Teacher-student feature prediction approaches

Spyros Gidaris & Andrei Bursuc

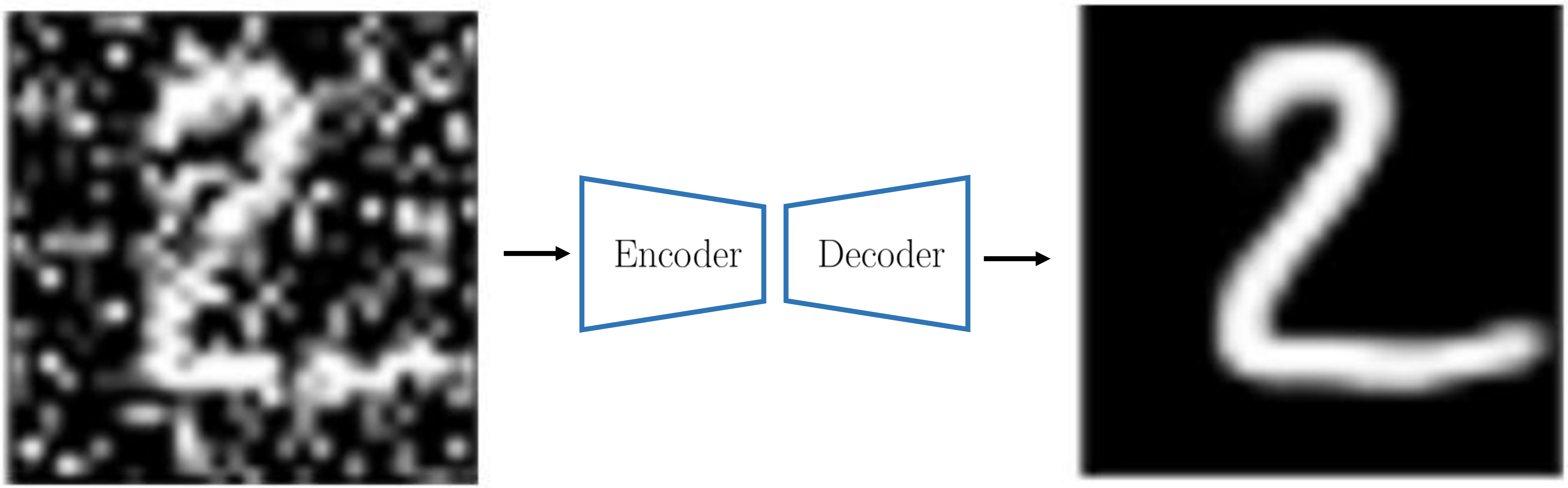
Agenda

- Input reconstruction
- Teacher-student feature reconstruction
- Wrap up evaluation

Agenda

- Input reconstruction
- Teacher-student feature reconstruction
- Wrap up evaluation

Input reconstruction for self-supervised representation learning



Perturb an image and then train a network to reconstruct the original version

- **Intuition:** to do that the network must recognize the visual concepts of the image
- One of the earliest methods for self-supervised representation learning

Denoising autoencoders (Vincent et al. 2008)

Denoising AutoEncoders

What is the noise and what the signal?

Denoising autoencoders (Vincent et al. 2008)

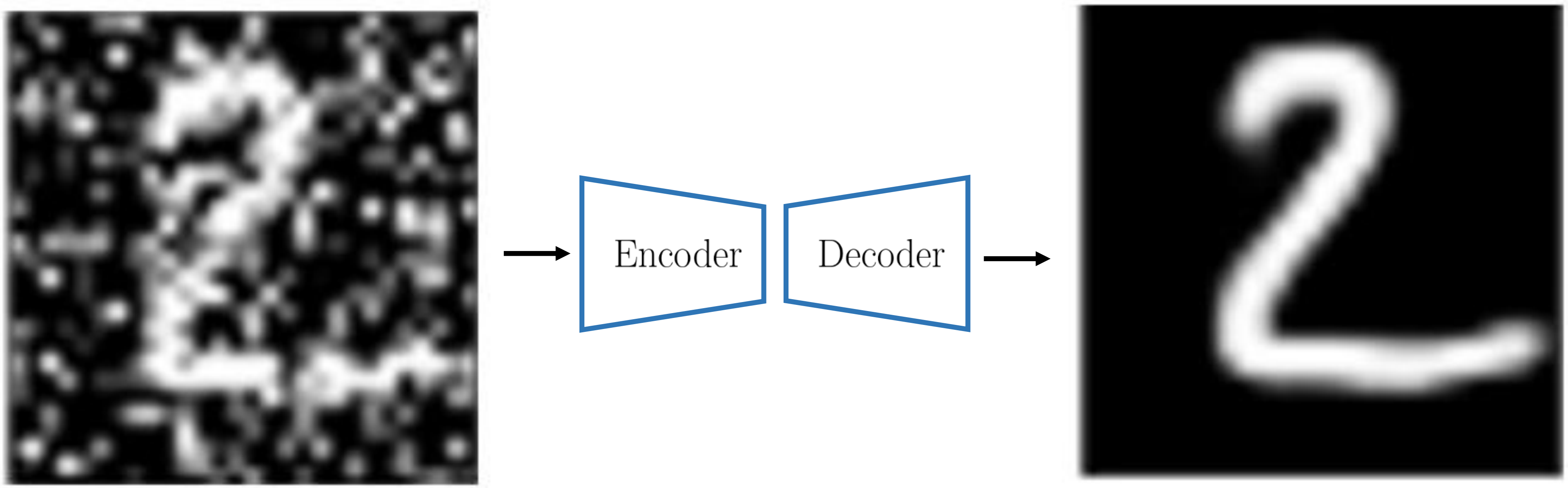
Denoising AutoEncoders

What is the noise and what the signal?

Recognizing the digit helps!

Denoising autoencoders (Vincent et al. 2008)

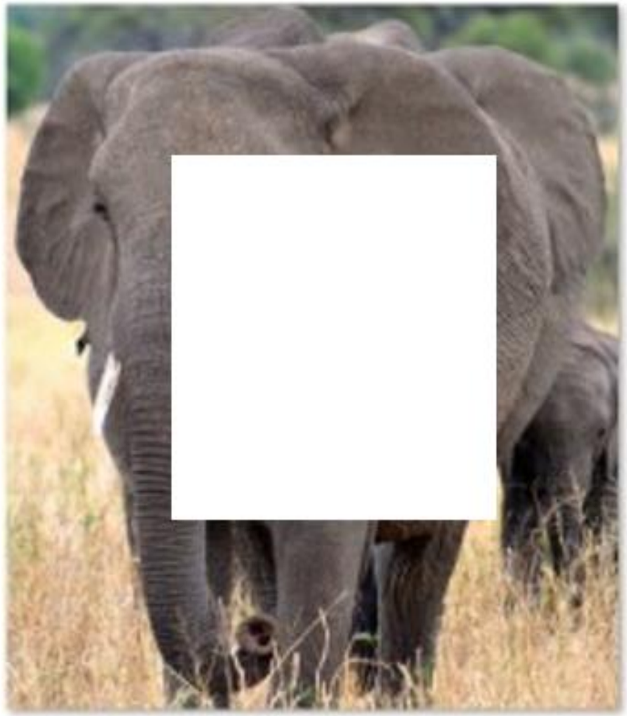
Denoising AutoEncoders



- Simple classical method
- Too easy, no need for semantics – low level cues are sufficient

Denoising autoencoders (Vincent et al. 2008)

Context Encoders

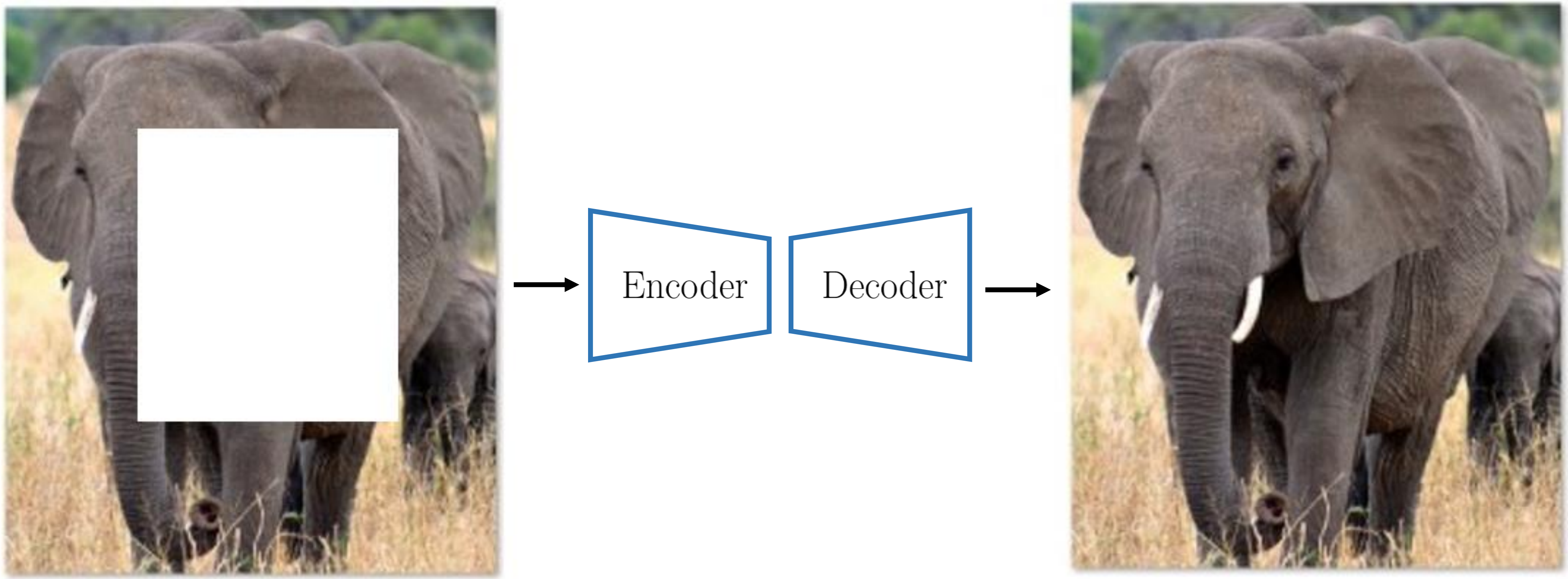


What goes in the middle?

Much easier if you recognize the objects!

Context Encoders (Pathak et al. 2016)

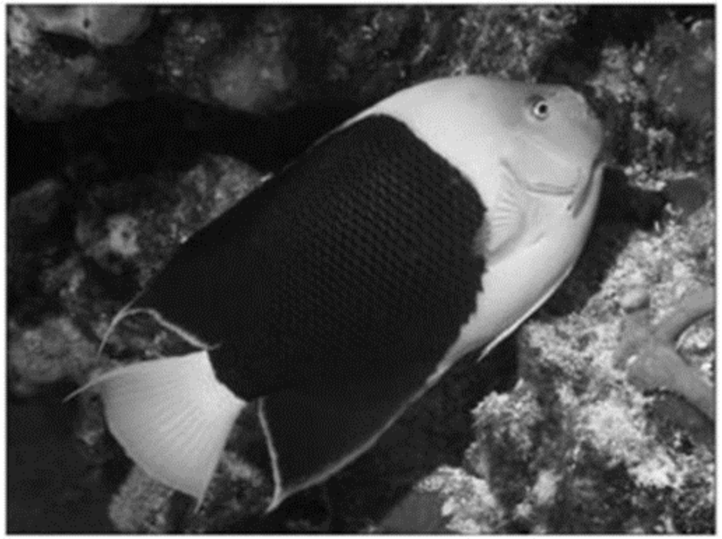
Context Encoders



- Requires preservation of fine-grained information and context-aware skills
- Input reconstruction is too hard and ambiguous
- Lots of effort spent on “useless” details: exact color, good boundary, etc.

Context Encoders (Pathak et al. 2016)

Colorization

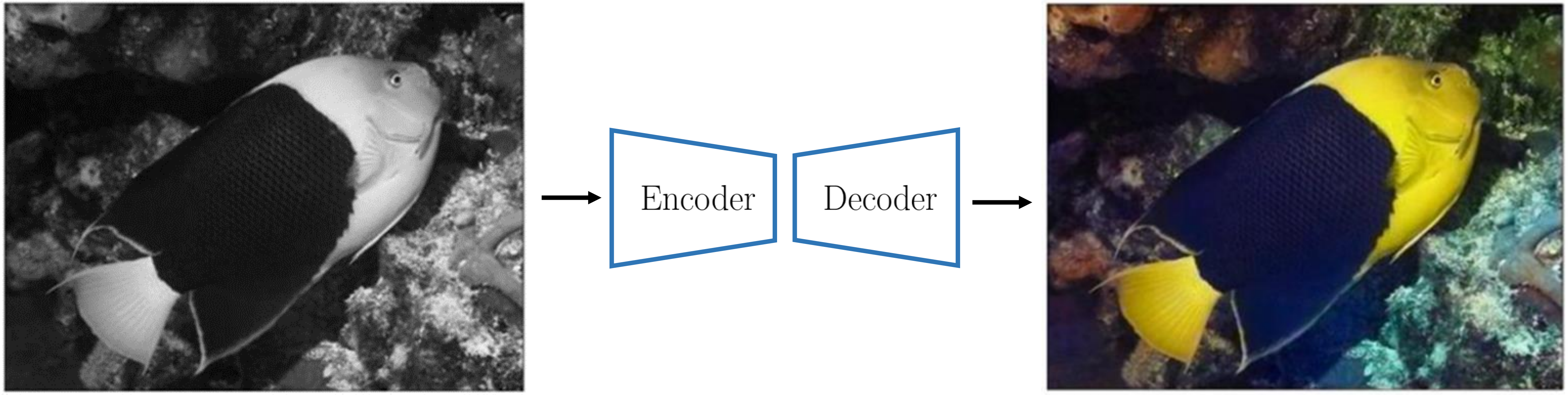


What is the colour of every pixel?

Hard if you don't recognize the objects!

Image colorization (Zhang et al. 2018)

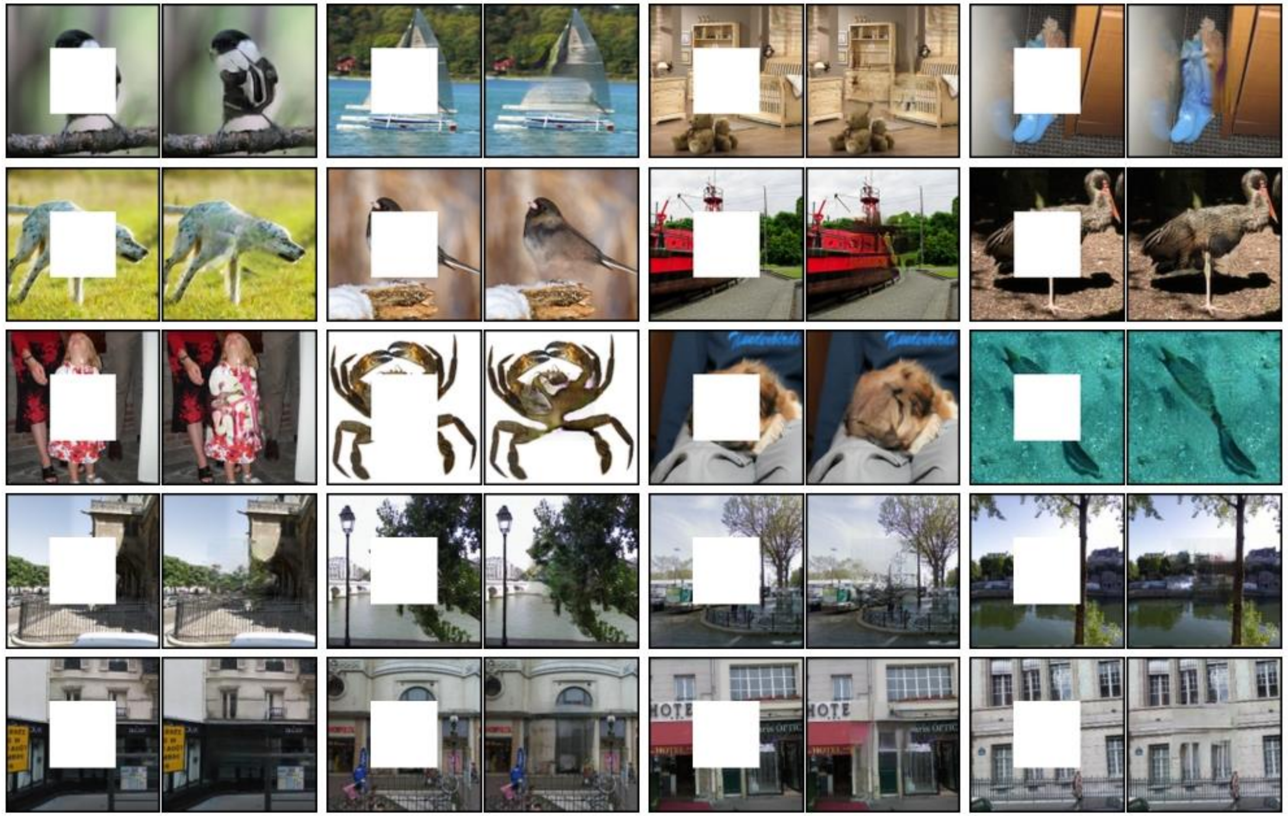
Colorization



- Requires preservation of fine-grained information
- Input reconstruction is too hard and ambiguous
- Lots of effort spent on “useless” details: exact color, good boundary, etc.

Image colorization (Zhang et al. 2018)

Recap: main limitations of input reconstruction methods

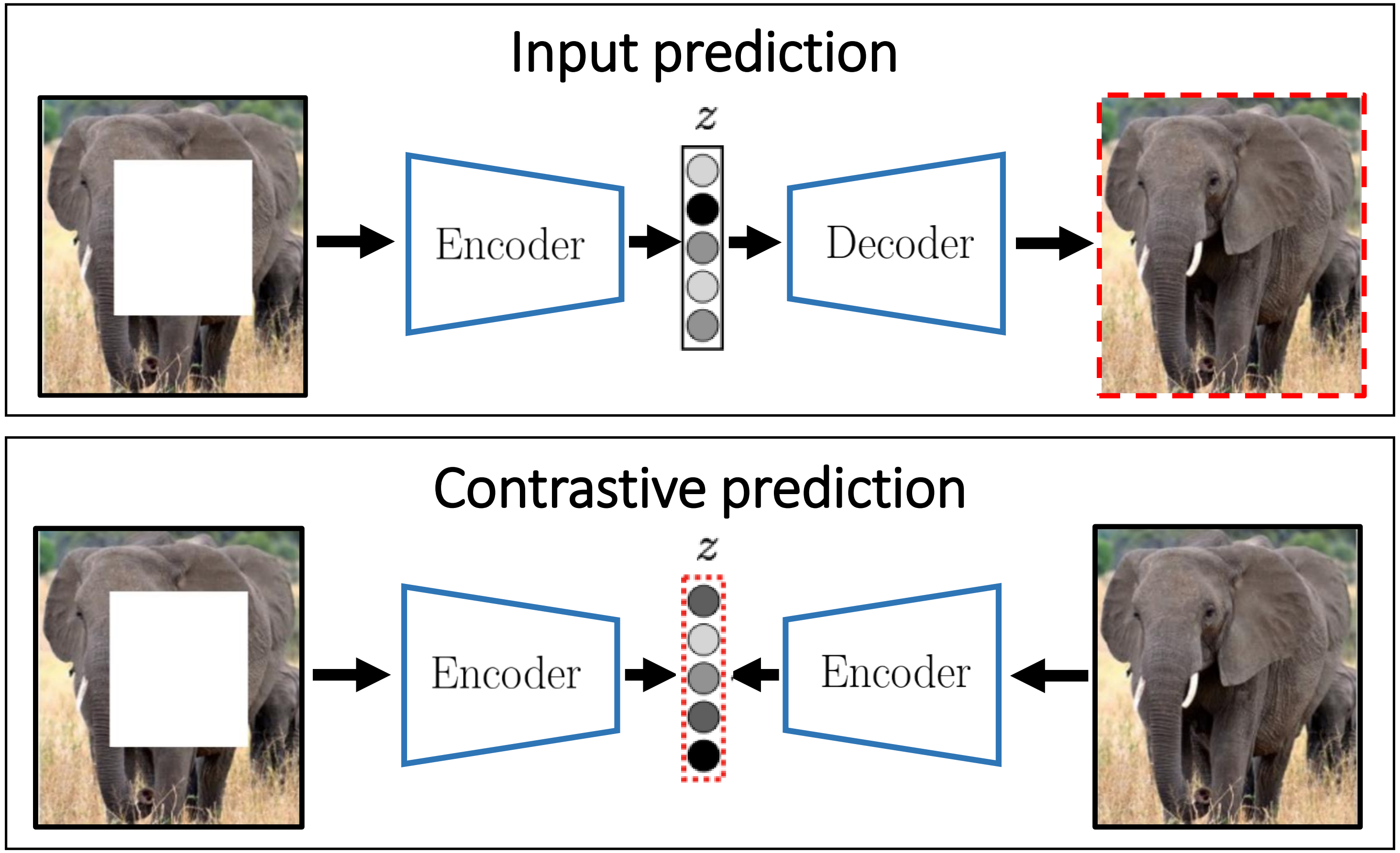


- **Hard and ambiguous task**
- **Effort spent on "useless" details**: exact color, good boundary, etc. Does not necessarily lead to features good for image understanding tasks

Contrastive learning

Formulates self-supervised tasks in terms of learned representations:

- Recognize different views of the same image in the presence of distracting negative image views
- Requires many negative examples
- How to choose negatives?
- Impossible to know whether a sample is actually negative or actually (i.e., from the same object)

