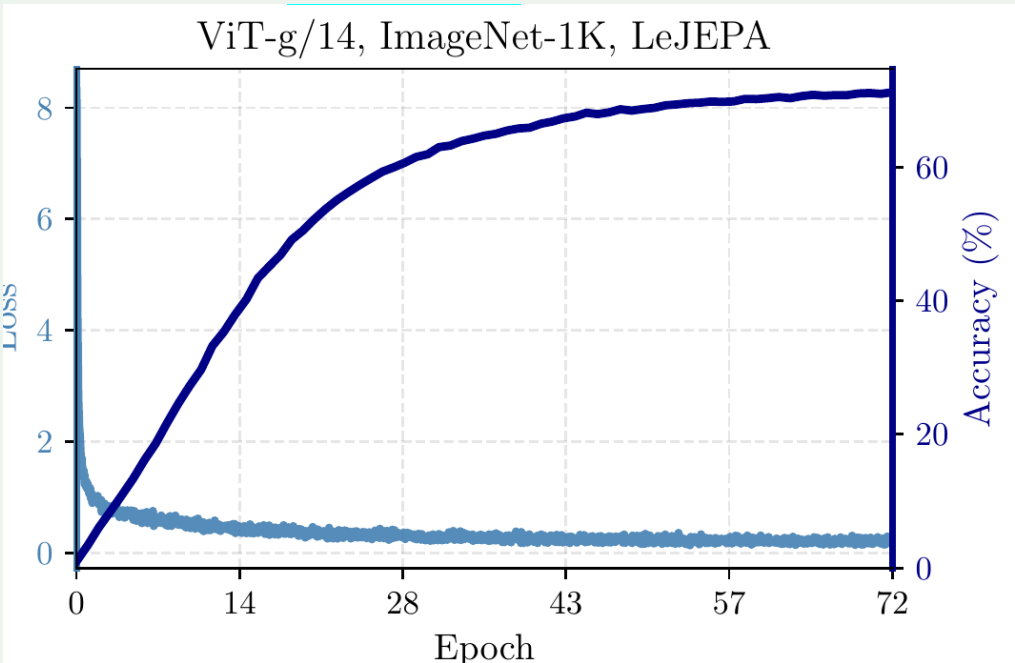
- Use the same directions and frequencies on every rank.
- Reduce sample counts, then normalize by the global count.
- Preserve gradients through the collective; keep DDP loss scaling consistent.

Communicate $2 \times \text{directions} \times \text{frequencies}$ statistics per view, rather than every sample embedding.

LeJEPa: Distribution and Training Experiments



LeJEPa Figure 6 · optimizing sample points under distribution penalties



LeJEPa Figure 1 · representation-training stability evidence

Left: a direct point-distribution experiment. Right: model-training behavior under the paper’s settings.

Better distribution control addresses collapse; task readouts still decide what the features are useful for.

Collapse Control: A Summary

Method	Collapse control	Remaining limitation
DINO / BYOL / predictive JEPA	Teacher updates and asymmetry; output shaping in DINO	The safeguards depend on the recipe.
VICReg	Feature variance + decorrelation, alongside view agreement	Second-order statistics can miss severe concentration.
SIGReg	Projected distributions matched to a Gaussian	Finite samples and sampled directions; a regularizer, not a complete method.
LeJEPA	View agreement + SIGReg	The chosen views still determine which information must be shared.

Batch dependence: statistical penalties depend on which examples are measured together.

Avoiding collapse does not establish that the representation preserves useful information.

Regularization of Dense Video Features

$$Z \in \mathbb{R}^{B \times T \times H \times W \times D}, \quad M = THWD$$

B clips · T time steps · H × W token grid · D features per token

Where is the constraint applied?	Cost / scope
Whole clip: covariance One M-dimensional vector per clip	M × M matrix; direct computation O(BM²) Spatial and temporal coordinates included
Each token: variance / covariance D-dimensional vectors	D × D statistics No direct constraint on the full joint distribution across positions
Whole clip: SIGReg K random projections of M-dimensional vectors	Projection cost O(BMK); no M × M matrix Finite samples and directions still limit what is checked
Illustrative clip: 8 × 14 × 14 × 512 → M = 802,816. A full covariance matrix alone needs about 2.6 TB in float32.	

