Semi-Supervised Learning with Contrastive Predictive Coding

Journal Club

Brian Pollack

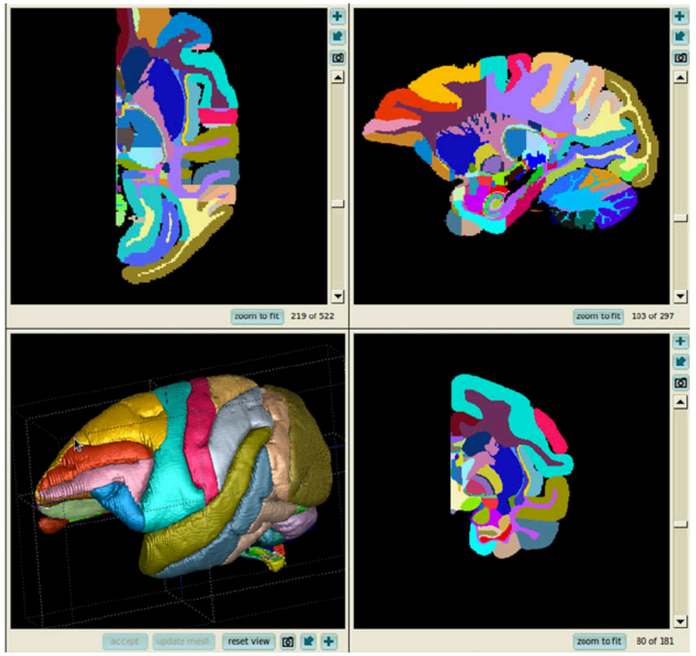
7/23/19

Semi-Supervised Learning

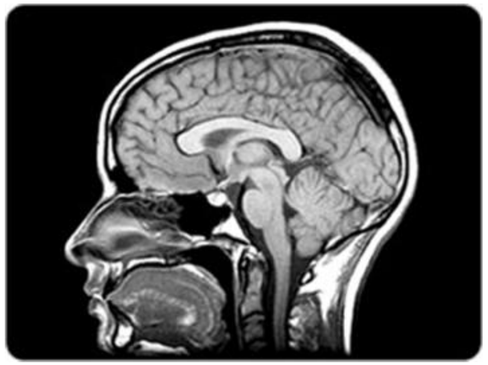
- ★ **Combing labeled and unlabeled training data to improve performance.**
  - Labeled data is rarer than unlabeled:
    - Funding is rare for labeled sets (c 2016).
    - Manpower is limited.
    - HIPPA restrictions.
    - Medical datasets are narrow in scope.

    https://ieeexplore.ieee.org/abstract/document/7463094
  - Much easier to obtained unlabeled (or weakly labeled) data.
    - Unlabeled data still contains rich information.