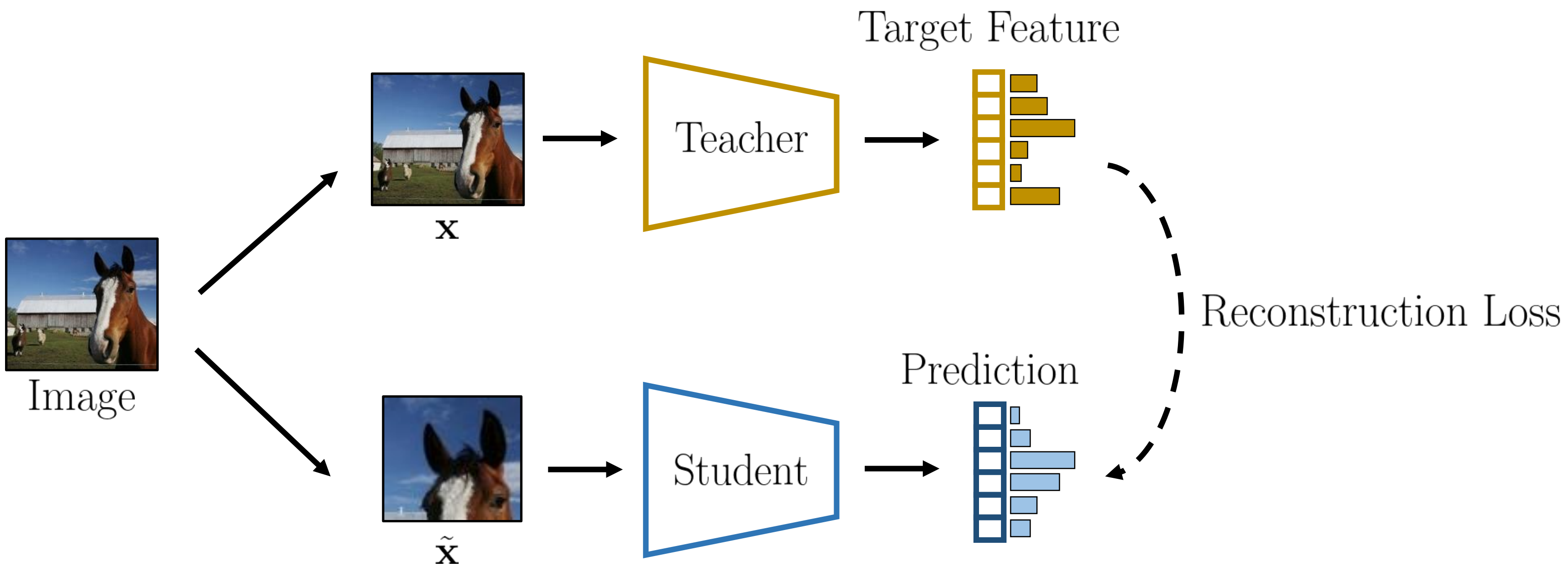


References:
"Representation learning with contrastive predictive coding", Oord et al, 2018
"Constraive multiview coding", Tian et al, 2020
"A simple framework for contrastive learning of visual representations", Chen et al, 2020
…

Agenda

- Input reconstruction
- Teacher-student feature reconstruction
- Wrap up evaluation

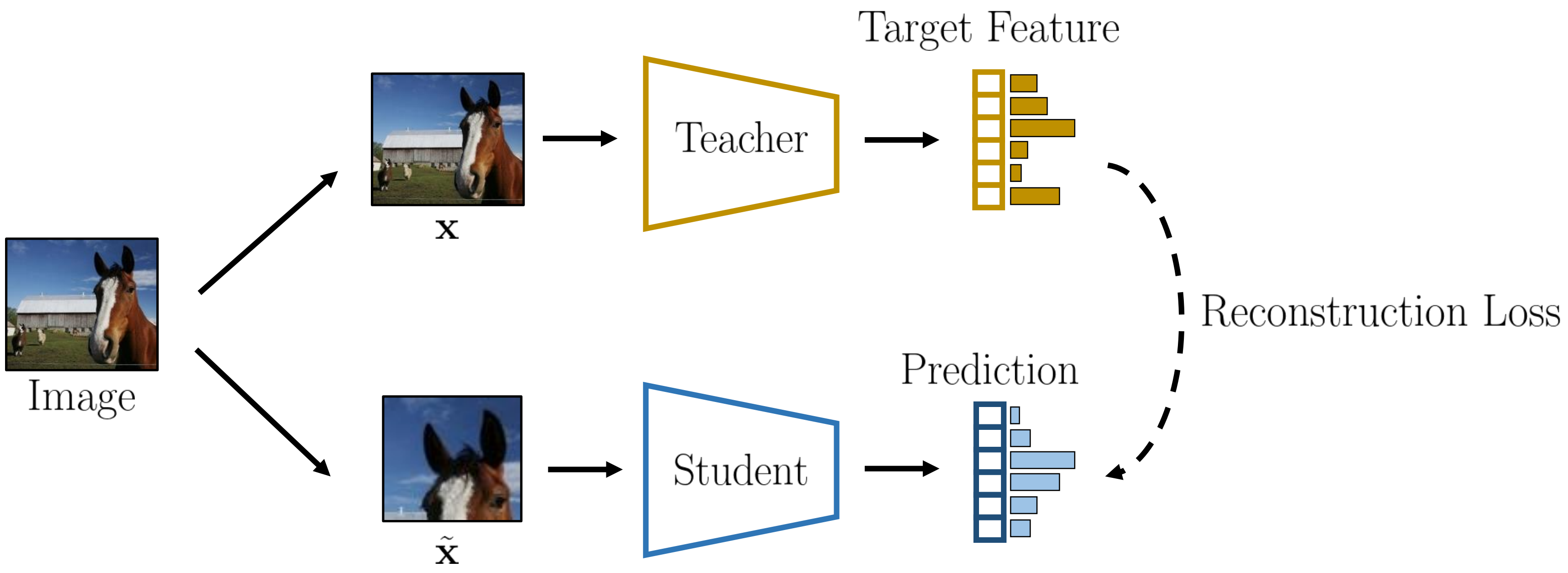
Teacher-student feature "reconstruction"



Teacher: generate a target feature vector from a given image

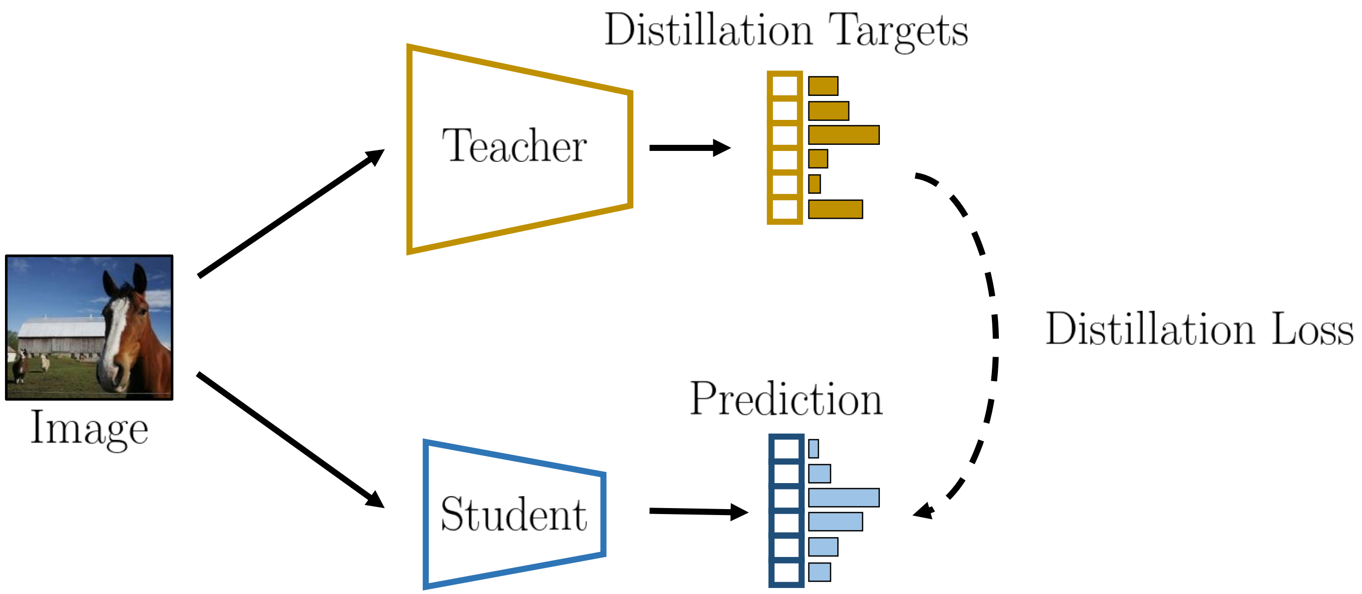
Student: predict this target, given as input a different random view of the same image

Teacher-student feature “reconstruction”



- Goal: **focus on reconstructing high-level visual concepts** rid of “useless” image details
- Enforces **perturbation-invariant representations** without **requiring negative examples**

Detour: teacher-student approaches for model compression

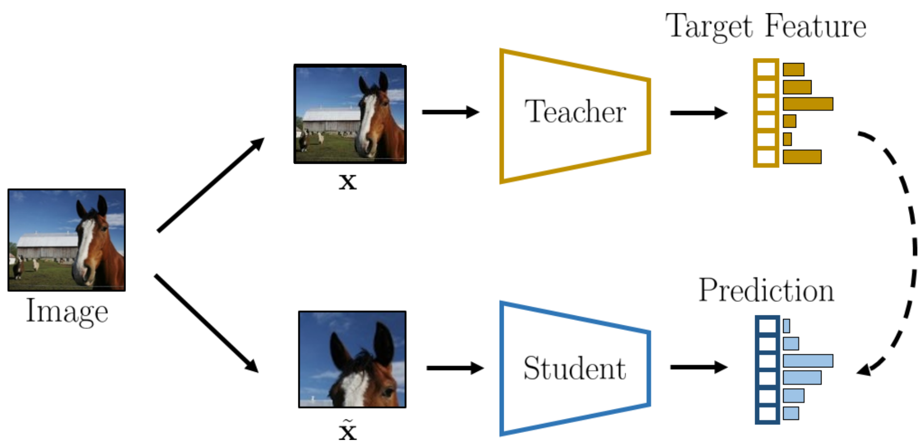


Goal: Distill the knowledge of a pre-trained teacher into a smaller student

- Commonly called **Knowledge Distillation**
- **Student:** trained to predict the teacher target when given the same input image
- Examples of targets: classification logits, intermediate features, attention maps, …

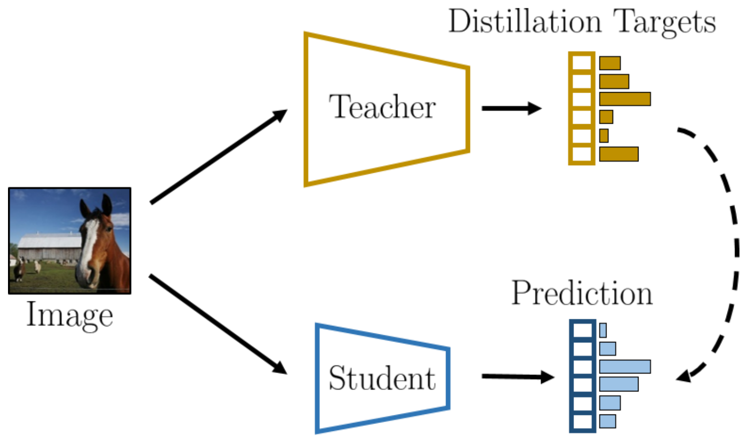
"Distilling the knowledge in a neural network", Hinton et al, 2018

Self-Supervised Learning



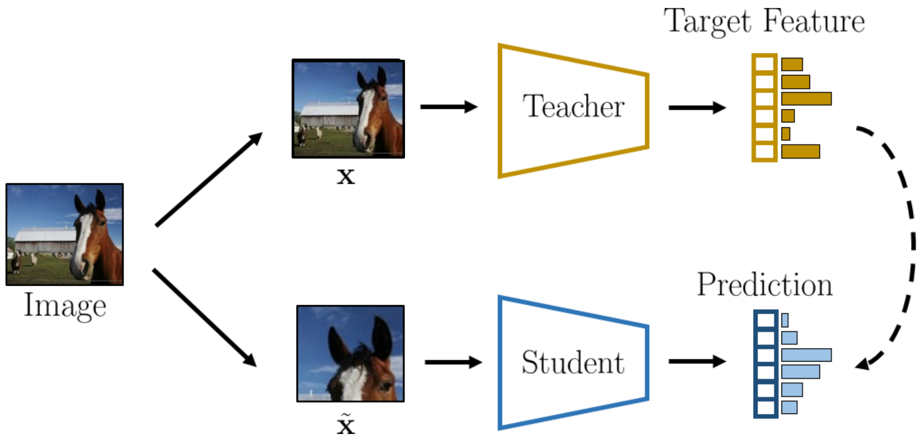
VS

Knowledge Distillation



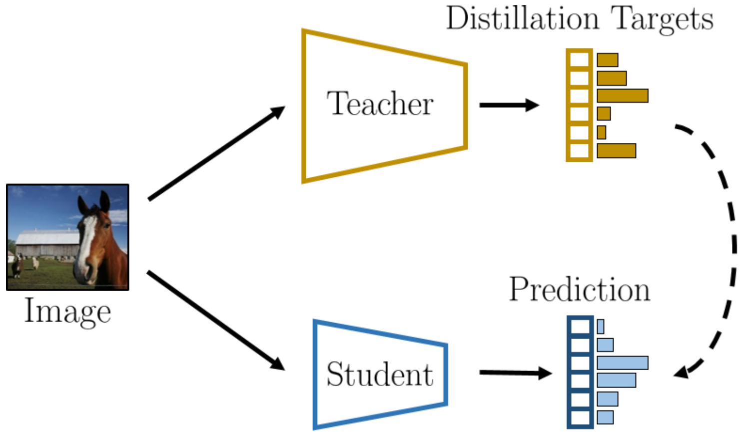
- Access to a "good" teacher
- (Typically) For the same exactly input, the outputs should match.
- (Typically) Hopefully the student would reach the teacher
- (Typically) The student network is smaller

Self-Supervised Learning



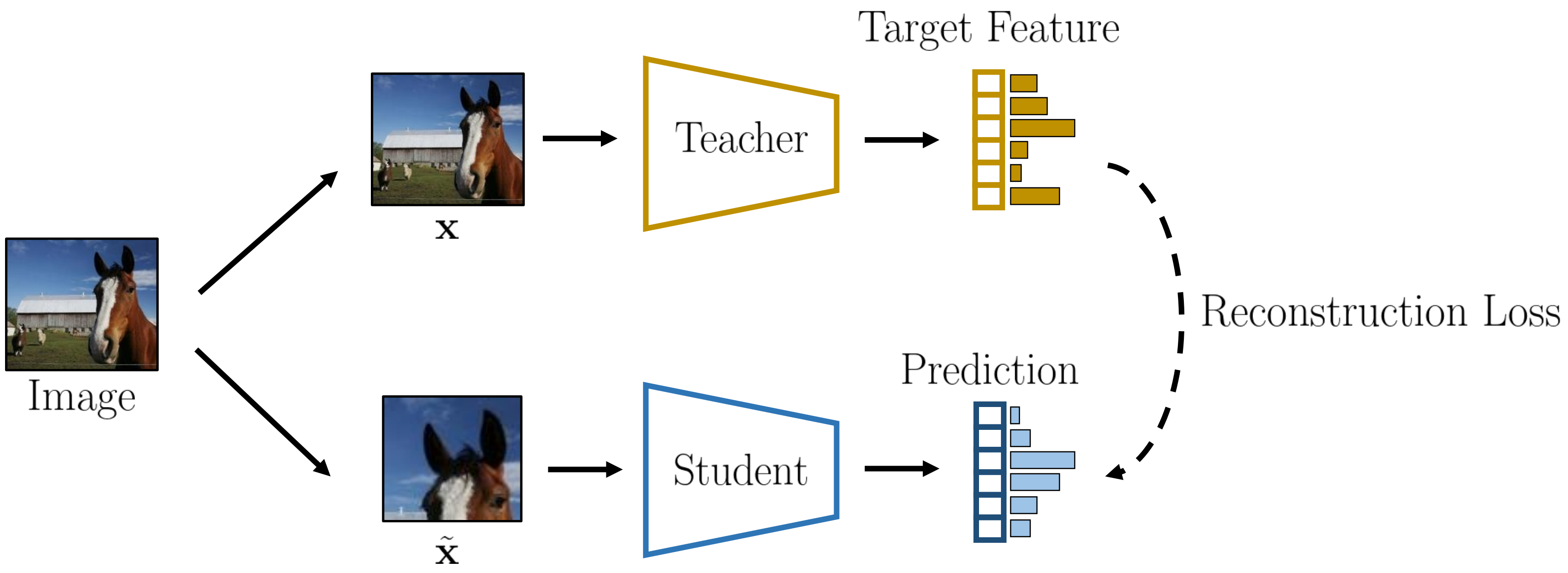
VS

Knowledge Distillation



- **No access to a "good" teacher**
- The student must predict the teacher output **given a different version of the image**
- The student **MUST surpass the initial teacher**
- **Both networks are of the same size**

Teacher-student feature "reconstruction"

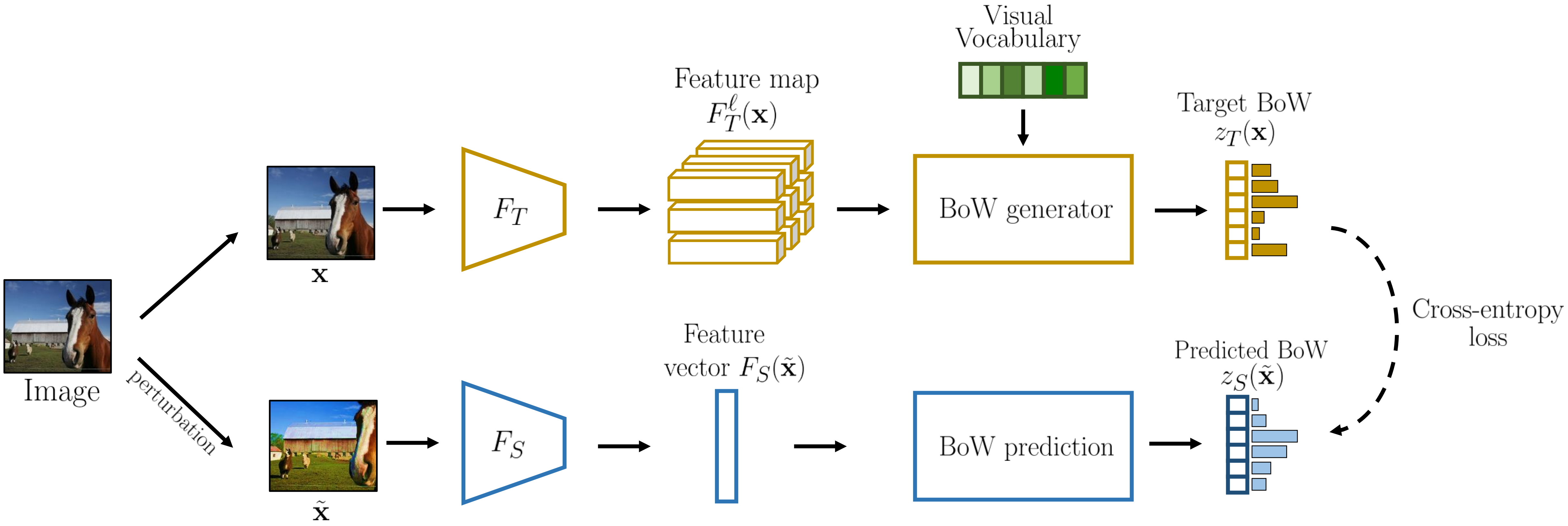


Key questions:

- What teacher to use?
- How to make the student surpass the teacher?
- What type of target features to use?