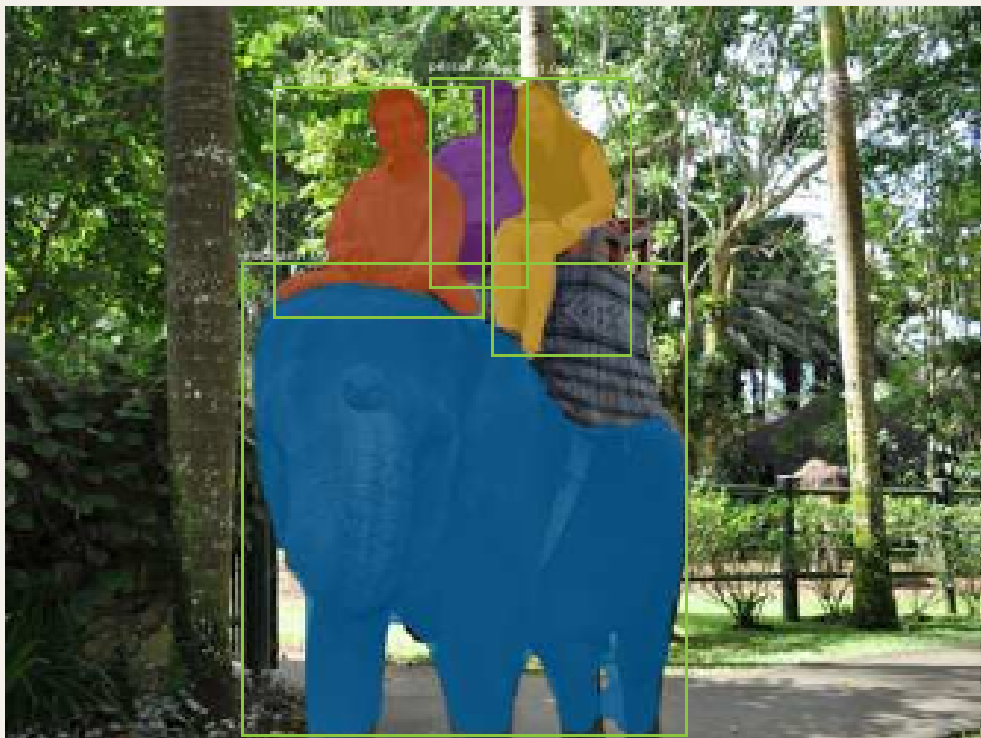
Well-distributed token features can still be identical across space and time within each clip.

Three Families of Representation Learning

Training family	What counts as the answer?	Main opportunity	What can go wrong?
Reconstruction	Original data / missing pixels	Keep rich spatial detail; model conditional outcomes	Task-irrelevant error can dominate; point estimates average
Contrastive	The paired view among candidates	Emphasize information shared across chosen views	False negatives; augmentation removes useful differences
Learned targets	Another encoding of a view or hidden content	Choose a more abstract prediction space	Collapse; target can erase useful distinctions

Choose the target, visible evidence, and collapse control together. No family guarantees a world-model-ready representation.

Downstream Evaluation



Examples of different visual readouts

Needed information	A concrete test
Object category	Frozen global linear probe
Object location / contact	Frozen local features + small spatial head
Motion direction	Frozen clip features + direction classifier
Future consequences	Later: action-conditioned prediction and control