Labeled

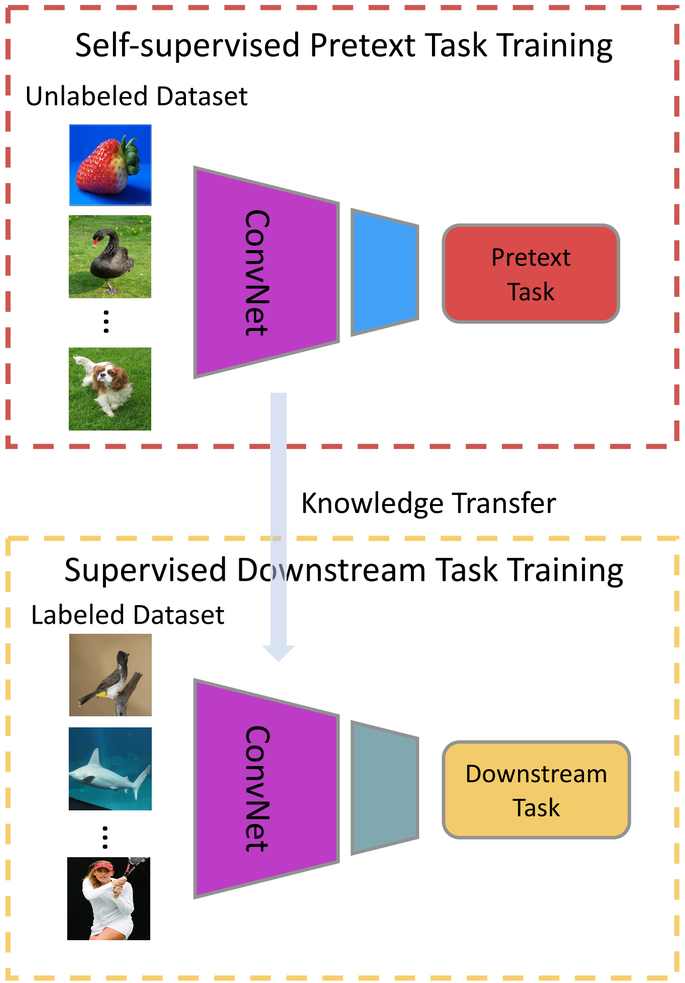


Unlabeled



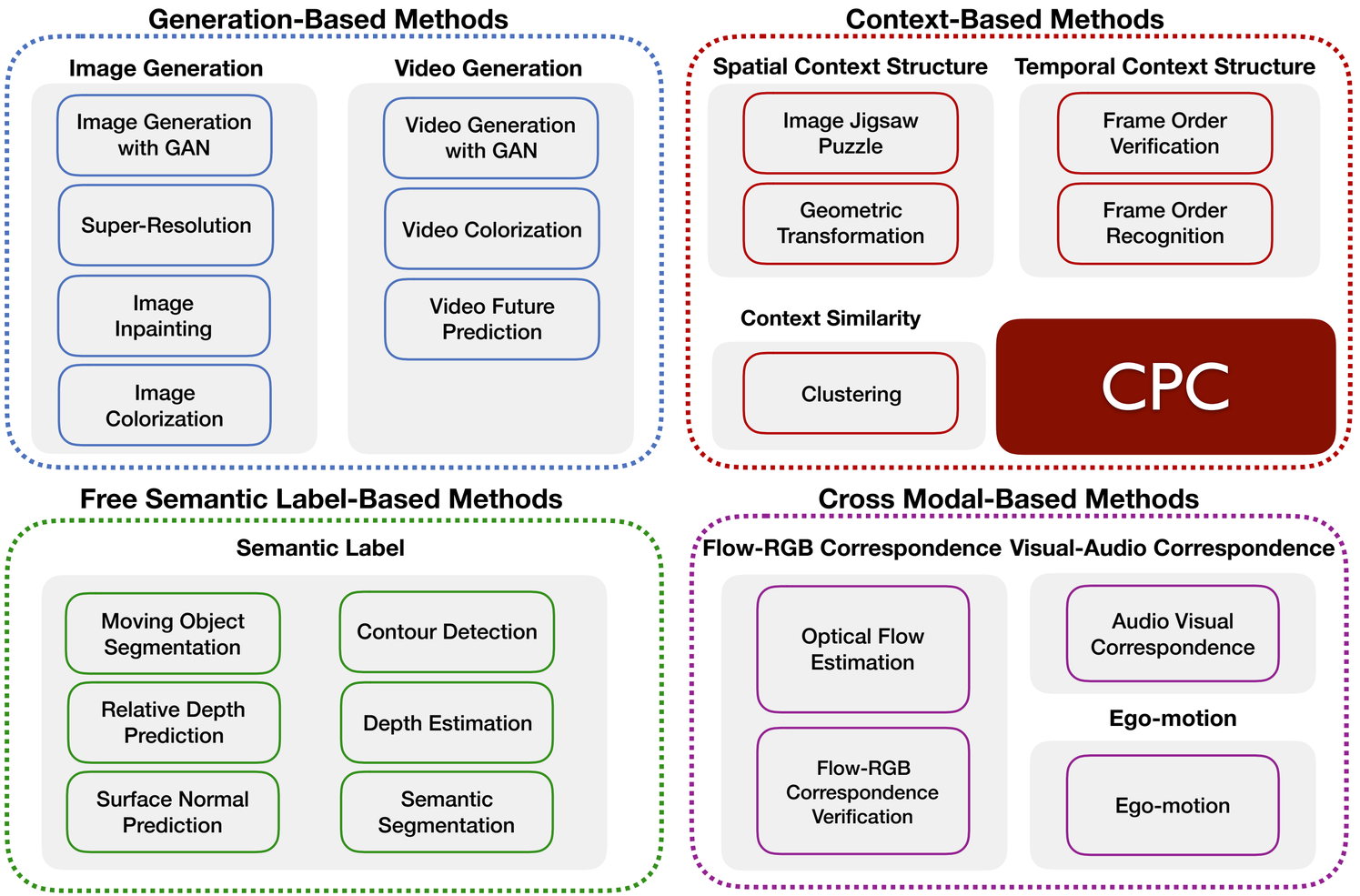
Self-Supervised Learning

- **Provide “pretext task” to unlabeled data and train self-supervised model.**
- **Use transfer learning to apply self-supervised model to labeled data training task.**



Self-supervised Visual Feature Learning with Deep Neural Networks: A Survey
https://arxiv.org/pdf/1902.06162.pdf

Pretext Tasks



Contrastive Predictive Coding

Self-supervised Visual Feature Learning with Deep Neural Networks: A Survey
https://arxiv.org/pdf/1902.06162.pdf