Feature “reconstruction” with static teachers

Predicting bag-of-words (BoWNet)

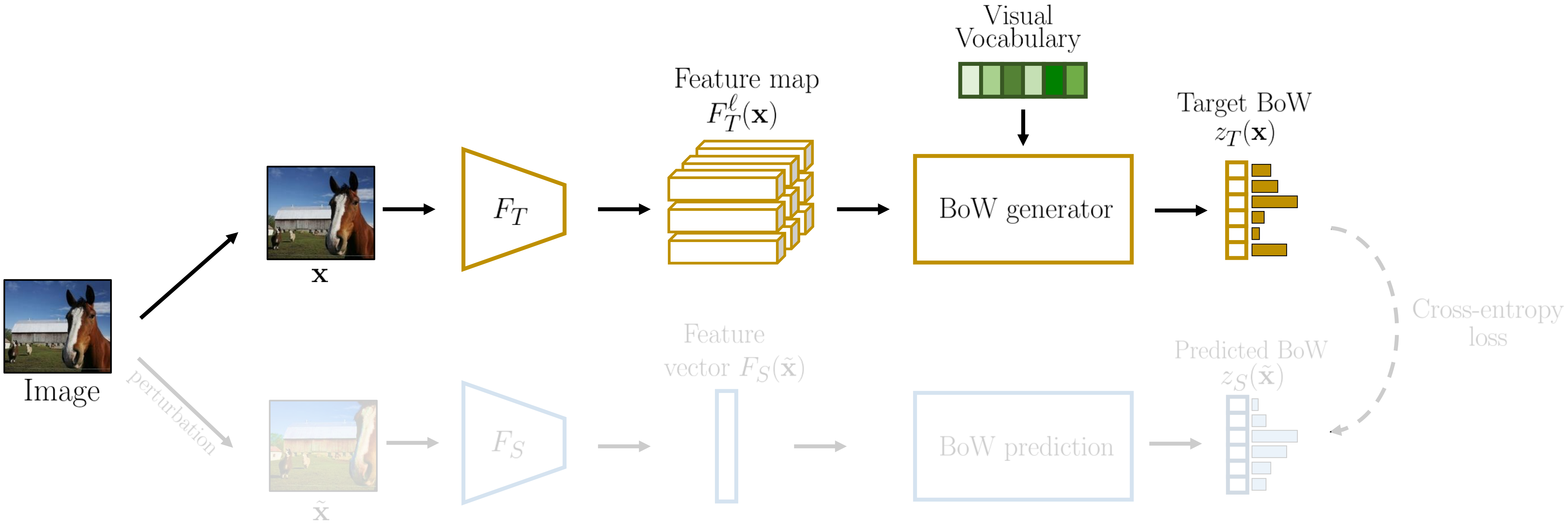


Feature reconstruction method defined over high-level discrete visual words:

- **Teacher:** extract feature maps + convert them to Bag-of-Words (BoW) vectors
- **Student:** must predict the BoW of an image, given as input a perturbed version

"Learning representations by predicting bags of visual words", Gidaris et al, CVPR 2020

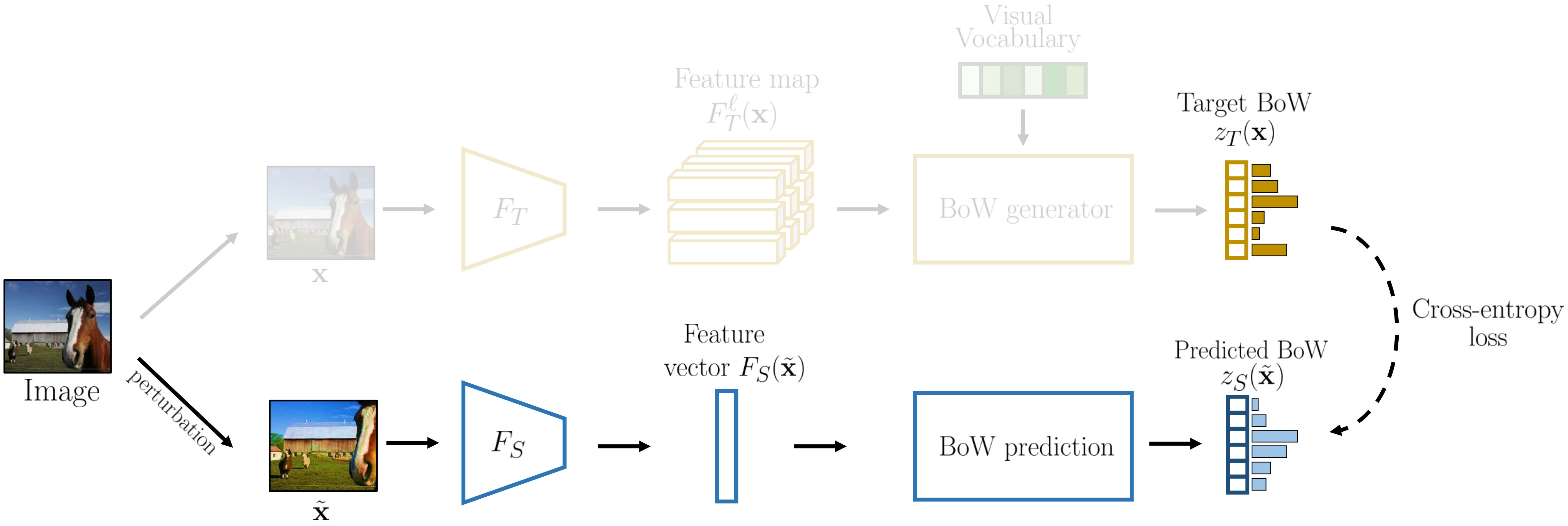
Predicting bag-of-words (BoWNet)



Feature reconstruction method defined over high-level discrete visual words:

- **Teacher:** extract feature maps + convert them to Bag-of-Words (BoW) vectors
- **Student:** must predict the BoW of an image, given as input a perturbed version

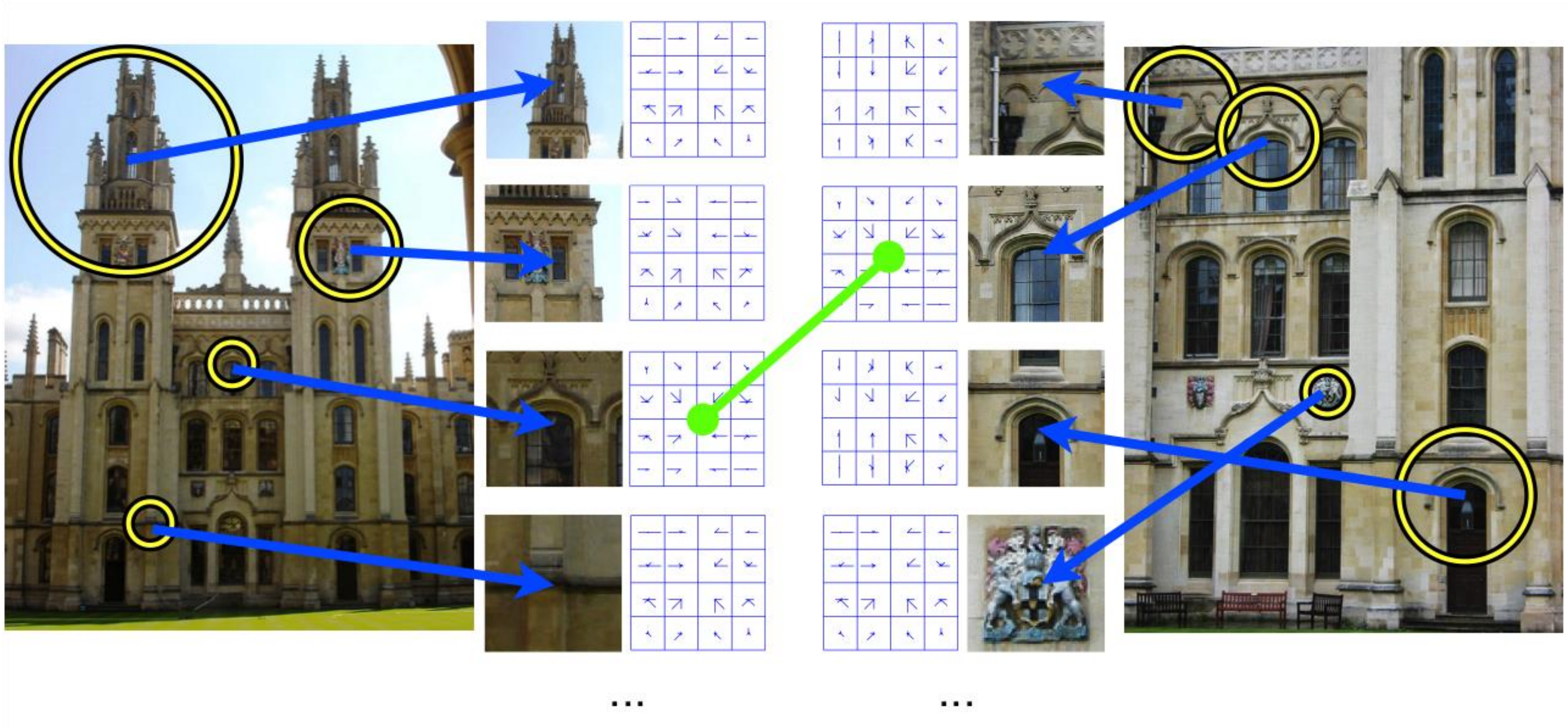
Predicting bag-of-words (BoWNet)



Feature reconstruction method defined over high-level discrete visual words:

- **Teacher:** extract feature maps + convert them to Bag-of-Words (BoW) vectors
- **Student:** must predict the BoW of an image, given as input a perturbed version

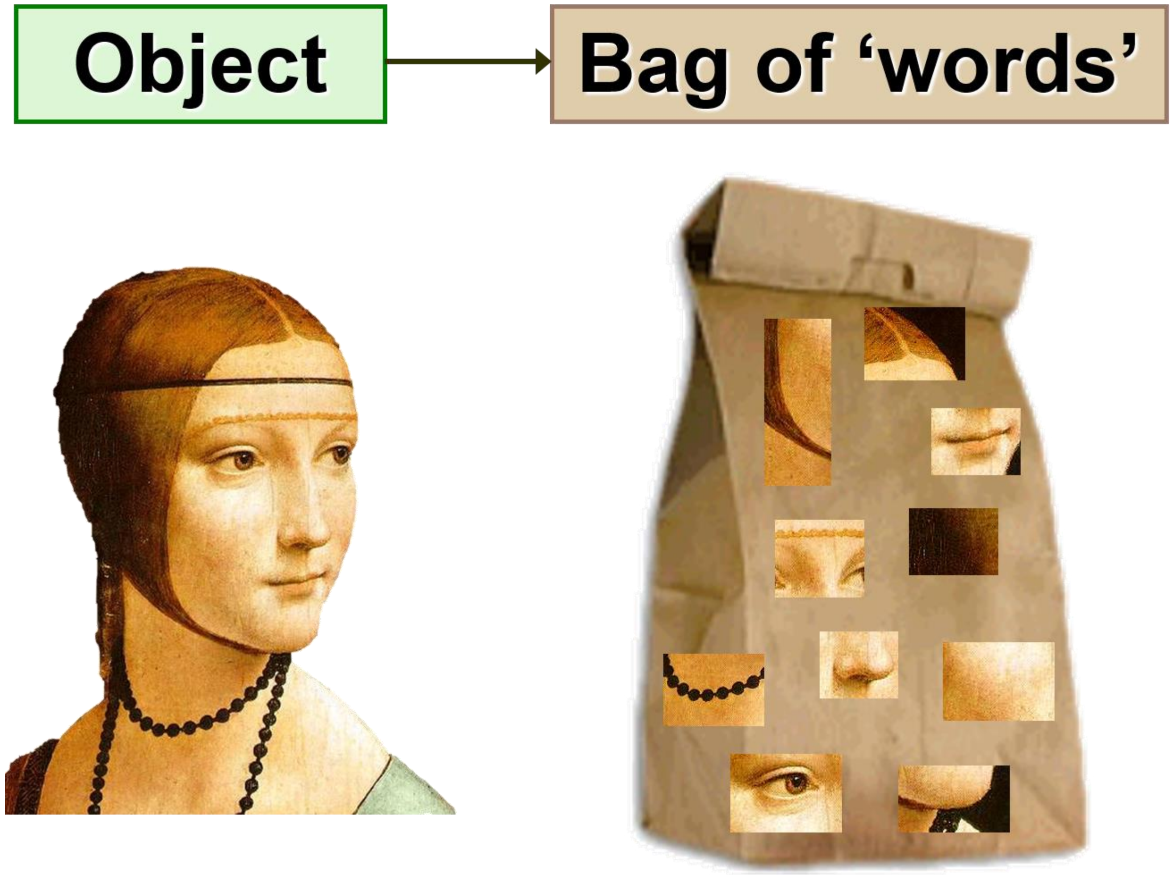
Bag-of-(visual-)words



Bag-of-(visual-)words are inspired from NLP. In computer vision are used for computing a single image-level descriptor from 100s-1000s of local patch descriptor

Slide credit: A. Vedaldi

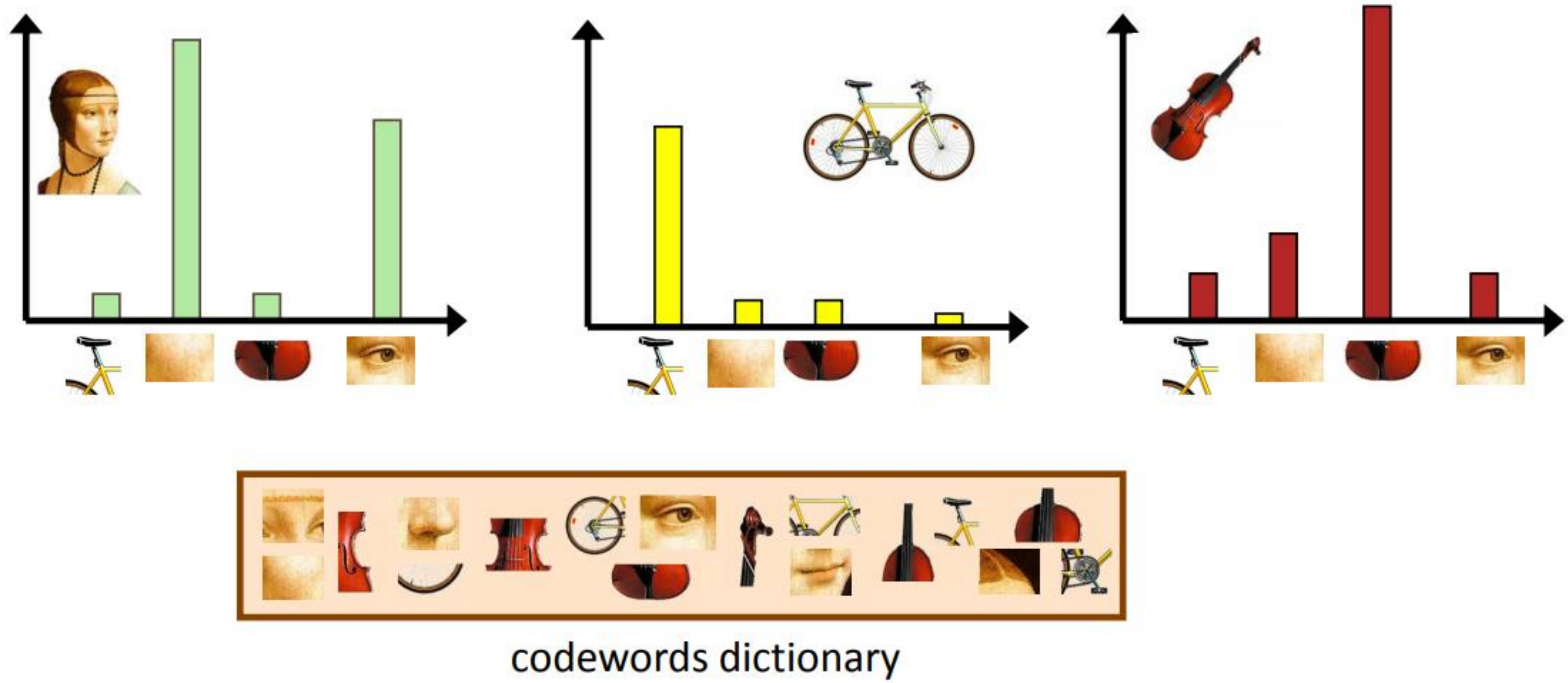
Bag-of-(visual-)words



Main idea: an object can be described by and recognized from statistics over local features

Slide credit: F.F. Li

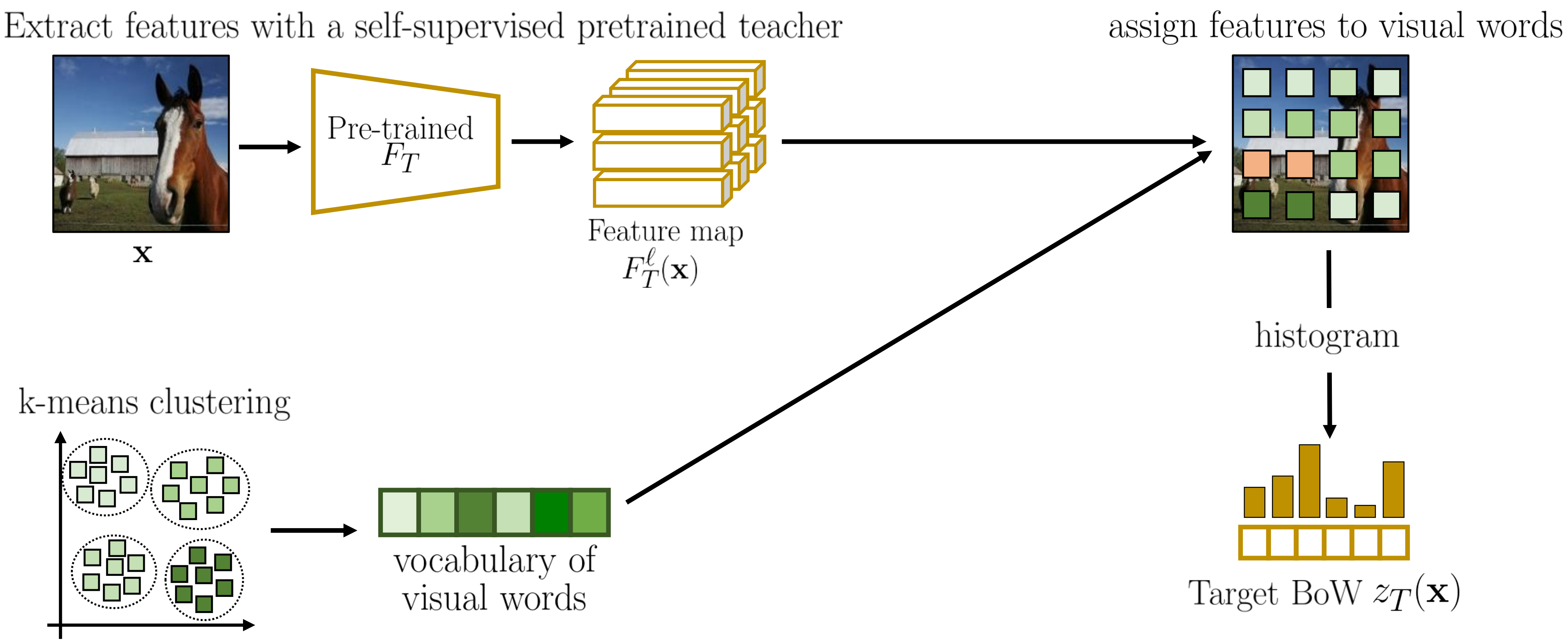
Bag-of-(visual-)words



- Compute a **dictionary of representative local features** (e.g., using k-means)
- Describe images as **histograms of occurrence** of these dictionary items in the image

Slide credit: F.F. Li

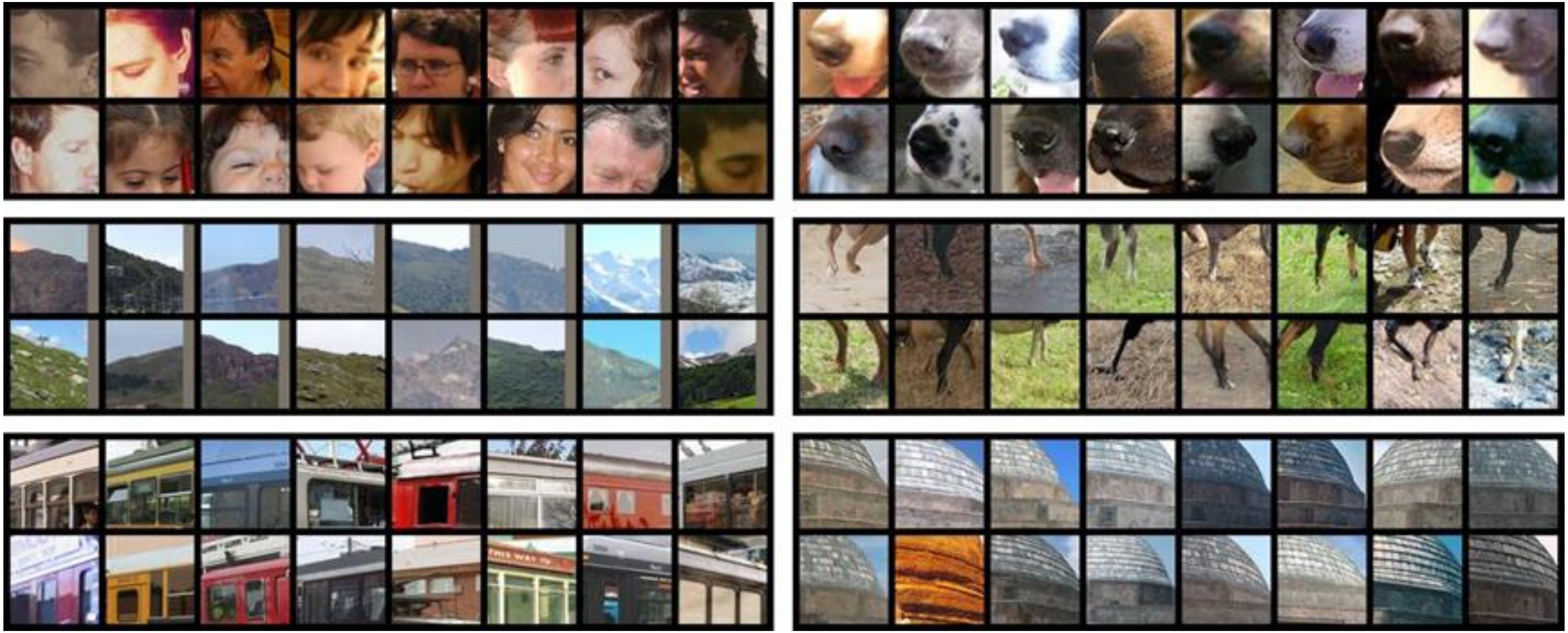
Teacher: BoW target generation



- Extract feature maps with another self-supervised pre-trained network (e.g., RotNet)
- Compute bag-of-words from the "pixels" of the teacher feature map

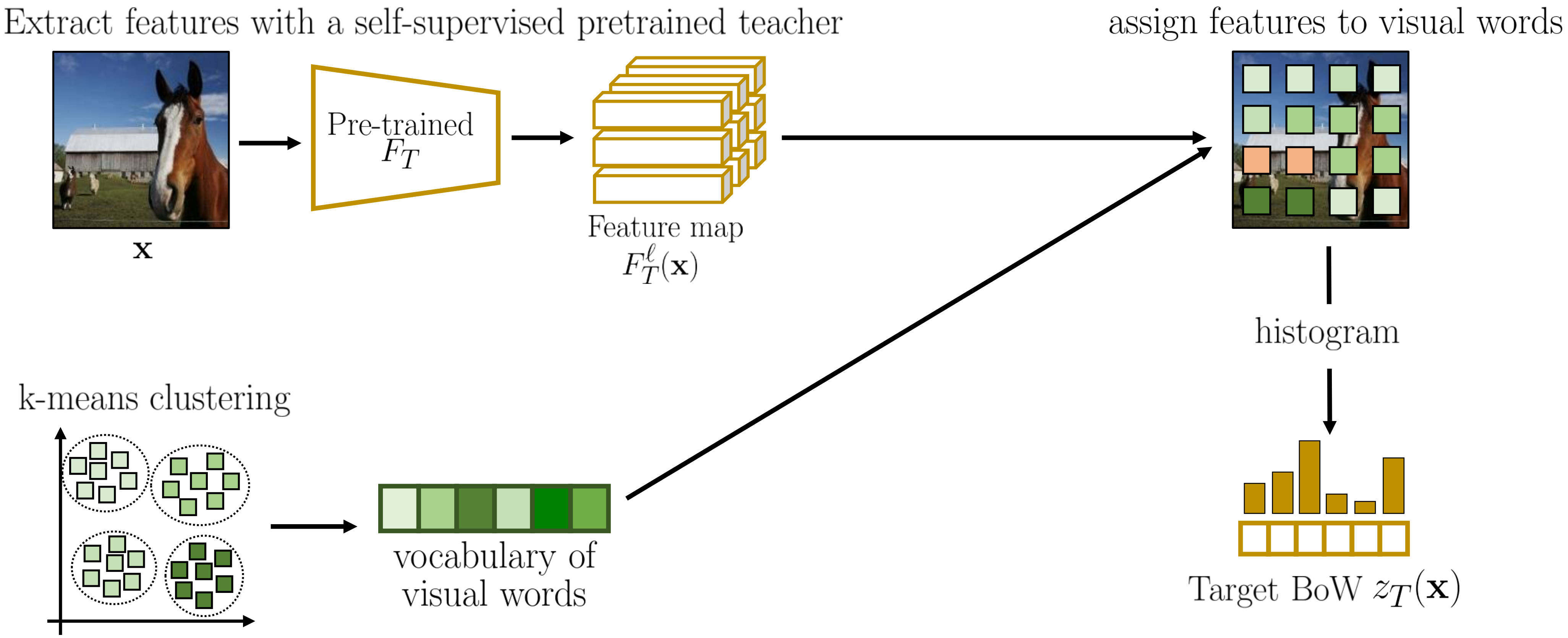
"Learning representations by predicting bags of visual words", Gidaris et al, CVPR 2020