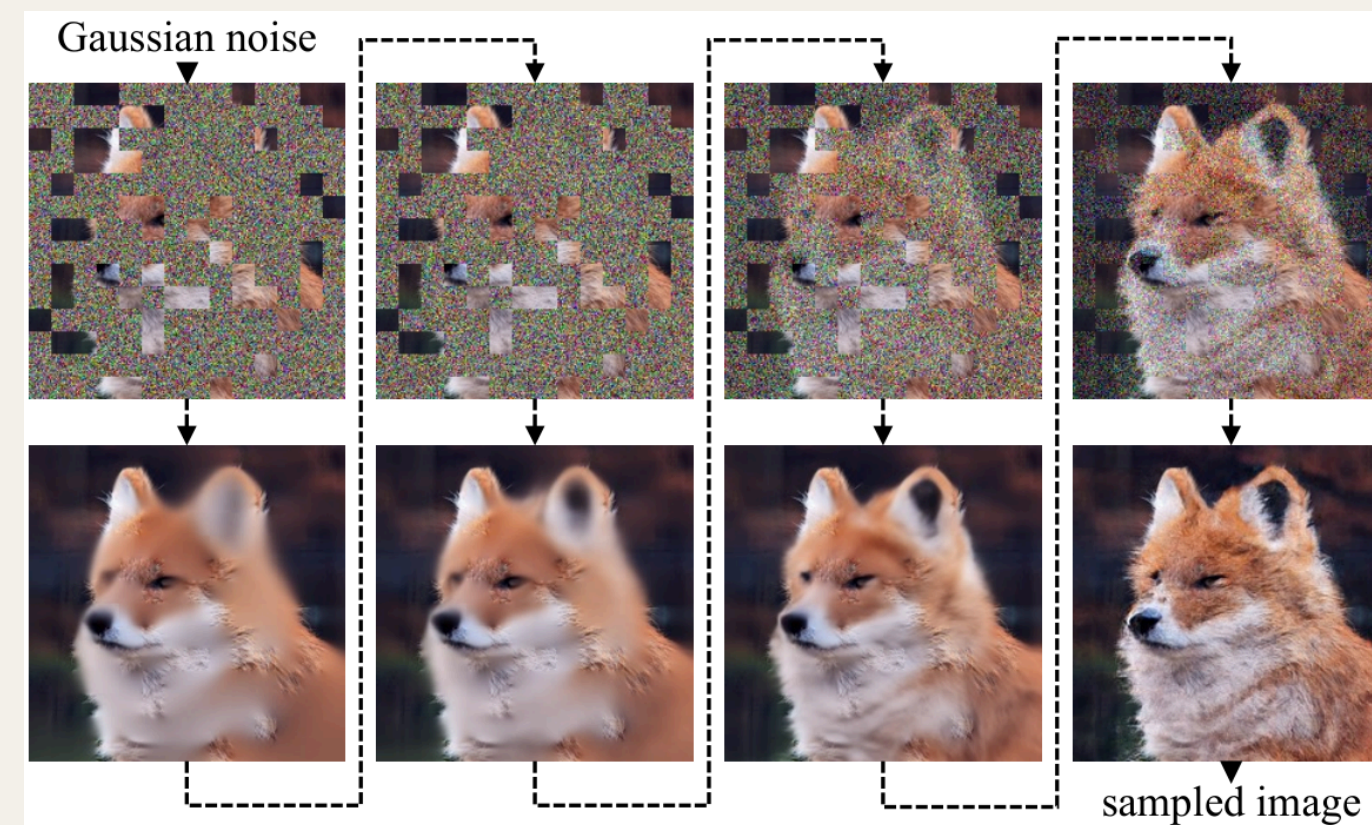
Specify the input and readout before saying “these features understand the world.”

JEPA Variants

Example	What is predicted or matched?	Additional constraint / objective
I-JEPA / V-JEPA	Hidden contextual image/video features	EMA target + recipe-specific masking and normalization
LeJEPA	Embeddings of views of the same image	SIGReg distribution constraint
VL-JEPA	Text embeddings conditioned on visual input	Bidirectional InfoNCE uses negative comparisons
LLM-JEPA	Latent predictions alongside language learning	Token cross-entropy remains in the training objective

To assess a new variant, read its target, conditioning, and complete objective—not only its name.