Clusters of visual words



“Learning representations by predicting bags of visual words”, CVPR 2020

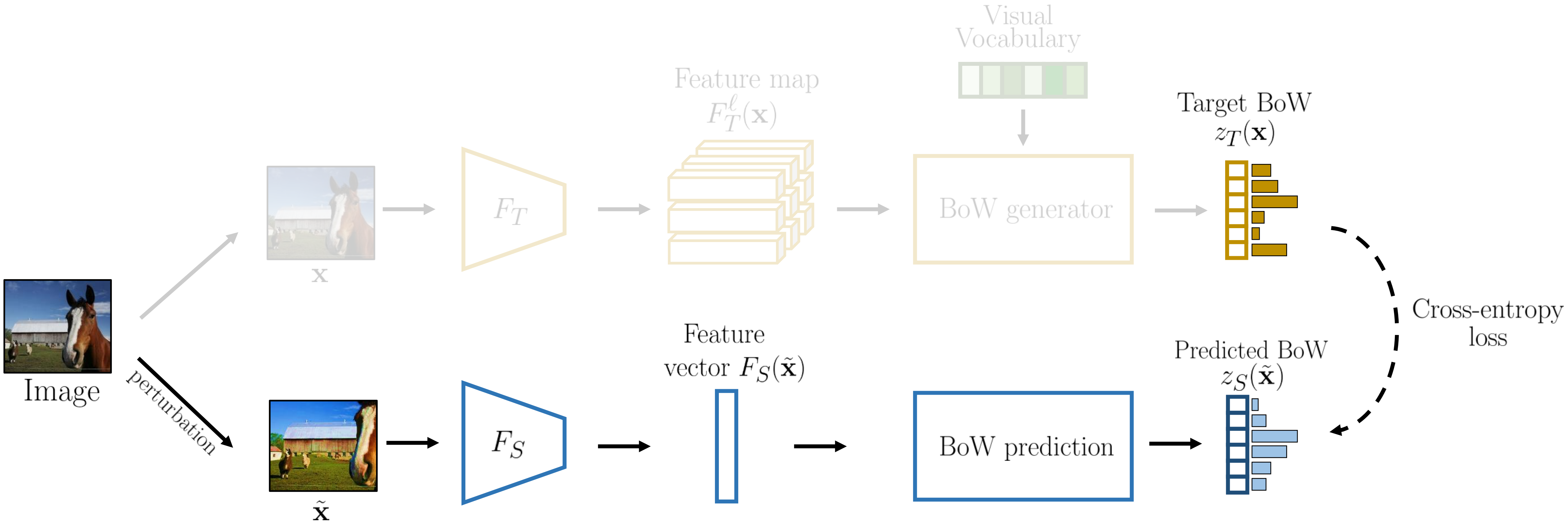
Teacher: BoW target generation



BoW targets: encode high-level image statistics from 100s of local features

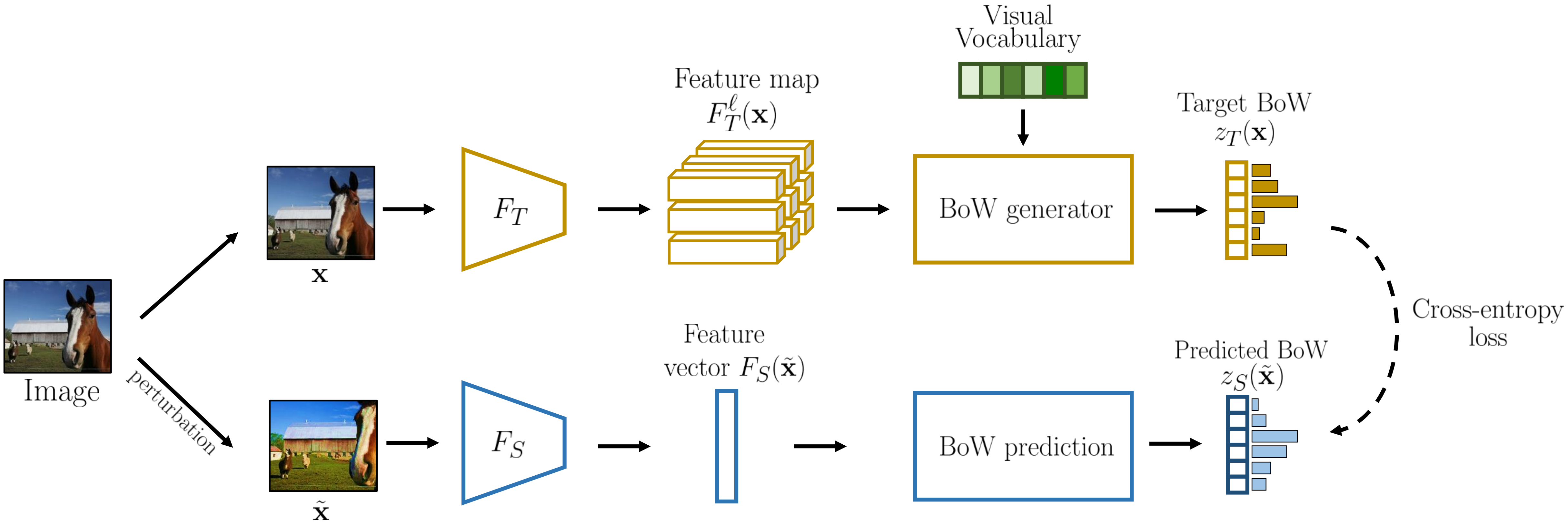
"Learning representations by predicting bags of visual words", Gidaris et al, CVPR 2020

Student: BoW prediction



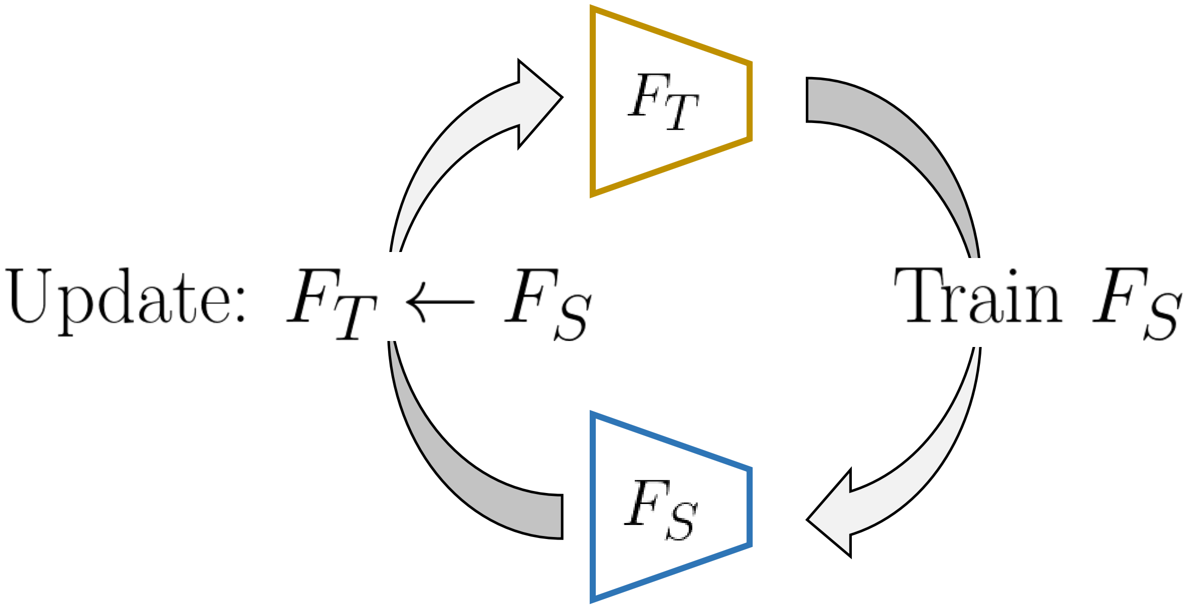
- **Feature extractor** F_S: extract a global feature vector from the image
- **BoW prediction:** implemented with a fully connected layer followed by softmax
- **Loss: cross-entropy** between the predicted softmax BoW distribution and the target BoW

Asymmetric architecture



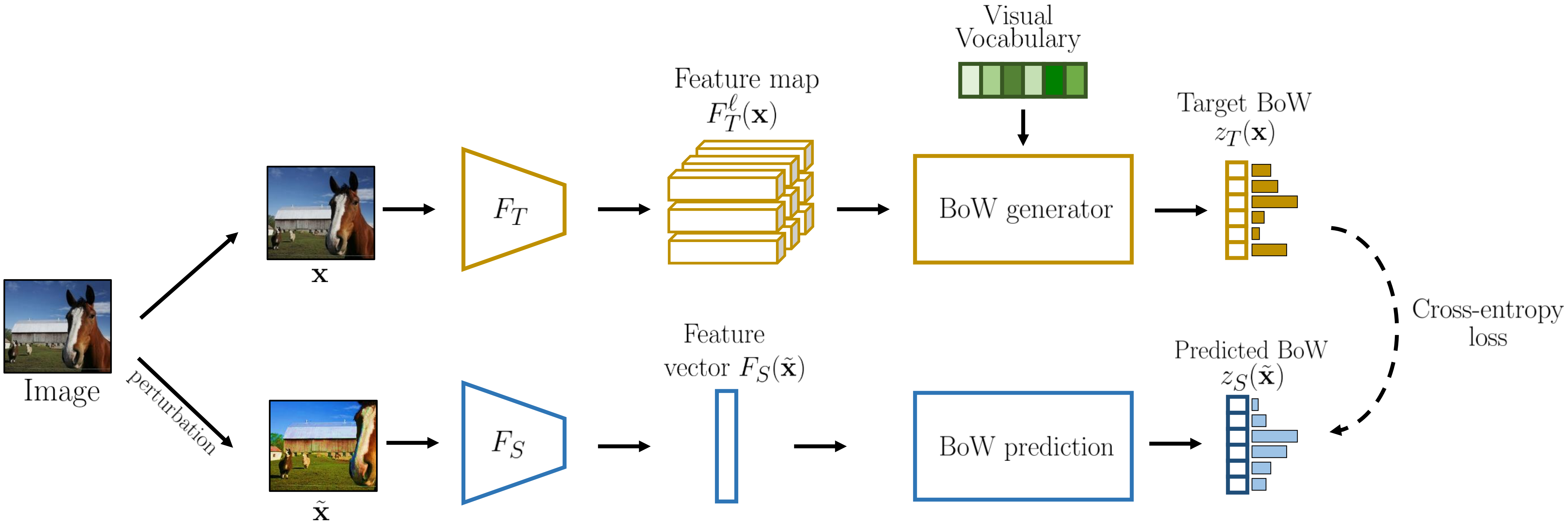
- **Teacher:** generates a BoW target from the feature map of an image
- **Student:** predicts a BoW using the global feature vector of an image

Model initialization and iterated training



1. Start from a self-supervised **pre-trained teacher**
2. Train the student on the BoW prediction task **till convergence (e.g., 100s of epochs)**
3. Update the teacher with the new student and repeat **the training process (go to step 2)**

Predicting bag-of-words (BoWNet)



- **BoW reconstruction task:** enforces the learning of
  1. **Perturbation invariant representations**
  2. **Contextual reasoning skills**: infer words of missing image regions
- **The new student surpasses the initial pre-trained (RotNet) teacher**

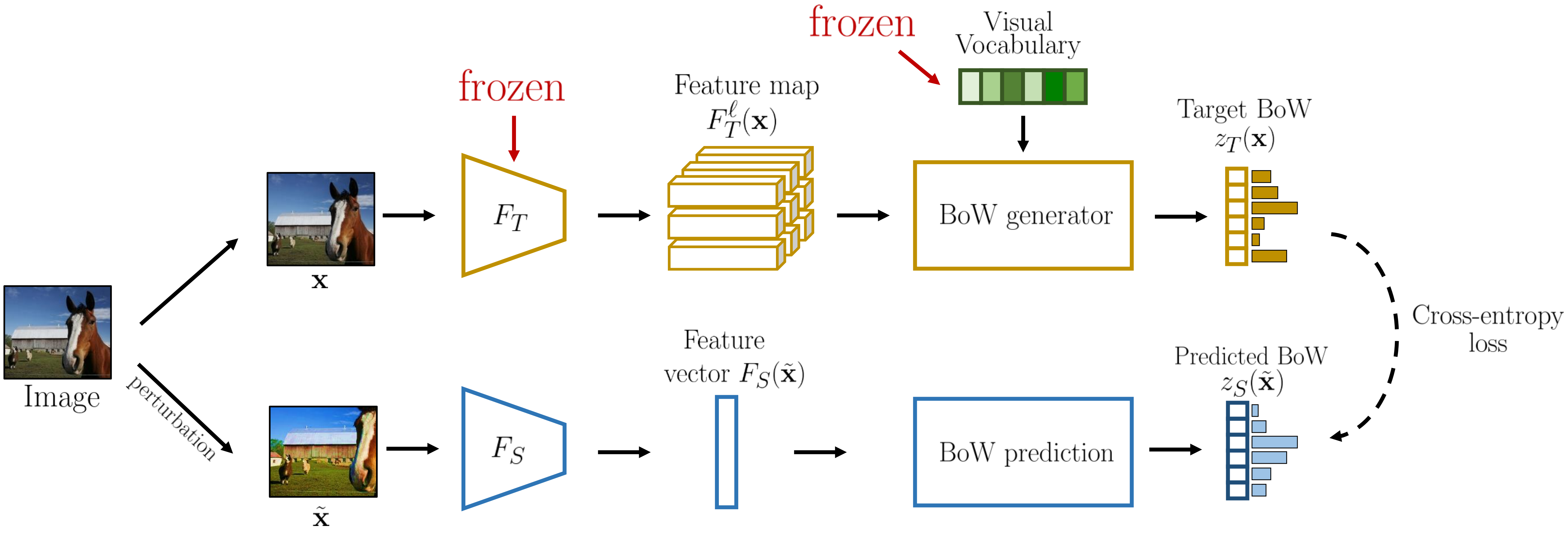
Surpassing the teacher network

Classes / Method	Novel $n = 1$	5	10	50	Base Linear
RotNet	40.8	56.9	61.8	68.1	52.3
BoWNet	48.7	67.9	74.0	79.9	65.0
BoWNet $\times 2$	**49.1**	67.6	73.6	79.9	65.6
BoWNet $\times 3$	48.6	**68.9**	**75.3**	**82.5**	**66.0**

Table 2: MiniImageNet linear classifier and few-shot results with WRN-28-4.

The new student surpasses the initial pre-trained (RotNet) teacher

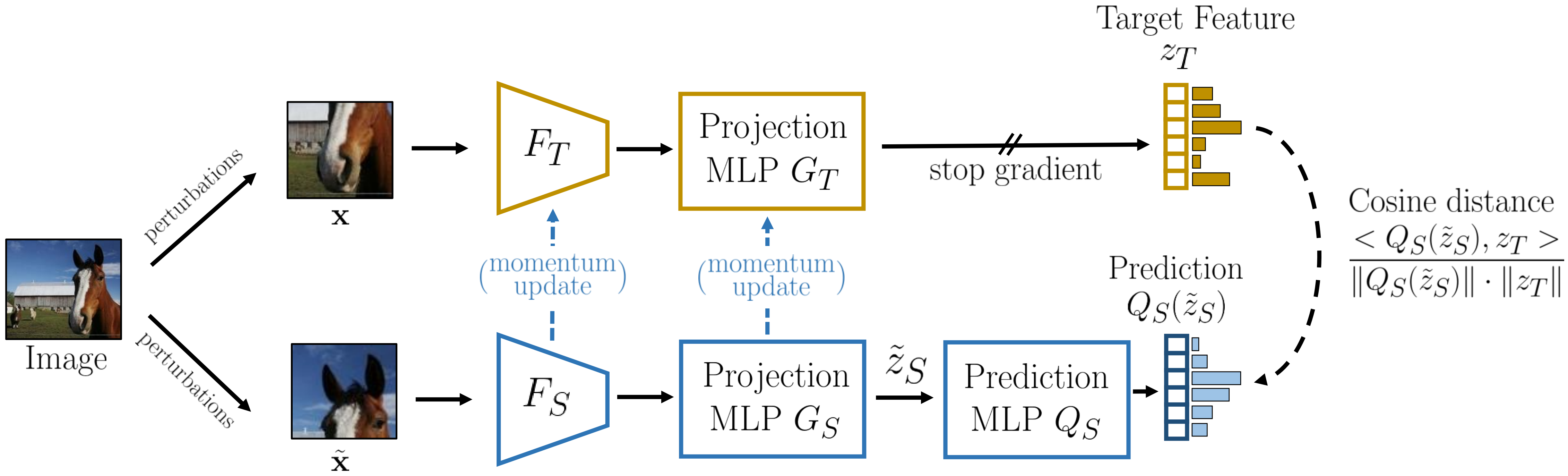
Limitation of BoWNet



- Requires pre-training the teacher with another self-supervised method
- The teacher remains frozen throughout long training cycles
- Leads to suboptimal supervisory signal to the student / slow convergence

Dynamic teacher-student feature "reconstruction" methods

Bootstrap Your Own Latent (BYOL)

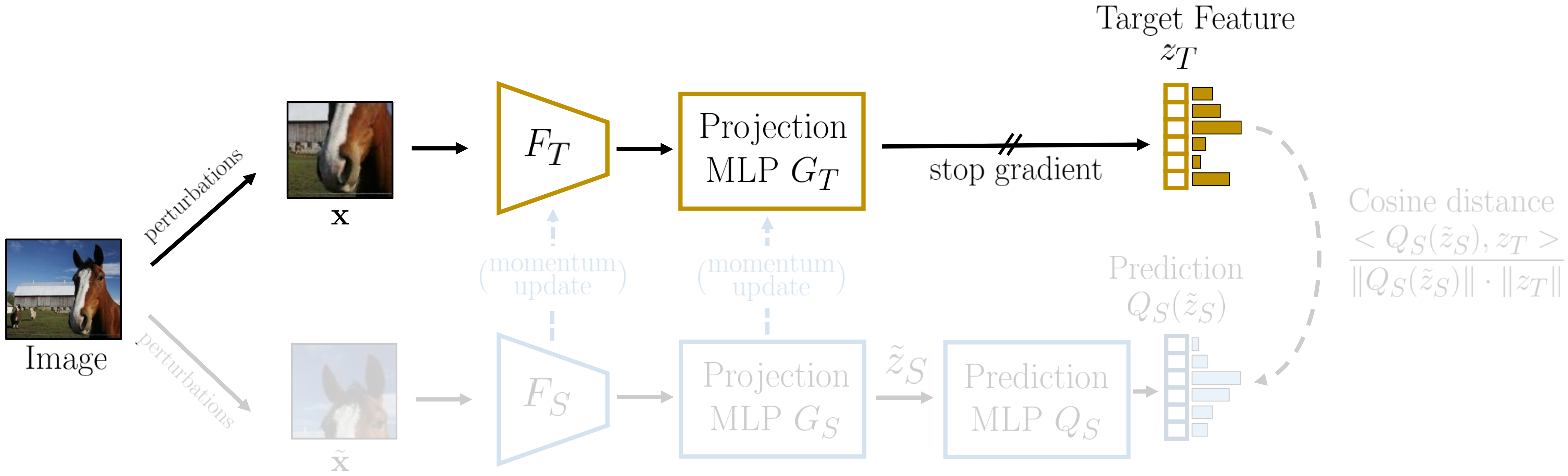


Feature reconstruction method:

- **Teacher:** extract a target feature vector from a random view of an image
- **Student:** predict this target, given as input a different random view of the same image

“Bootstrap Your Own Latent: a new approach to self-supervised learning”, NeurIPs 2020

Bootstrap Your Own Latent (BYOL)

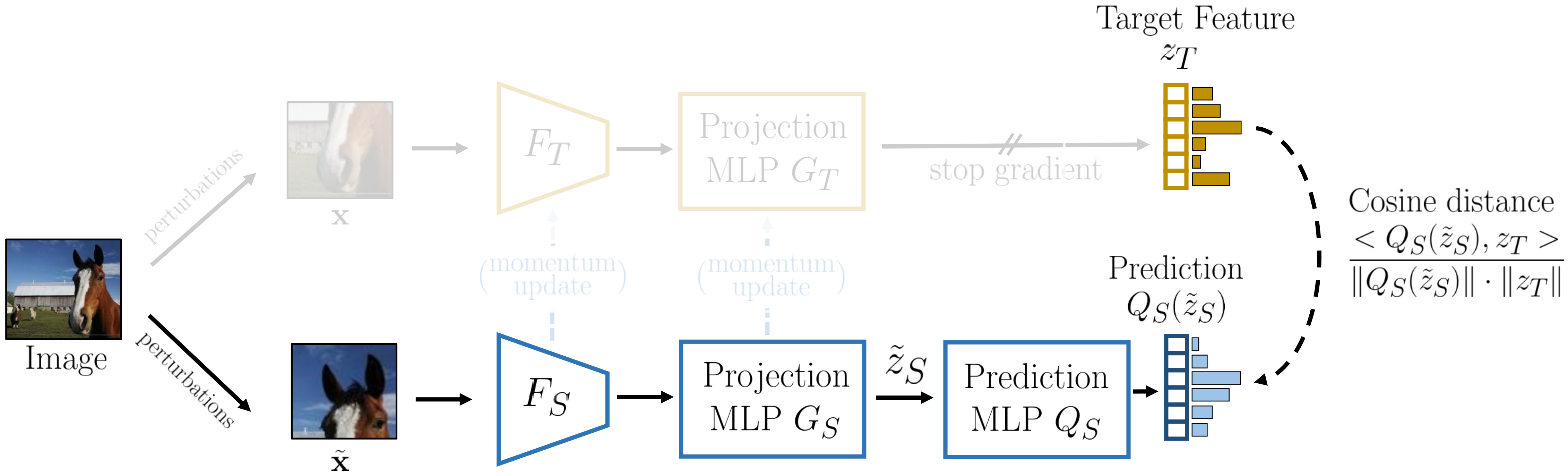


Feature reconstruction method:

- **Teacher:** extract a target feature vector from a random view of an image
- **Student:** predict this target, given as input a different random view of the same image

"Bootstrap Your Own Latent: a new approach to self-supervised learning", NeurIPs 2020

Bootstrap Your Own Latent (BYOL)

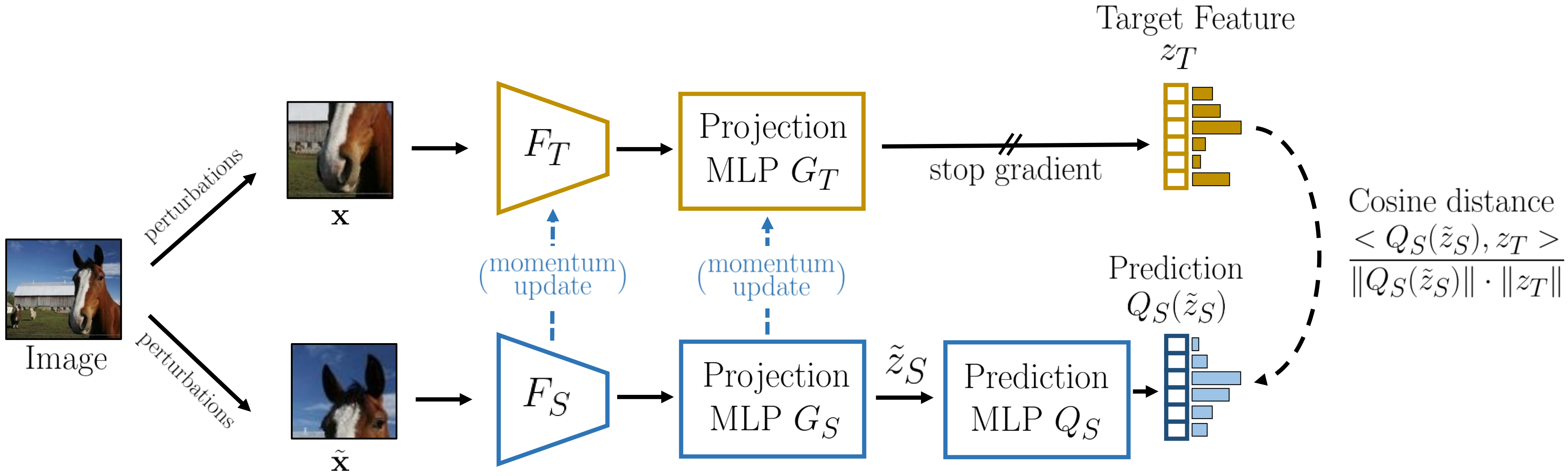


Feature reconstruction method:

- **Teacher:** extract a target feature vector from a random view of an image
- **Student:** predict this target, given as input a different random view of the same image

"Bootstrap Your Own Latent: a new approach to self-supervised learning", NeurIPs 2020

Bootstrap Your Own Latent (BYOL)

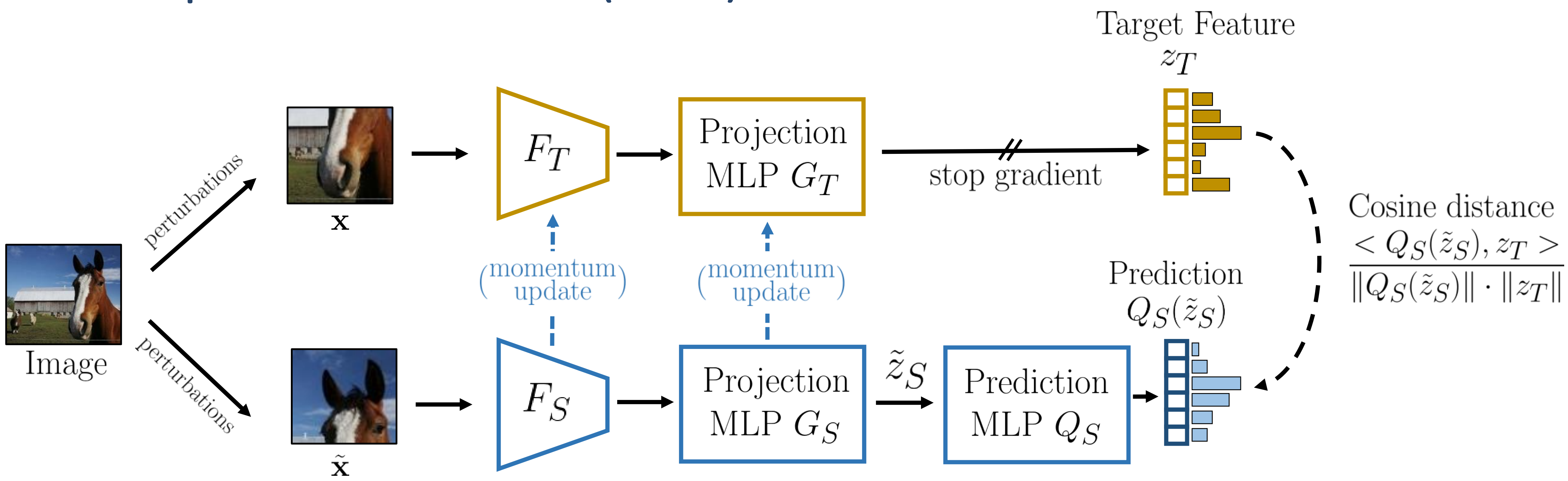


Feature reconstruction method:

- **Teacher:** extract a target feature vector from a random view of an image
- **Student:** predict this target, given as input a different random view of the same image
- **Symmetric loss:** from $\tilde{\mathbf{x}}$ predict the target of $\mathbf{x}$ and from $\mathbf{x}$ predict the target of $\tilde{\mathbf{x}}$

"Bootstrap Your Own Latent: a new approach to self-supervised learning", NeurIPs 2020

Bootstrap Your Own Latent (BYOL)

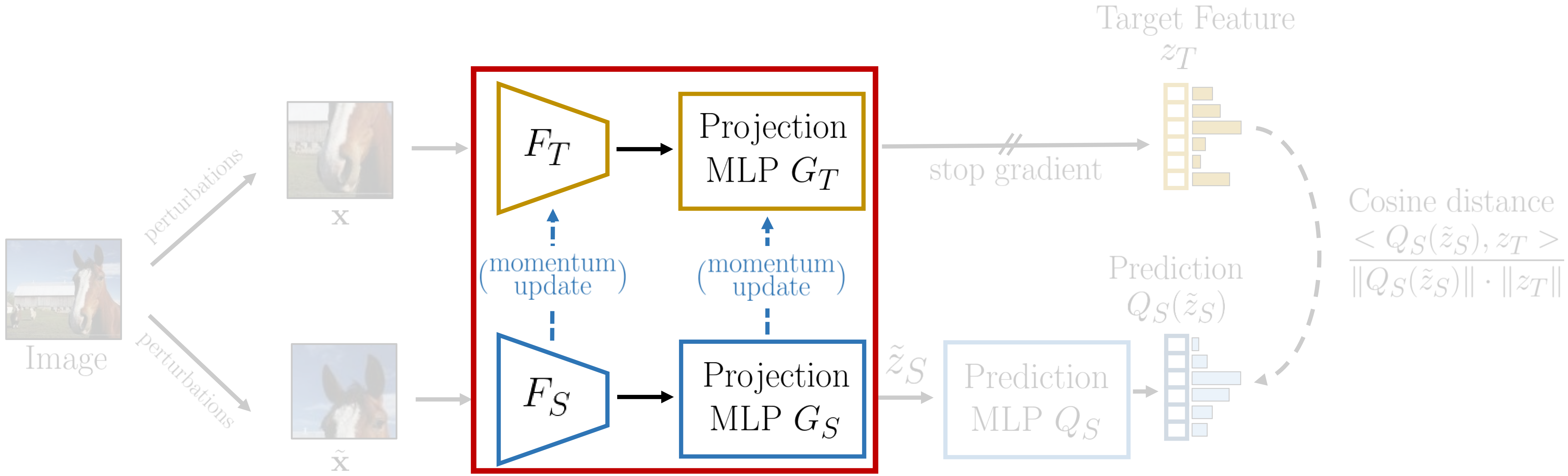


Bootstrap idea: builds a sequence of student representations of increasing quality

- Given a teacher, train a new enhanced student by predicting the teacher's features
- Iteratively apply this procedure by updating the teacher with the new student

"Bootstrap Your Own Latent: a new approach to self-supervised learning", NeurIPs 2020

Online updating the teacher with exponential moving average



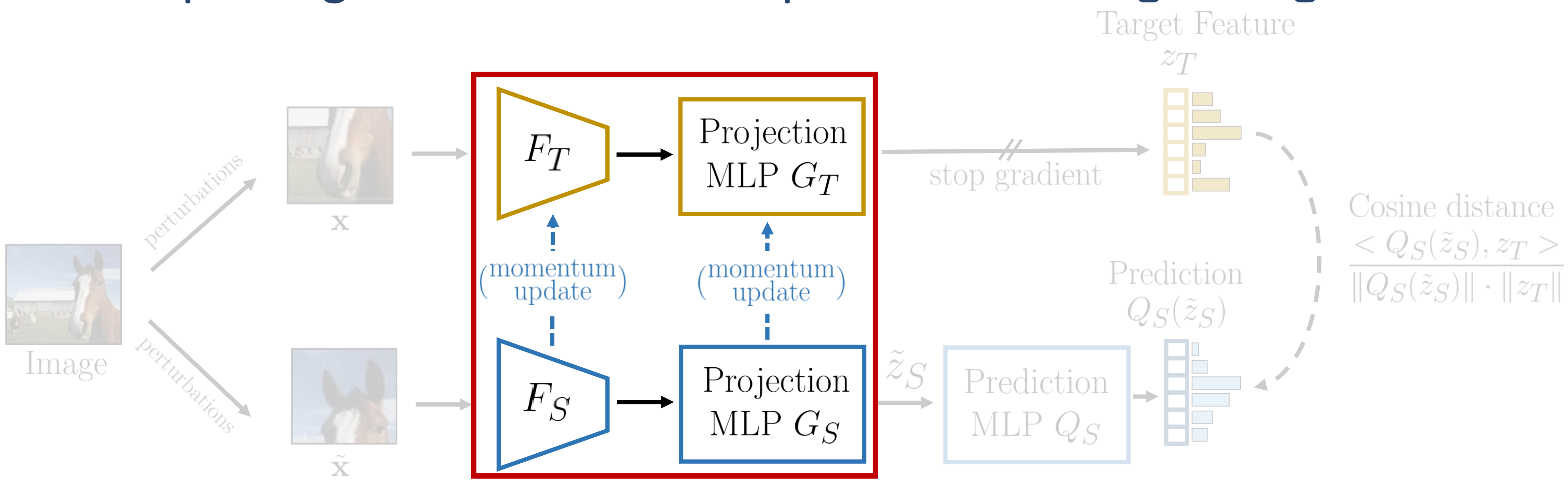
Use exponential moving average for online updating the teacher at each training step:

$$\theta_{\mathrm{T}}^{(t)} \leftarrow \alpha \cdot \theta_{\mathrm{T}}^{(t-1)} + (1-\alpha) \cdot \theta_{\mathrm{S}}^{(t)}$$

θ_{T} : teacher parameters

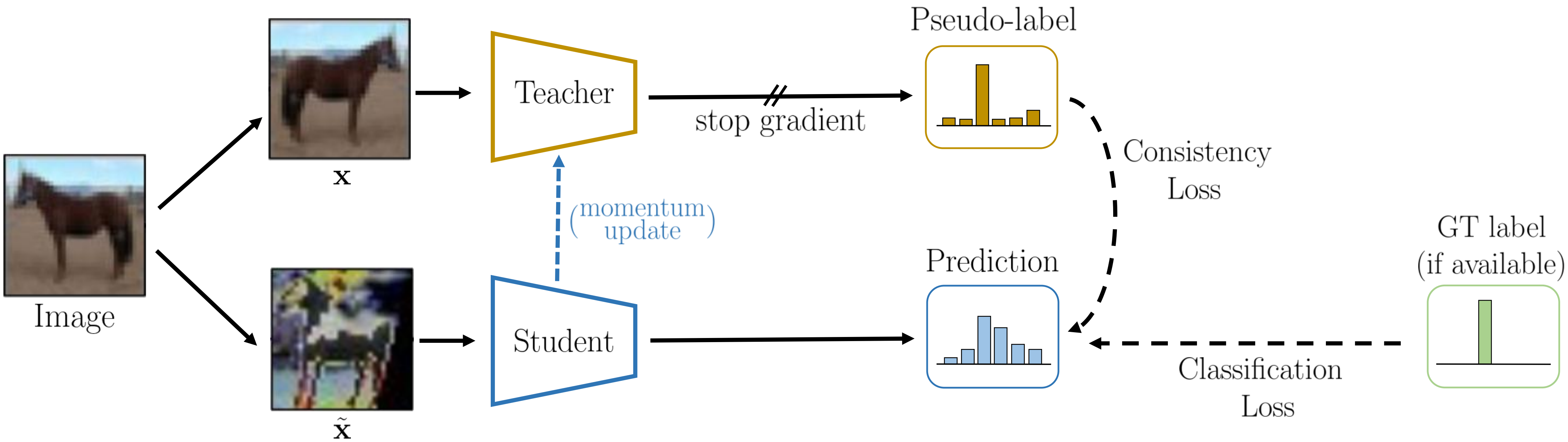
θ_{S} : student parameters

Online updating the teacher with exponential moving average



This type of teacher is typically called **momentum or mean teacher.**

Detour: mean / momentum teacher in semi-supervised learning

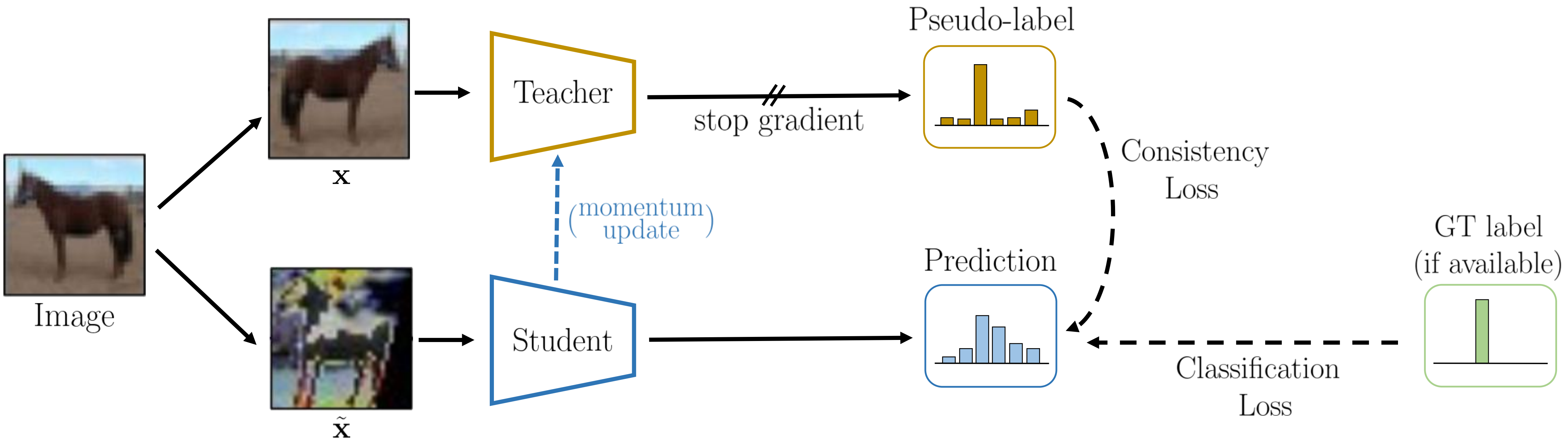


Teacher-student approaches are common in semi-supervised learning:

- **Teacher:** generate target classification predictions from an image
- **Student:** trained to predict this target given a different random view of the same image

"Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results", NeurIPs 2017

Detour: mean / momentum teacher in semi-supervised learning

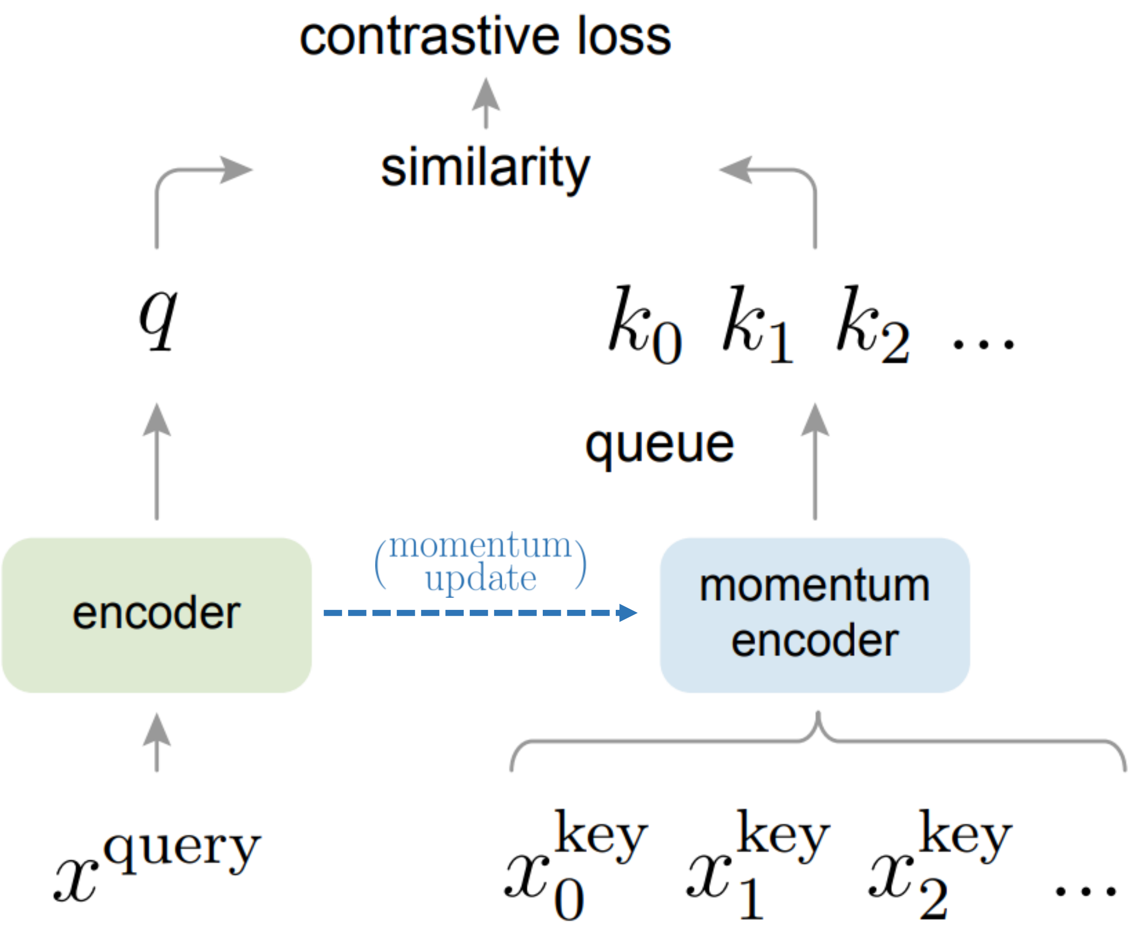


Mean teachers have been shown to improve the results:

- Similar to temporal ensembles of the student model but instead of averaging the predictions it averages the model weights
- More stable and accurate version of the student

“Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results”, NeurIPs 2017

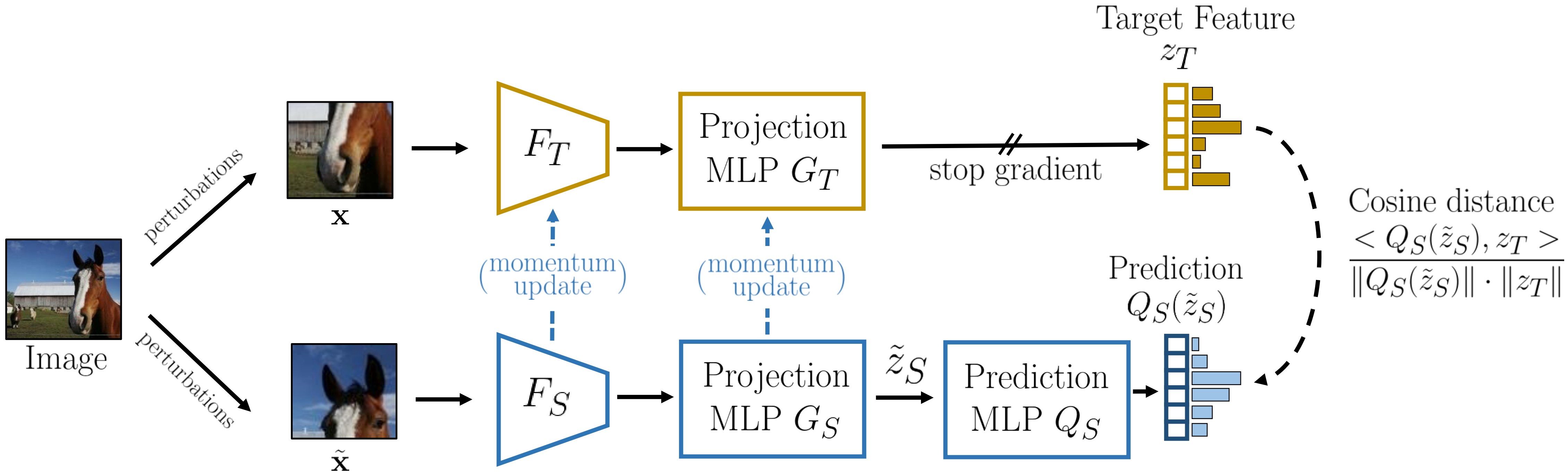
Detour: momentum / mean teacher in contrastive learning



MoCo exploits a momentum encoder network for maintaining a large and consistent dictionary of keys (positives + negatives examples) for contrastive learning.

"Momentum Contrast for Unsupervised Visual Representation Learning", CVPR 2020

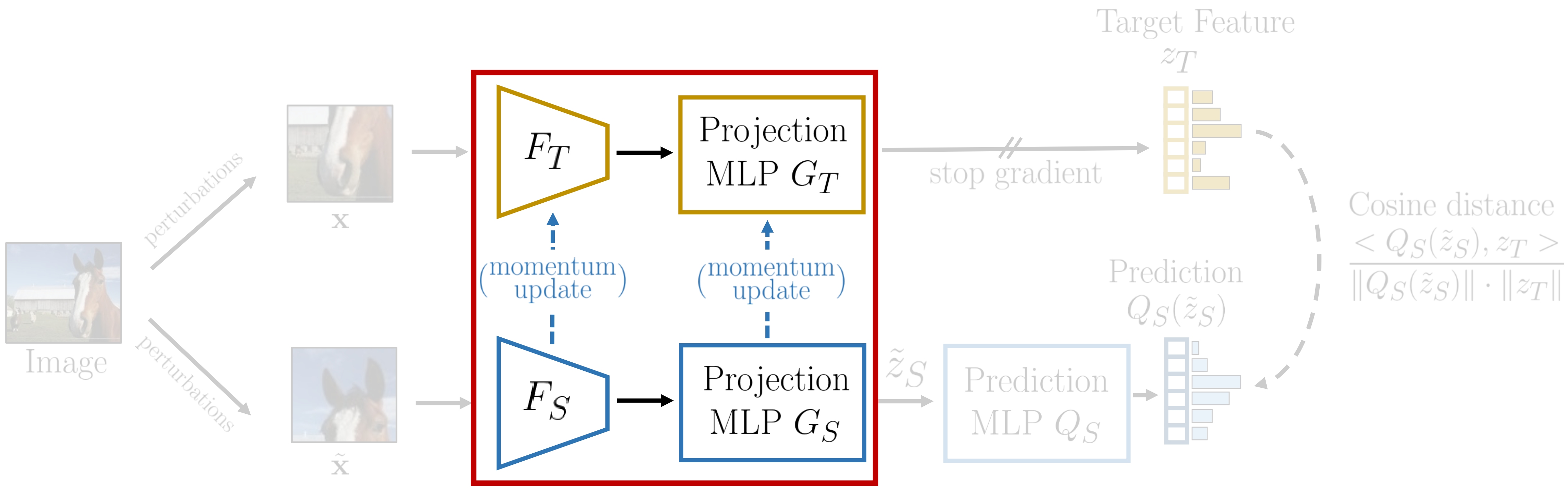
Back to BYOL - Mean teacher for the feature reconstruction task



A mean teacher approach without any labels

- Offers stable but slowly evolving feature targets
- More efficient than using a fixed pre-training teacher that is updated only after the end of each training cycle (as BoWNet does)

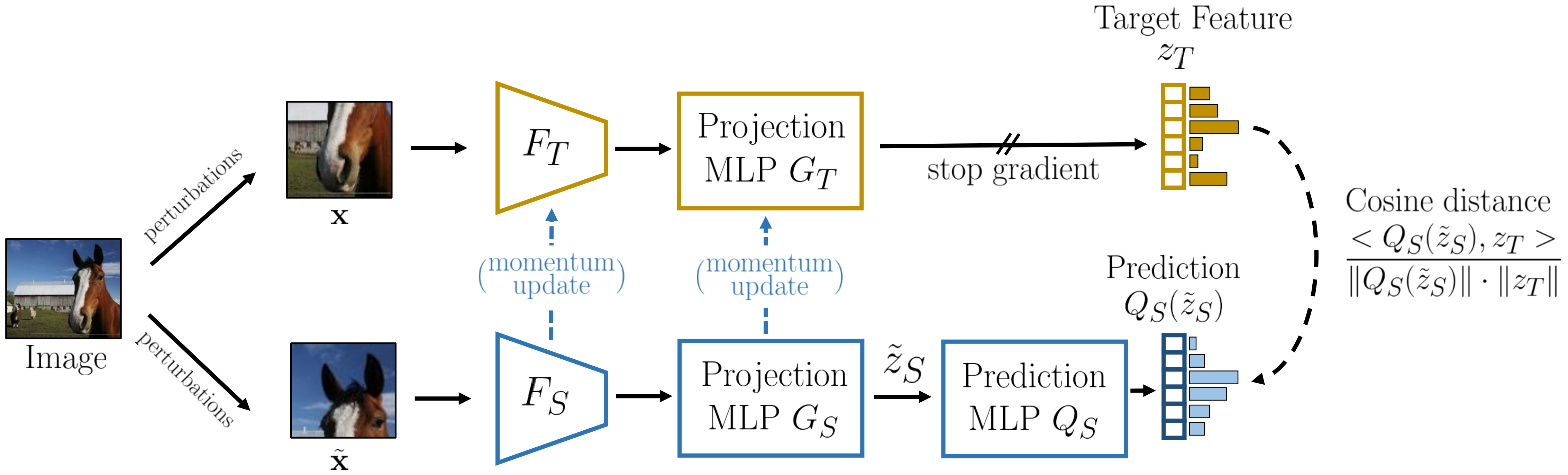
Back to BYOL - Mean teacher for the feature reconstruction task



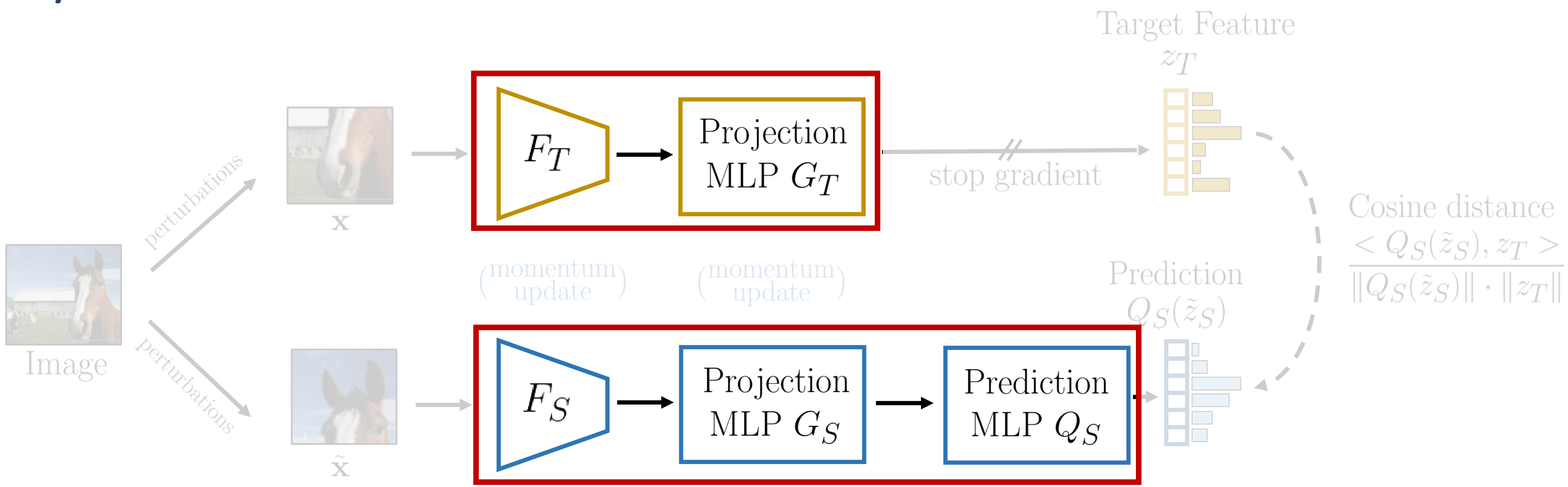
A mean teacher approach without any labels

- Offers stable but slowly evolving feature targets
- More efficient than using a fixed pre-training teacher that is updated only after the end of each training cycle (as BoWNet does)

Bootstrap Your Own Latent (BYOL)

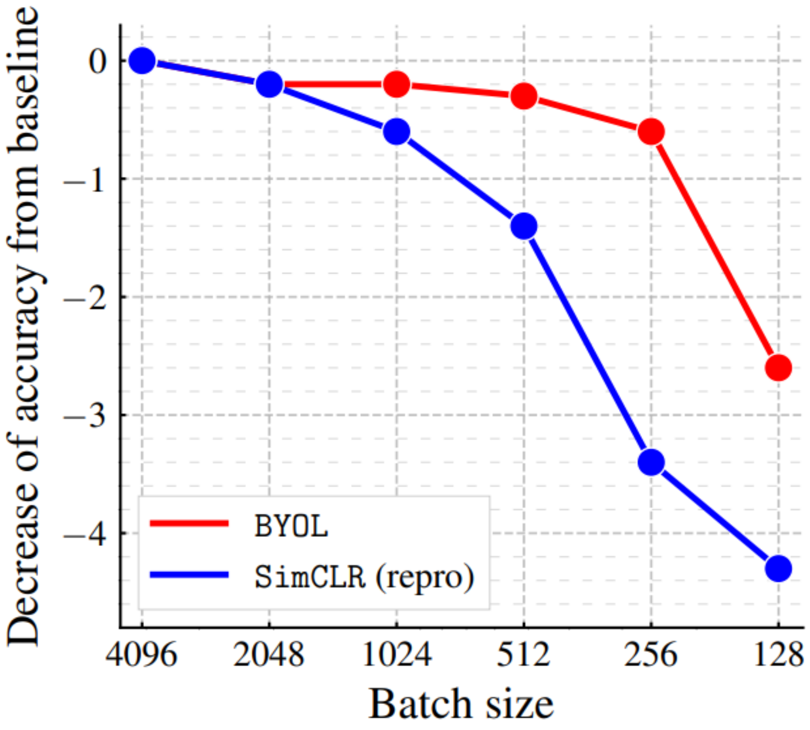


Asymmetric architecture

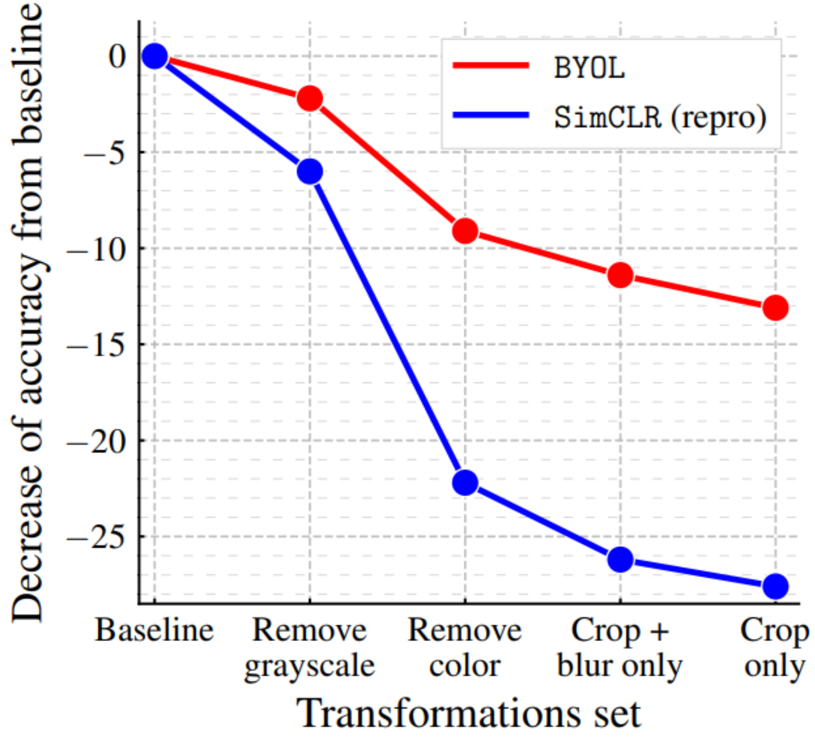


Asymmetric architecture: the student has an extra prediction MLP head

BYOL vs Contrastive methods (SimCLR)



(a) Impact of batch size

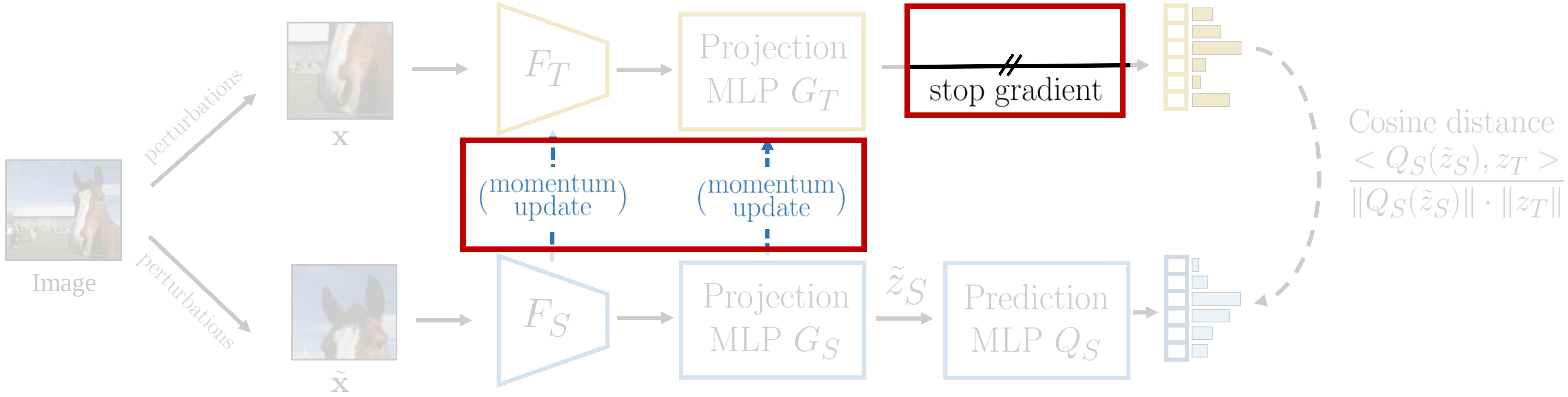


(b) Impact of progressively removing transformations

- BYOL **does not require negative examples** as the contrastive method SimCLR
- **More robust** to the choice of image augmentations and the batch-size
- Cropping is more important for BYOL and color jittering more important for SimCLR

Key question: Why it avoids feature collapse?

Why it avoids feature collapse?



The teacher parameter updates ARE NOT NECASSARILY in the direction of minimizing the loss, i.e., BYOL does not explicitly optimize the loss w.r.t. the teacher parameters.

"Bootstrap Your Own Latent: a new approach to self-supervised learning", NeurIPs 2020

Why it avoids feature collapse?

Batch Normalization (BN) in BYOL implicitly causes a form contrastive learning: collapse is avoided because all samples in the mini-batch cannot take on the same value after BN

- suggested in "Understanding self-supervised and contrastive learning with BYOL", Fetterman et al).

However, according to BYOL authors **"BYOL works even without batch statistics"**

- Either by better tuning the network initialization
- Or replacing BN with Group Normalization and Weight Standardization (GN + WS)

Table 2: top-1 accuracy with linear evaluation on ImageNet

BYOL variant	Vanilla BN	No BN	Modified init.	GN + WS
Uses batch statistics	Yes	No	No	No
Top-1 accuracy (%)	74.3	0.1	65.7	73.9

Why it avoids feature collapse?

Batch Normalization (BN) in BYOL implicitly causes a form contrastive learning: collapse is avoided because all samples in the mini-batch cannot take on the same value after BN

- suggested in "Understanding self-supervised and contrastive learning with BYOL", Fetterman et al).

However, according to BYOL authors **"BYOL works even without batch statistics"**

- Either by better tuning the network initialization
- Or replacing BN with Group Normalization and Weight Standardization (GN + WS)

Table 2: top-1 accuracy with linear evaluation on ImageNet

BYOL variant	Vanilla BN	No BN	Modified init.	GN + WS
Uses batch statistics	Yes	No	No	No
Top-1 accuracy (%)	74.3	0.1	65.7	73.9

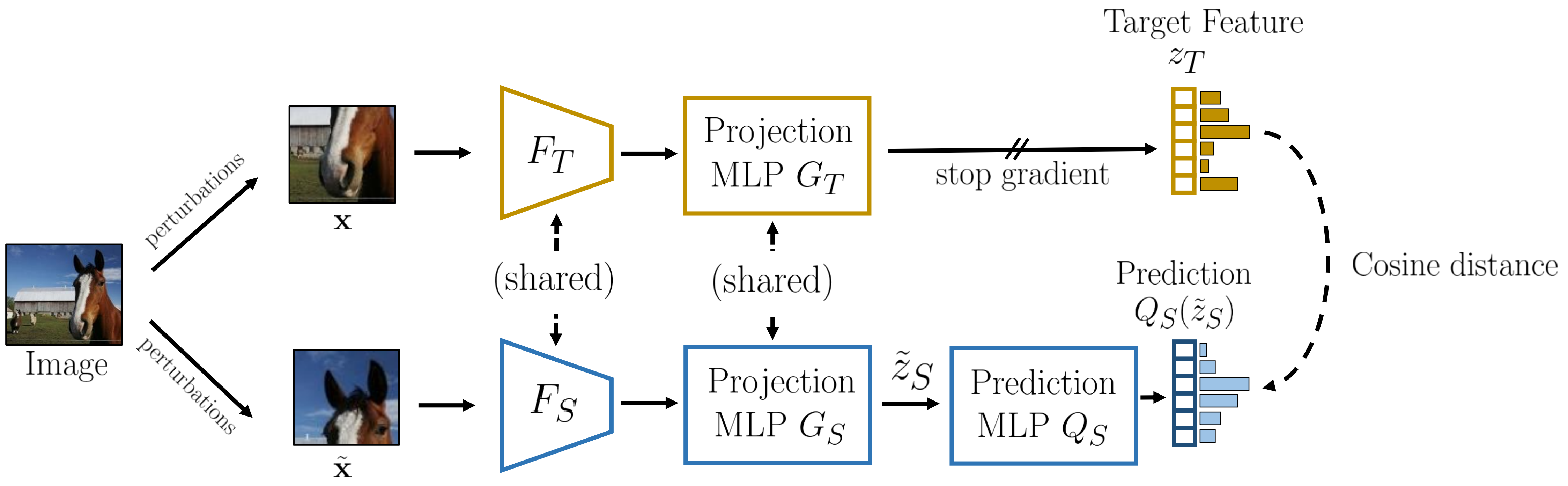
Why it avoids feature collapse?

(Hypothesis in BYOL) Thanks to student's prediction head and using EMA for the teacher. The **momentum teacher** allows to have a **near-optimal student predictor** that forces the student to encode more and more information within its projected features

Method	Predictor	Target network	Top-1
BYOL	✓	✓	**72.5**
—	✓		0.3
—		✓	0.2
—			0.1

ImageNet Top-1 linear classification accuracy. Removing the student Predictor or the Target network (using the student itself as teacher) leads to feature collapse.