Generative Reconstruction and Feature Prediction



DiffMAE · generative reconstruction revisited

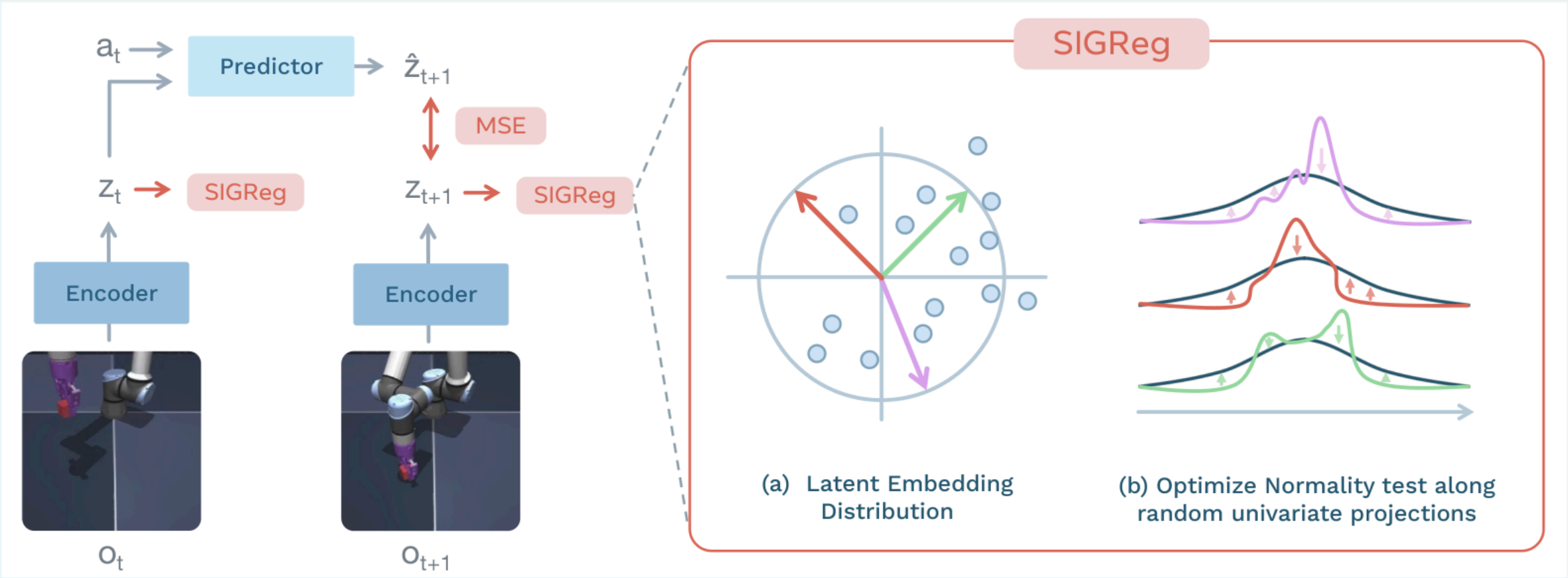
DiffMAE: denoising reconstruction can support representation learning.

D-JEPA: combine denoising / generative objectives with feature learning.

VarJEPA: represent predictive uncertainty; evidence depends on its tested domains.

A distribution can preserve alternatives. A learned feature target can reduce detail. These are compatible design choices.