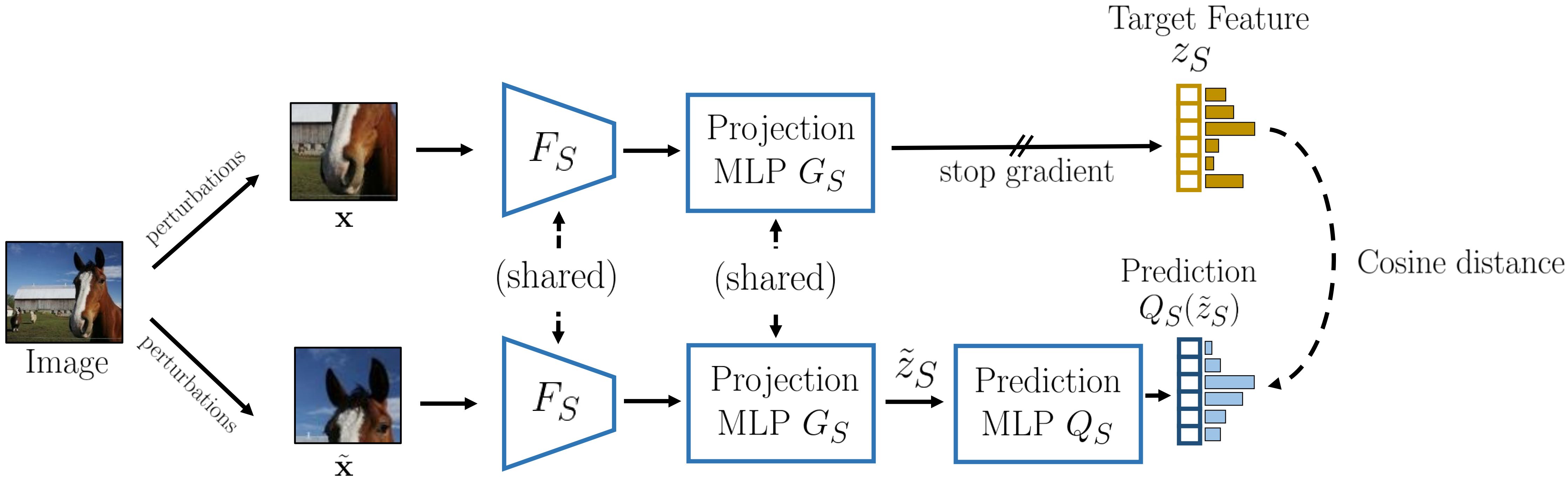
SimSiam



SimSiam: BYOL without the momentum teacher (the teacher is identical to the student)

"Exploring Simple Siamese Representation Learning", CVPR 2021

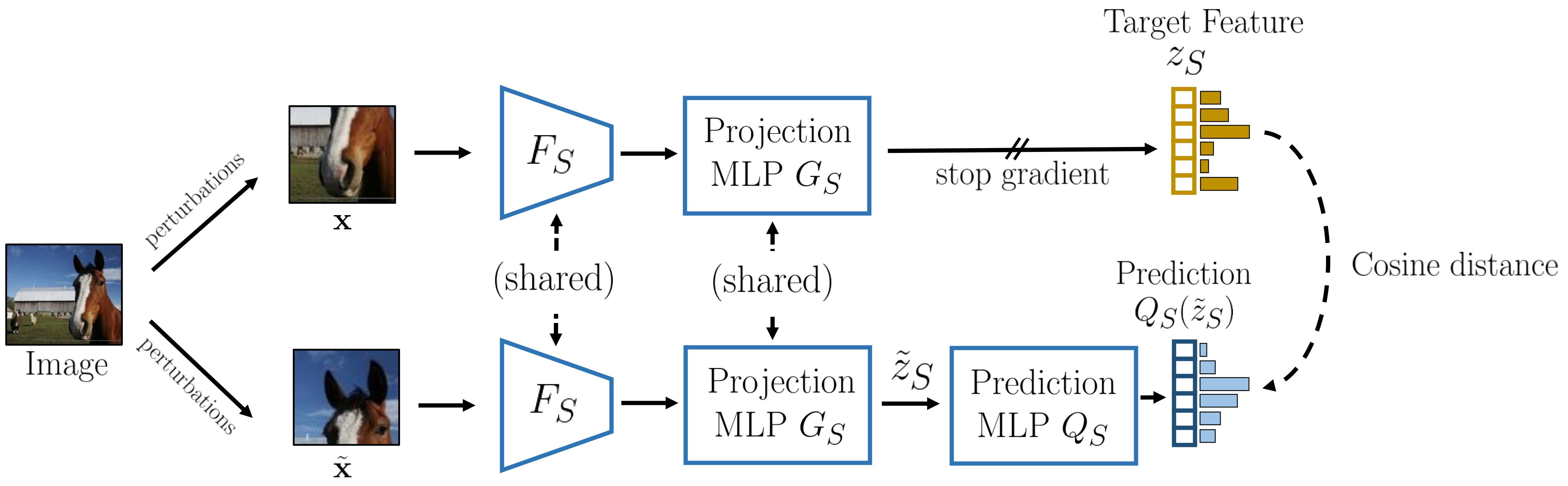
SimSiam



SimSiam: BYOL without the momentum teacher (the teacher is identical to the student)

"Exploring Simple Siamese Representation Learning", CVPR 2021

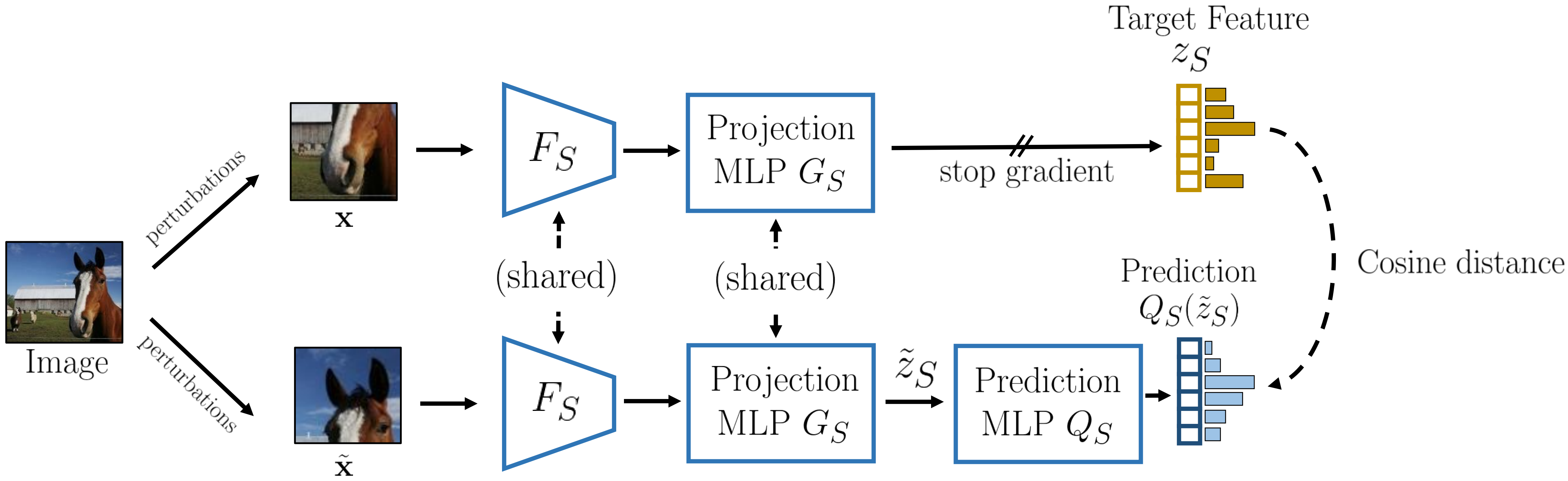
SimSiam



Momentum teacher: **improves performance but not necessary for avoiding feature collapse**

method	momentum encoder	100 ep	200 ep	400 ep	800 ep
BYOL	✓	66.5	**70.6**	**73.2**	**74.3**
SimSiam		**68.1**	70.0	70.8	71.3

SimSiam: When it avoids feature collapse?



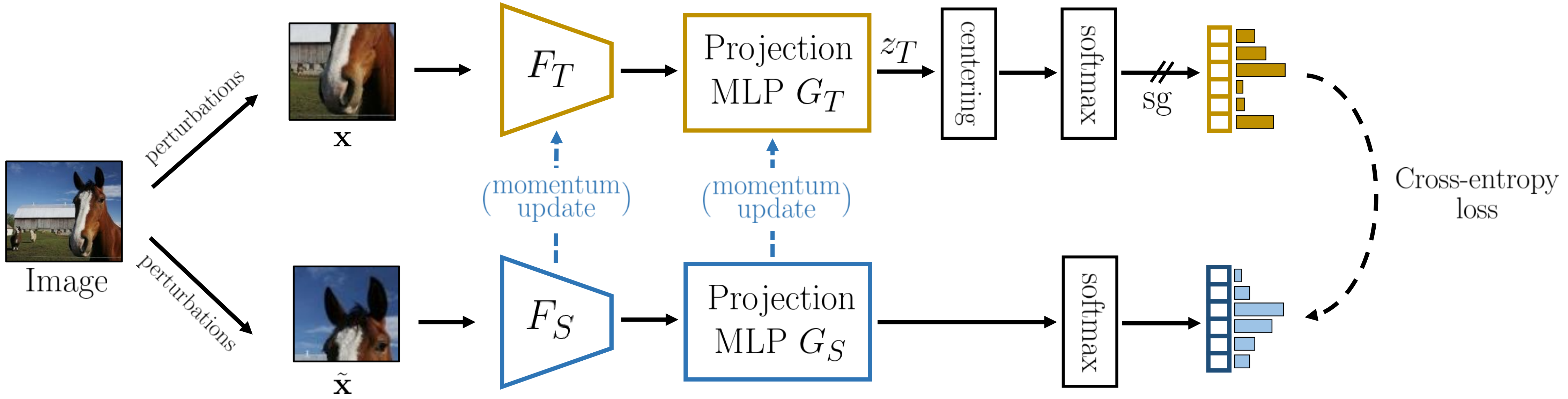
Without **stop-gradient** or the **predictor head** the network is trained to minimize the reconstruction loss for both image views at the same time, leading to constant features

	pred. MLP h	acc. (%)
baseline	lr with cosine decay	67.7
(a)	no pred. MLP	0.1

Table 1. **Effect of prediction MLP**

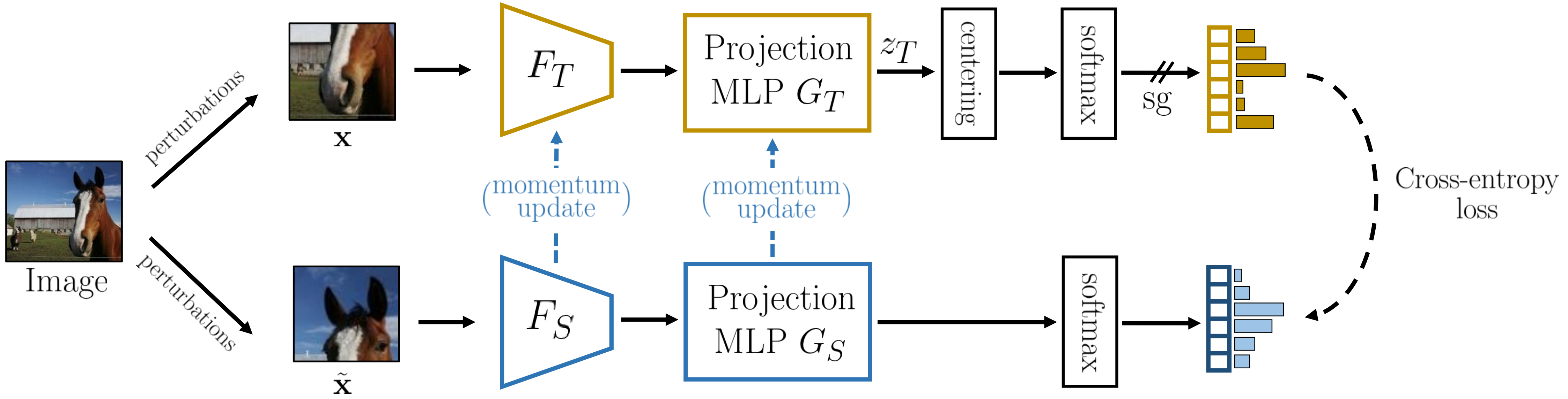
	acc. (%)
w/ stop-grad	67.7±0.1
w/o stop-grad	0.1

DINO



"Emerging Properties in Self-Supervised Vision Transformers", arXiv 2021

DINO



No prediction head - post-processing of teacher outputs to avoid feature collapse:

- **Centering by subtracting the mean feature:** prevents collapsing to constant 1-hot targets
- **Sharpening by using low softmax temperature:** prevents collapsing to a uniform target vector

"Emerging Properties in Self-Supervised Vision Transformers", arXiv 2021

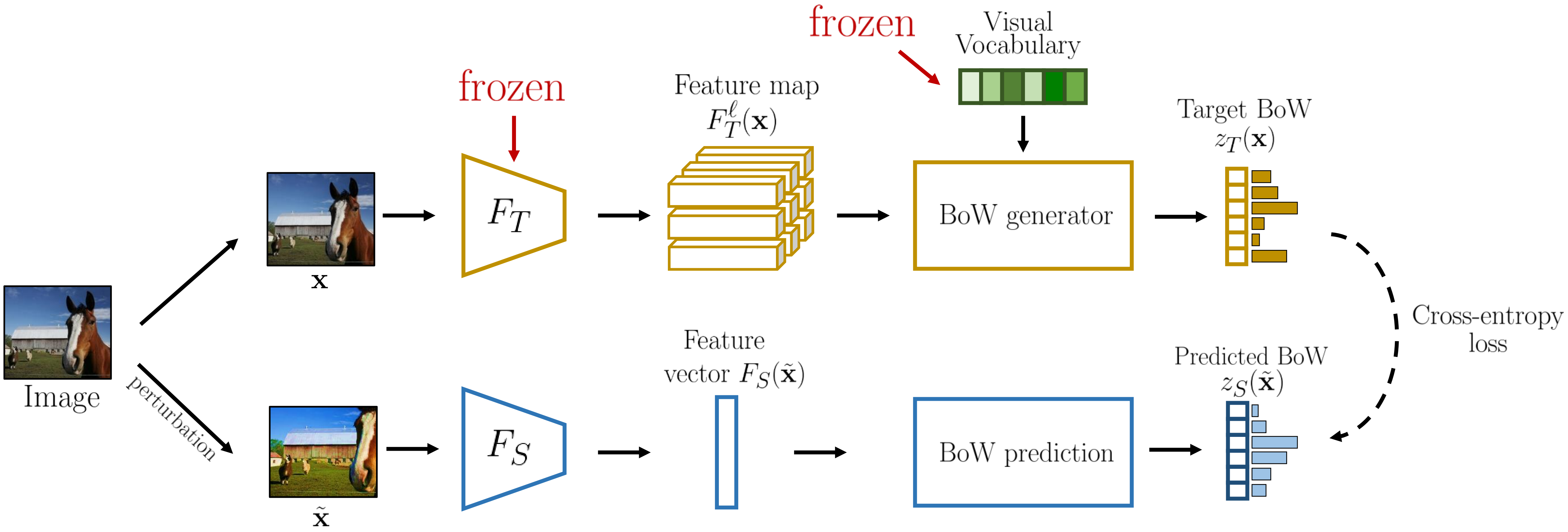
DINO

Method	Mom.	Loss	Pred.	k-NN	Lin.
DINO	✓	CE	✗	72.8	76.1
	✗	CE	✗	0.1	0.1
	✓	MSE	✗	52.6	62.4
	✓	CE	✓	71.8	75.6
BYOL	✓	MSE	✓	66.6	71.4

- **Loss:** Cross-Entropy (CE) instead of Mean-Squared Error (MSE)
- **Momentum teacher:** avoid collapsing
- **Better without predictor**

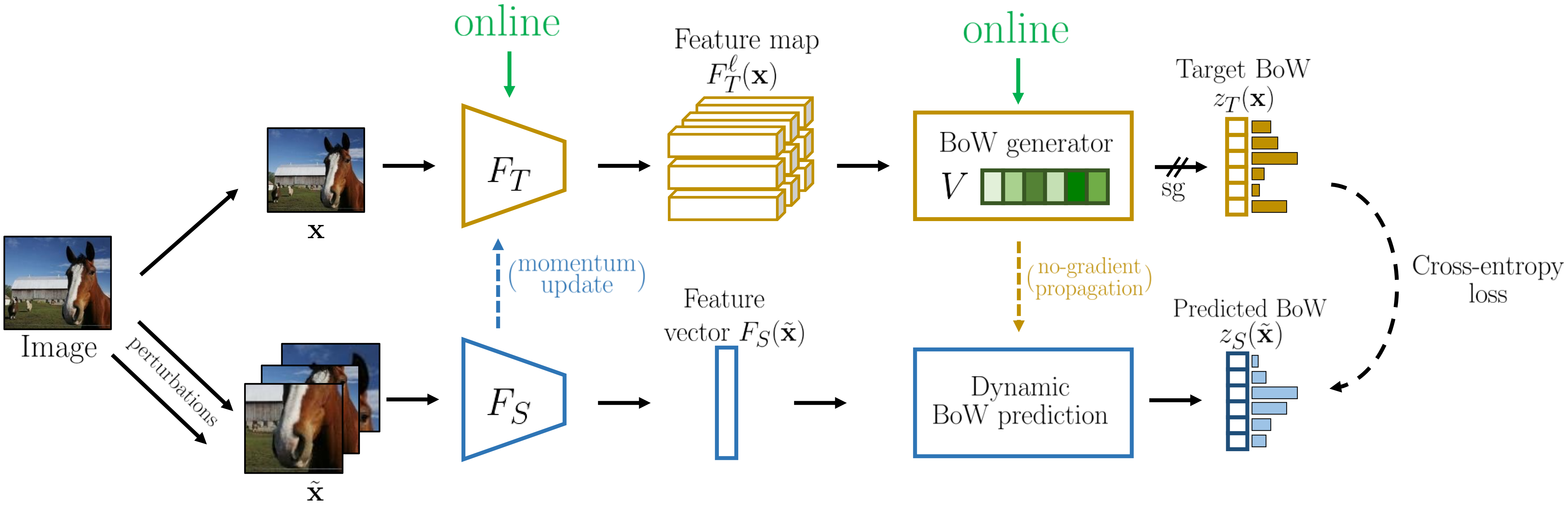
“Emerging Properties in Self-Supervised Vision Transformers”, arXiv 2021

BoWNet



- Enforcing the learning of **perturbation invariant** and **context-aware features**
- Frozen teacher ➔ suboptimal supervisory signal for the student training

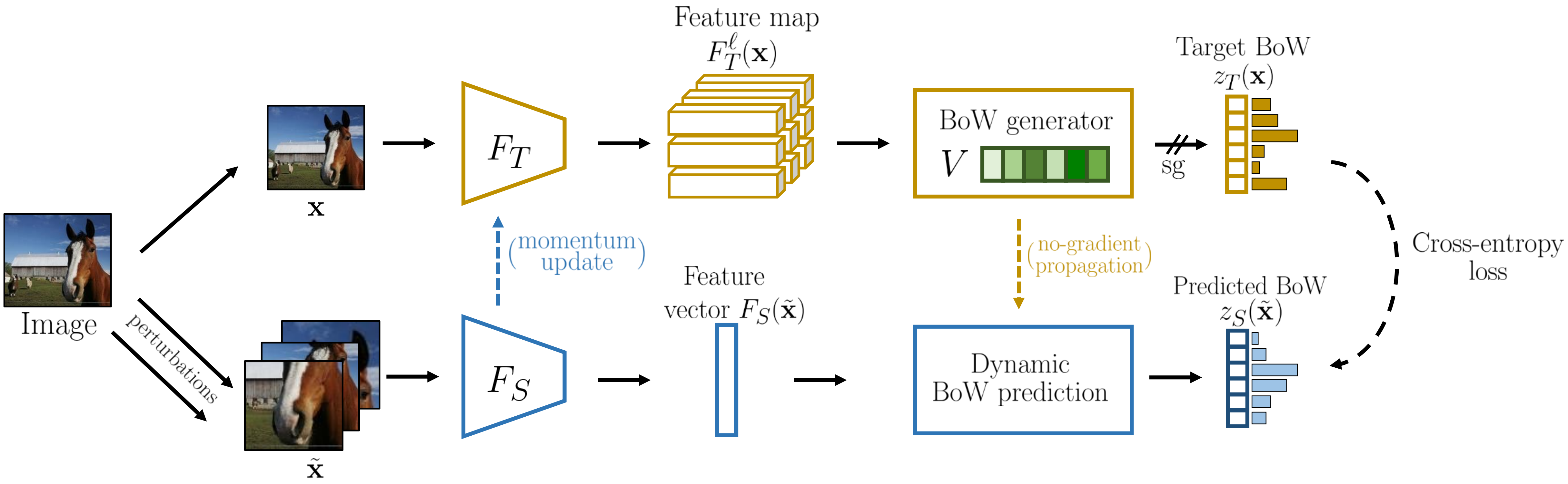
OBoW: an improved BoW-based self-supervised approach



- **Fully online bag-of-visual-words generation**
- Representation learning based on **enhanced contextual reasoning**

"OBoW: Online Bag-of-Visual-Words Generation for Self-supervised Learning ", Gidaris et al, CVPR 2021

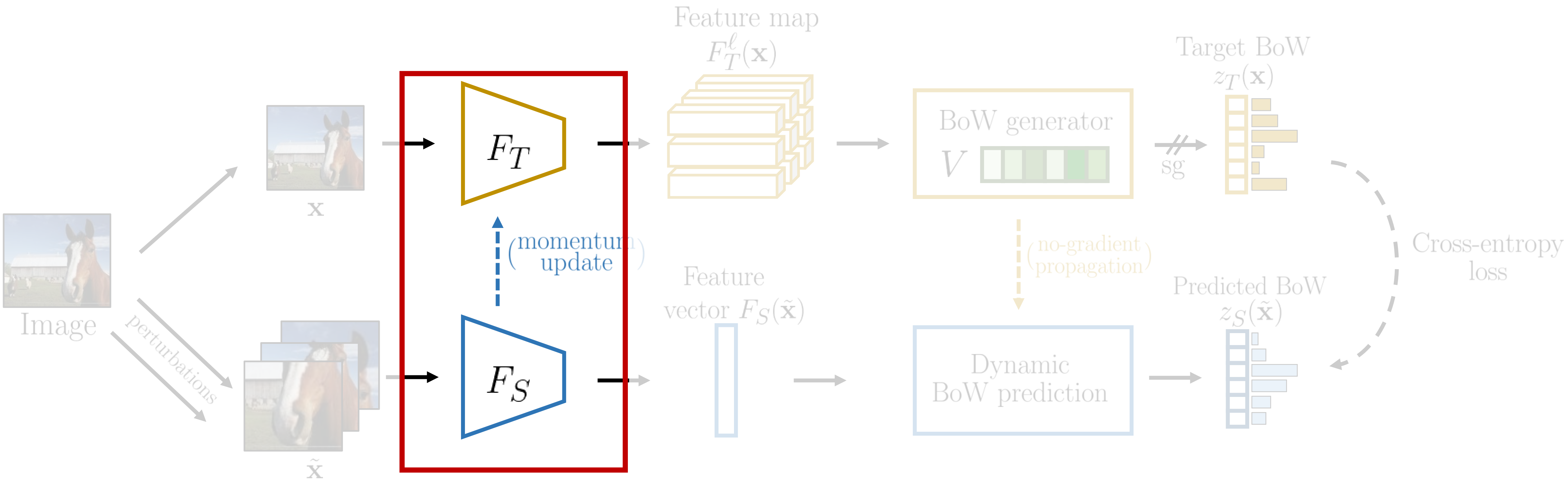
Fully online BoW-based learning



Teacher components: **(1) network parameters, (2) visual-words vocabulary**

- **BoWNet:** offline pre-trained; fixed during student training
- **OBoW:** Both are **online updated together with the student**

Online updating of the teacher network parameters



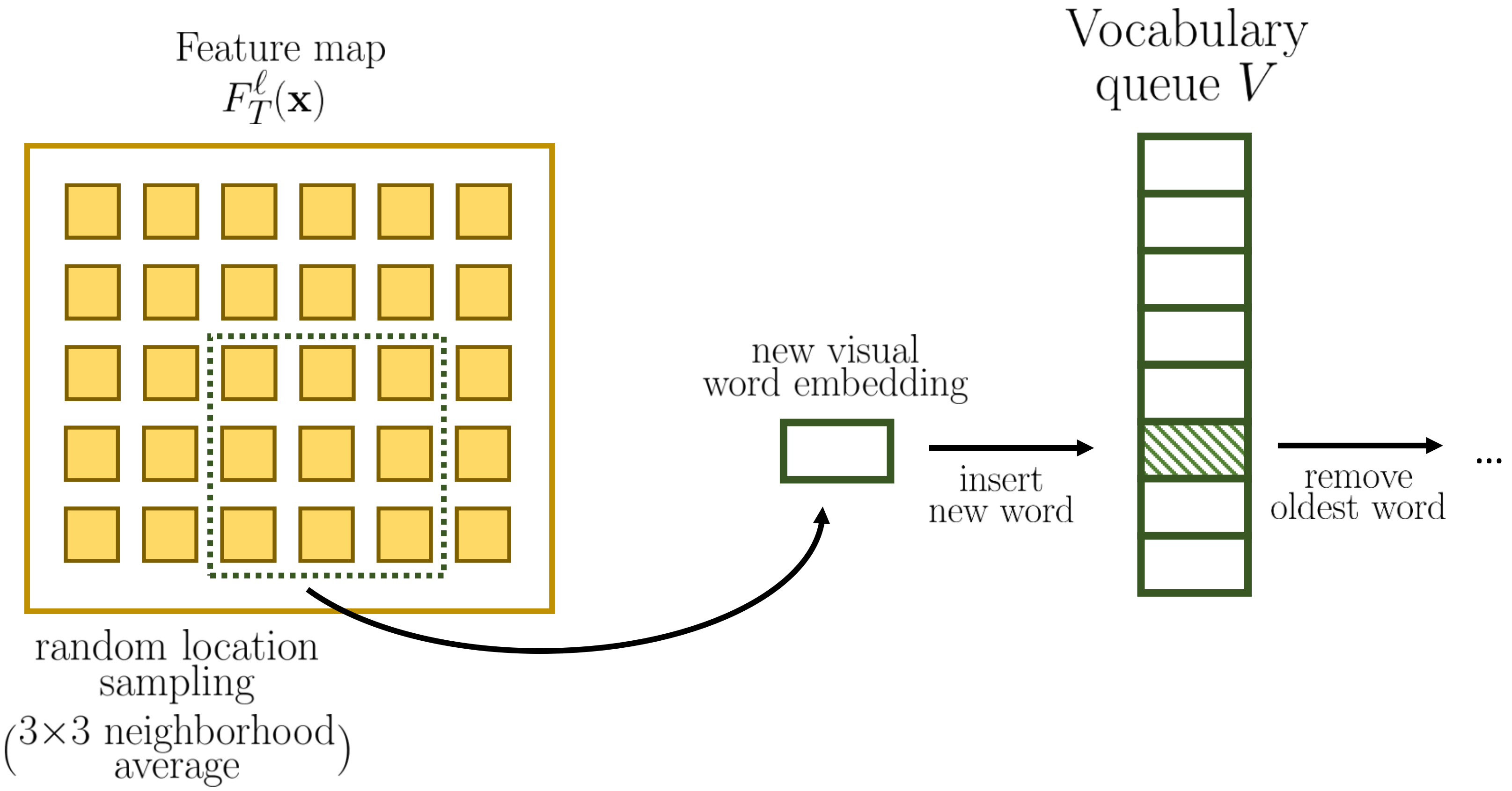
Exponential moving average update: after each SGD training step t

$$\theta_{\mathrm{T}}^{(t)} \leftarrow \alpha \cdot \theta_{\mathrm{T}}^{(t-1)} + (1-\alpha) \cdot \theta_{\mathrm{S}}^{(t)}$$

θ_{T} : teacher parameters

θ_{S} : student parameters

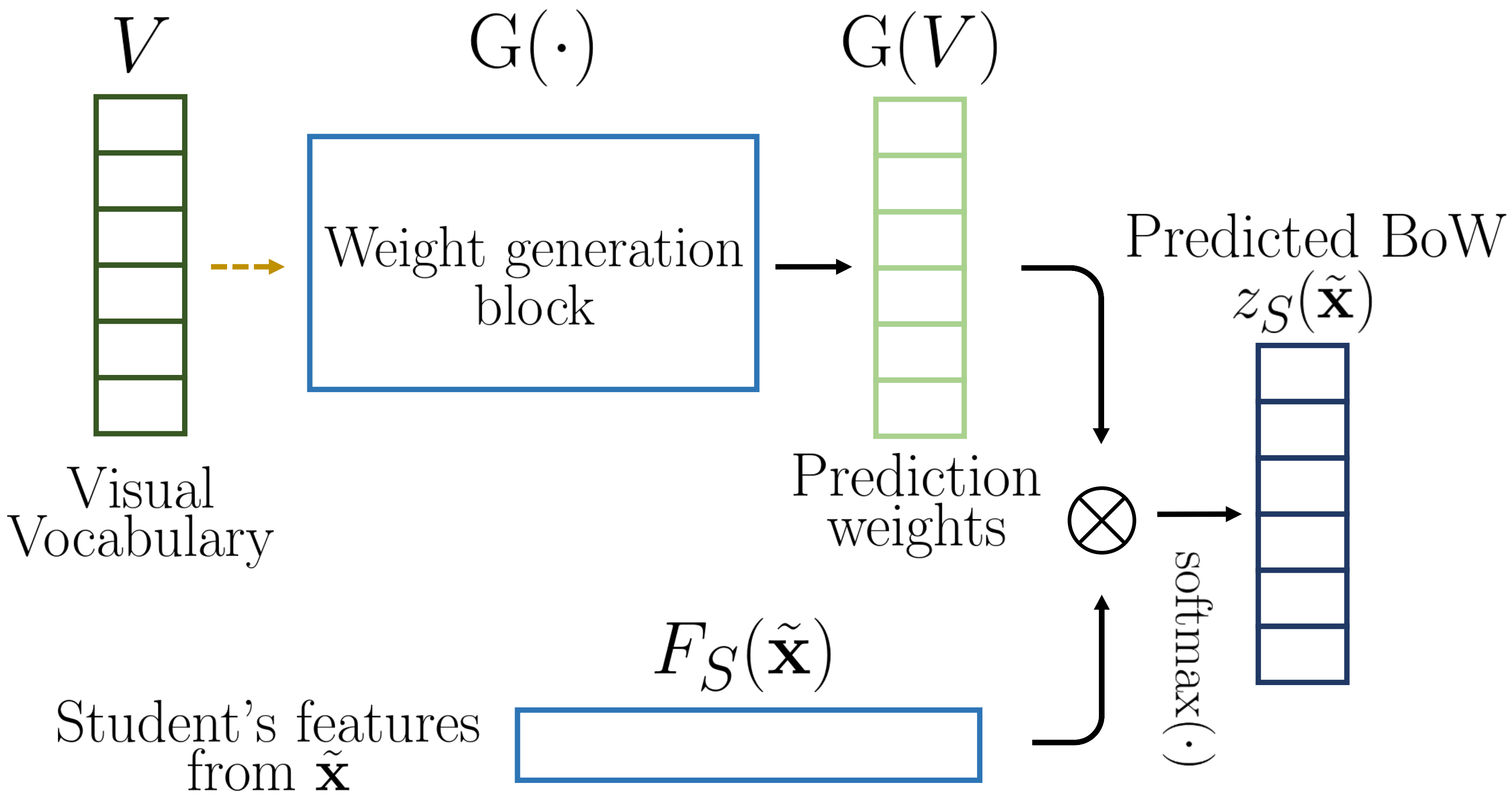
Queue-based vocabulary from randomly sampled local features



Online updating of queue-based vocabulary. At each training step:

- Randomly select one feature vector per training image as visual word
- Add it to a K-sized queue while removing its oldest item/word

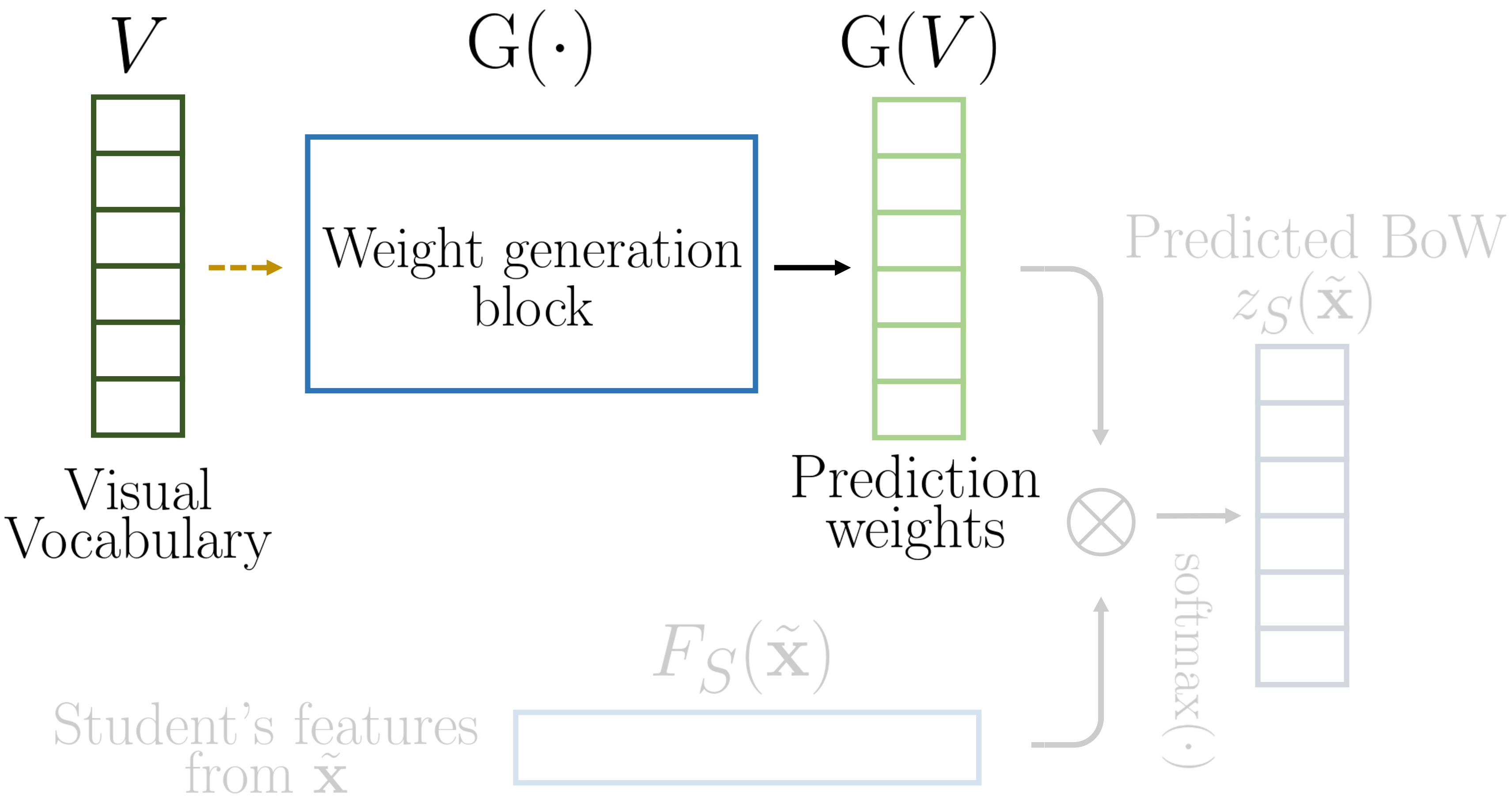
Dynamic bag-of-visual-word prediction



BoWNet: ~~fixed linear prediction layer for BoW prediction~~

OBoW: constantly updated vocabulary ➔
requires dynamic generation of prediction weights

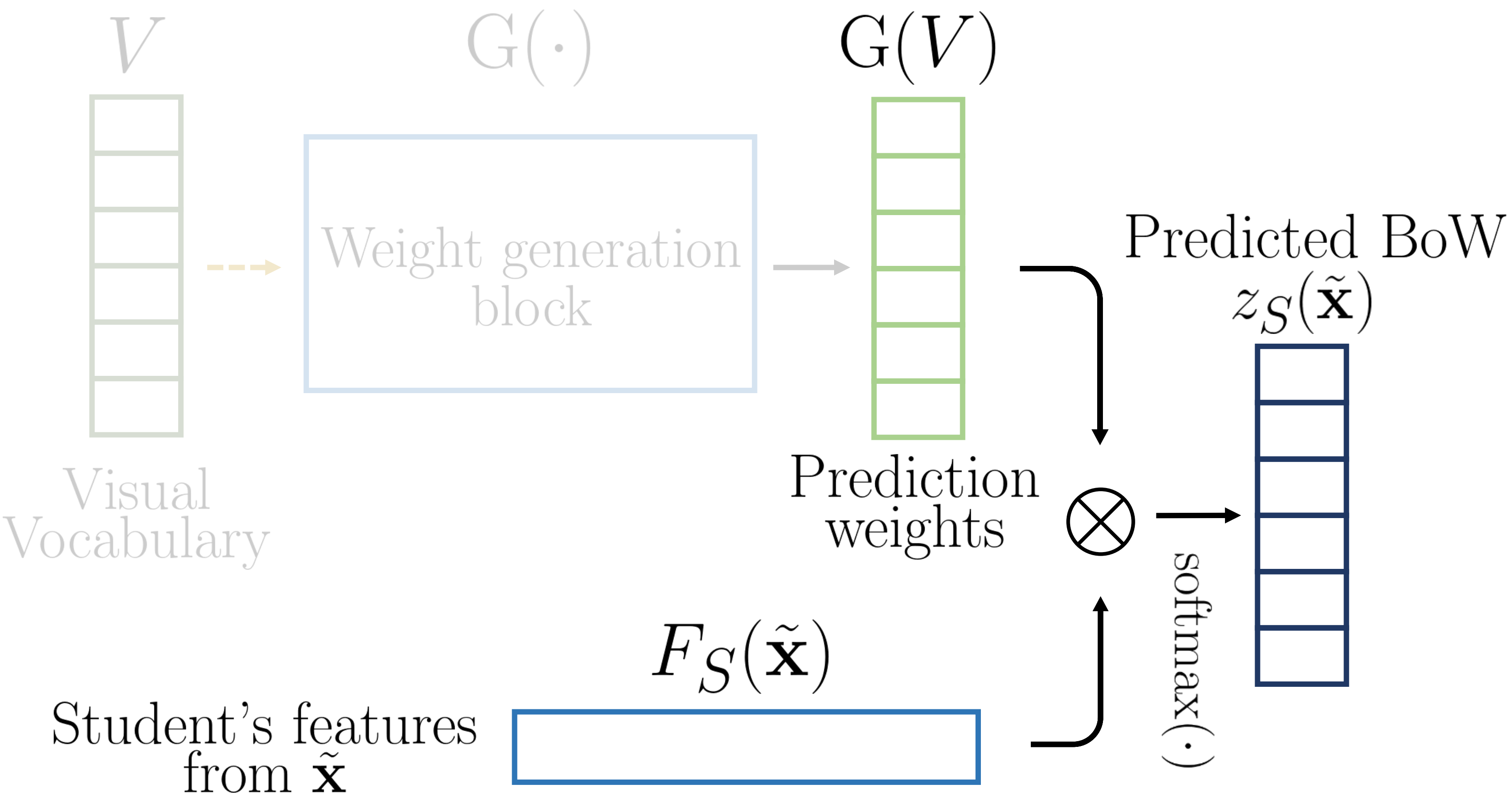
Dynamic bag-of-visual-word prediction



BoWNet: ~~fixed linear prediction layer for BoW prediction~~

OBoW: constantly updated vocabulary ➔ requires dynamic generation of prediction weights

Dynamic bag-of-visual-word prediction



BoWNet: ~~fixed linear prediction layer for BoW prediction~~

OBoW: constantly updated vocabulary ➔
requires dynamic generation of prediction weights

Representation learning based on enhanced contextual reasoning

1. Predicting BoW from small crops of the original image

Input to the teacher

Central 224x224 crop

Input to the student

160x160 crops

96x96 patches

2. **Multi-scale BoW reconstruction targets** (conv5 and conv4 layers of ResNet)

- Also using the conv4 further promotes the learning of context-aware features.

OBoW: Avoiding feature collapse

Since the BoW targets are computed using a constantly updated set of randomly sampled local features, **OBoW by construction does not suffer from feature collapsing**, thus making it **robust to the momentum coefficient** used for the momentum teacher updates.

α	lr	Few-shot 1-shot	1-shot	Linear
$0.99 \rightarrow 1$	0.05	**42.11**	**62.44**	45.86
0.999	0.05	40.87	61.41	45.76
0.99	0.05	41.19	61.65	**46.25**
0.9	0.05	40.79	60.92	44.89
0.5	0.03	39.52	60.18	43.82
0.0	0.01	33.80	55.02	39.90

Table 2: Influence of the momentum coefficient α used for the teacher updates.

Agenda

- Input reconstruction
- Teacher-student feature reconstruction
- Wrap up evaluation

Evaluating ResNet50 self-supervised pre-trained networks

Method	Epochs	Batch	Linear Classification ImageNet	Places205	VOC07
Supervised	100	256	76.5	53.2	87.5
Feature prediction methods					
BoWNet	325	256	62.1	51.1	79.3
OBoW	200	256	73.8	**56.8**	**89.3**
BYOL	1000	4096	74.3	54.0	86.6
SimSiam	1000	256	71.3	-	-
Barlow Twins	1000	2048	73.2	54.1	86.2
DINO	800	1024	**75.3**	-	-
Contrastive methods					
MoCo v2	800	256	71.1	52.9	87.1
SimCLR	1000	4096	69.3	53.3	86.4
Clystering-style methods					
SwAV	800	4096	**75.3**	56.5	88.9

Evaluating ResNet50 self-supervised pre-trained networks

Method	Epochs	Batch	Semi-supervised learning 1% Labels	Semi-supervised learning 10% Labels
Supervised	100	256	48.4	80.4
Feature prediction methods				
BoWNet	325	256	-	-
OBoW	200	256	**82.9**	**90.7**
BYOL	1000	4096	78.4	89.0
SimSiam	1000	256	-	-
Barlow Twins	1000	2048	79.2	89.3
DINO	800	1024	-	-
Contrastive methods				
MoCo v2	800	256	-	-
SimCLR	1000	4096	75.5	87.8
Clystering-style methods				
SwAV	800	4096	78.5	89.9

Evaluating ResNet50 self-supervised pre-trained networks

Method	Epochs	Batch	VOC Detection AP^{50}	AP^{75}	AP^{all}
Supervised	100	256	81.3	58.8	53.5
Feature prediction methods					
BoWNet	325	256	81.3	61.1	55.8
OBoW	200	256	**82.9**	**64.8**	**57.9**
BYOL	1000	4096	81.4	55.3	61.1
SimSiam	1000	256	82.4	57.0	63.9
Barlow Twins	1000	2048	56.8	82.6	63.4
DINO	800	1024	-	-	-
Contrastive methods					
MoCo v2	800	256	82.5	64.0	57.4
SimCLR	1000	4096	-	-	-
Clystering-style methods					
SwAV	800	4096	82.6	62.7	56.1

Conclusions

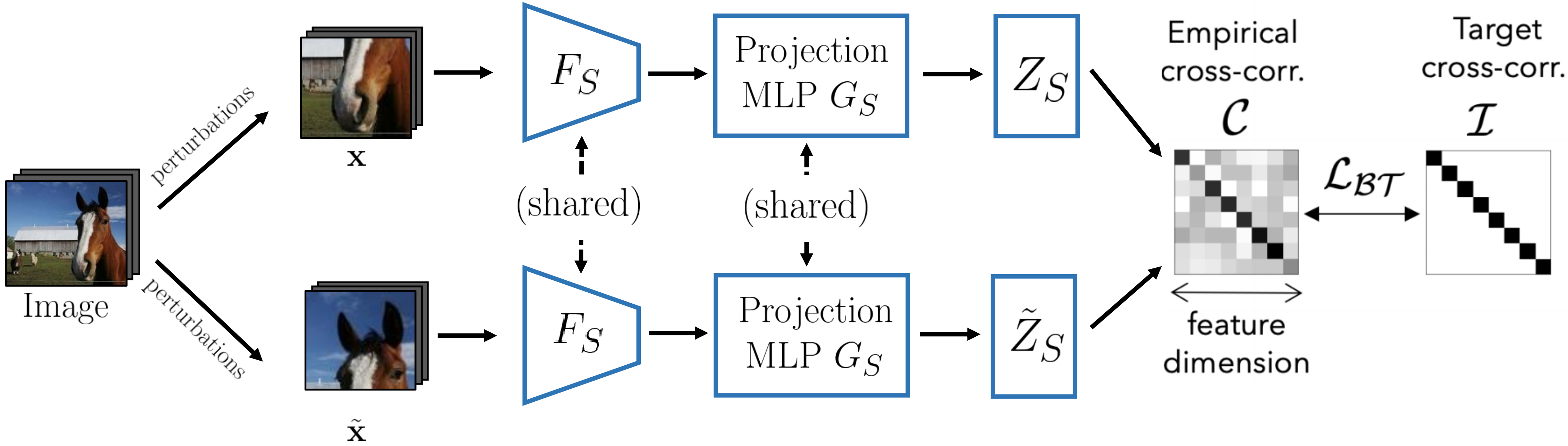
- Feature “reconstruction” self-supervised methods are gaining increased attention

- Manage to learn SOTA self-supervised representations without requiring negatives
  - Surpassing even supervised representations

- However, it’s not entirely clear why they avoid feature collapse

- Recent trends: mid-way between contrastive and feature reconstruction
  - “Whitening for self-supervised representation learning”, arXiv 2020
  - “Barlow Twins: self-supervised learning via redundancy reduction”, ICML 2021
  - “VICReg: Variance-Invariance-Covariance Regularization for self-supervised learning”, arXiv 2021
  - …

Barlow Twins



$$\mathcal{C}_{ij} \triangleq \frac{\sum_b Z_S[b,i] \cdot \tilde{Z}_S[b,j]}{\sqrt{\sum_b Z_S[b,i]^2}\sqrt{\sum_b \tilde{Z}_S[b,j]^2}}$$

$$\mathcal{L}_{\mathcal{BT}} \triangleq \underbrace{\sum_i (1-\mathcal{C}_{ii})^2}_{\text{invariance term}} + \lambda \underbrace{\sum_i \sum_{j \neq i} {\mathcal{C}_{ij}}^2}_{\text{redundancy reduction term}}$$

Computes the cross-correlation matrix between the outputs of two identical networks fed with distorted versions of a sample, makes it as close to identity matrix as possible.

"Barlow Twins: Self-Supervised Learning via Redundancy Reduction", ICML 2021

Conclusions

- Feature “reconstruction” self-supervised methods are gaining increased attention
- Manage to learn SOTA self-supervised representations without requiring negatives
  - Surpassing even supervised representations
- However, it’s not entirely clear why they avoid feature collapse
- Recent trends:
  - “Barlow Twins: self-supervised learning via redundancy reduction”, ICML 2021
  - “Whitening for self-supervised representation learning”, arXiv 2020
  - “VICReg: Variance-Invariance-Covariance Regularization for self-supervised learning”, arXiv 2021
  - …
- **Clustering-style methods can be seen as teacher-student approaches. See next talk!!!**

The end