LeWorldModel

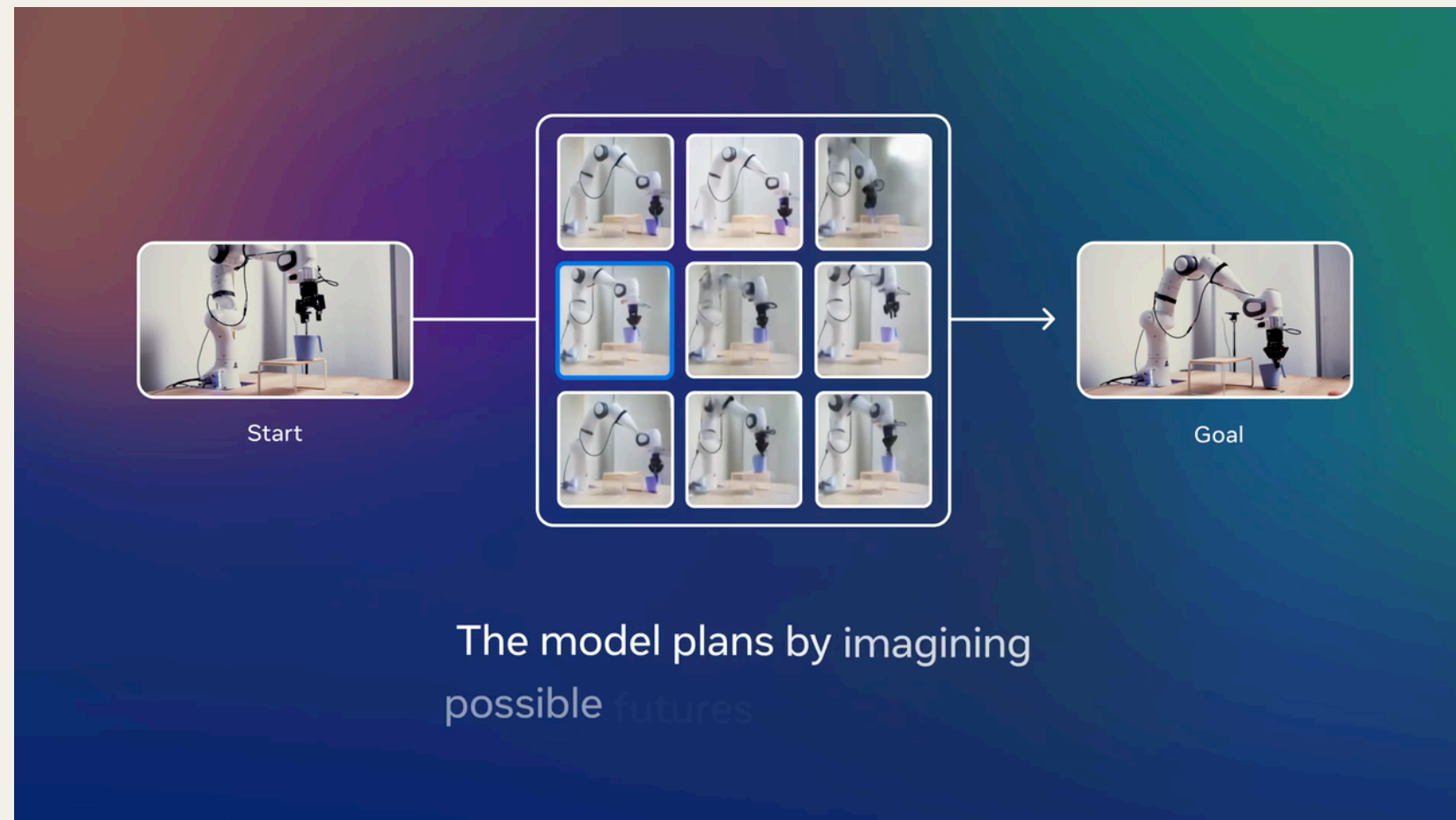


LeWorldModel Figure 1 · original training pipeline

$$\hat{z}_{t+1} = p_{\phi}(z_{\leq t}, a_{\leq t}), \quad \mathcal{L} = \mathcal{L}_{\text{future feature}} + \lambda R_{\text{SIGReg}}$$

Same regularizer, a different task: actions and past observations predict future features.

V-JEPA 2: Robot Demonstration



A representation benchmark measures accessible information.

A robot result also depends on action-conditioned training, prediction, planning and execution.

V-JEPA 2's control system includes additional action-conditioned training.

Official V-JEPA 2 robot demonstration · action-conditioned system.

Keep the evidence chain explicit: features → learned dynamics → the evaluated control system.

Next Lecture: Predictors and Generative Models

From **what information to represent** to **how to predict with it**.

LECTURES 5–6 · REPRESENTATIONS

The prediction space

What information should the encoder retain?

Which training targets produce useful features?

How can we prevent collapse and evaluate the result?

NEXT LECTURE · PREDICTION AND GENERATION

Predictor design

Architectures, conditioning, and prediction structure.

Generative model foundations

Conditional distributions, multiple possible outcomes, and sampling.

A different focus and a different test

Useful features for downstream tasks → accurate conditional predictions and plausible outcomes.

The representation sets the prediction space; the predictor models relationships and uncertainty within it.