Pretext Tasks

Generation-Based Methods

Image Generation

Inpainting

with GAN

Video Generation

Video Generation with GAN

Prediction

Depth Estimation

Surface Normal Prediction

Semantic Segmentation

Context-Based Methods

Spatial Context Structure

Image Jigsaw Puzzle

Geometric Transformation

Temporal Context Structure

Frame Order Verification

Frame Order Recognition

Context Similarity

Clustering

CPC

Contrastive Predictive Coding

Cross Modal-Based Methods

Flow-RGB Correspondence

Optical Flow Estimation

Flow-RGB Correspondence Verification

Visual-Audio Correspondence

Audio Visual Correspondence

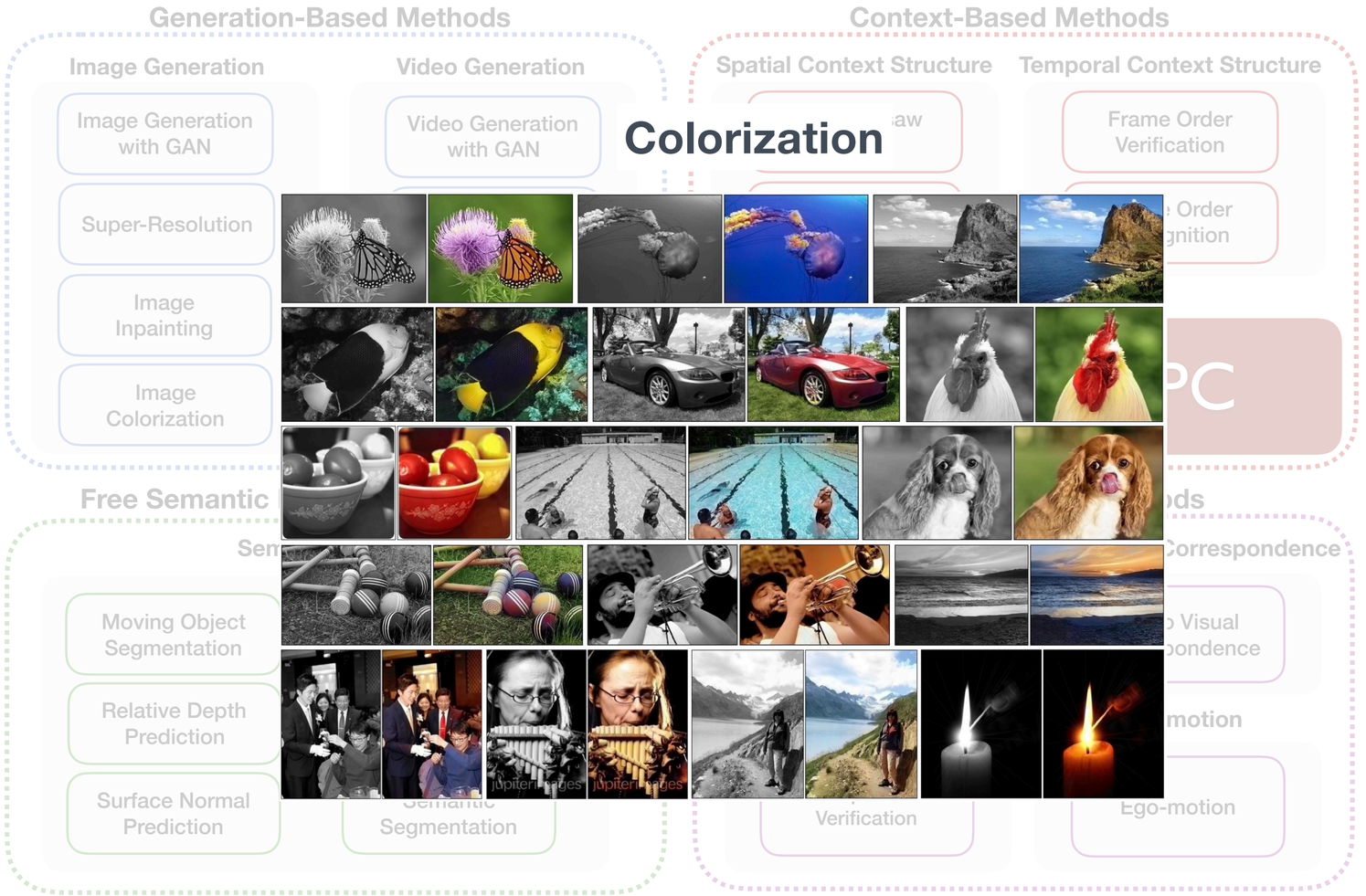
Ego-motion

Ego-motion

Self-supervised Visual Feature Learning with Deep Neural Networks: A Survey

https://arxiv.org/pdf/1902.06162.pdf

Pretext Tasks



Self-supervised Visual Feature Learning with Deep Neural Networks: A Survey
https://arxiv.org/pdf/1902.06162.pdf

Pretext Tasks

Generation-Based Methods

Image Generation

Image Generation with GAN

Video Generation

Video Generation with GAN

Context-Based Methods

Spatial Context Structure

Image Jigsaw Puzzle

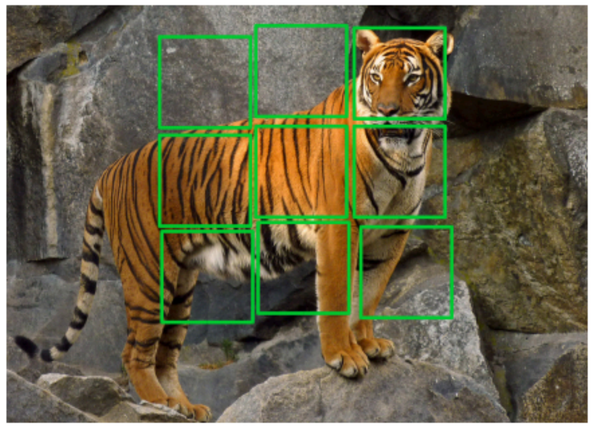
Geometric

Temporal Context Structure

Frame Order Verification

Frame Order

Jigsaw



(a)

(b)

(c)

ontrastive

redictive

Coding

Prediction

Depth Estimation

Surface Normal Prediction

Semantic Segmentation

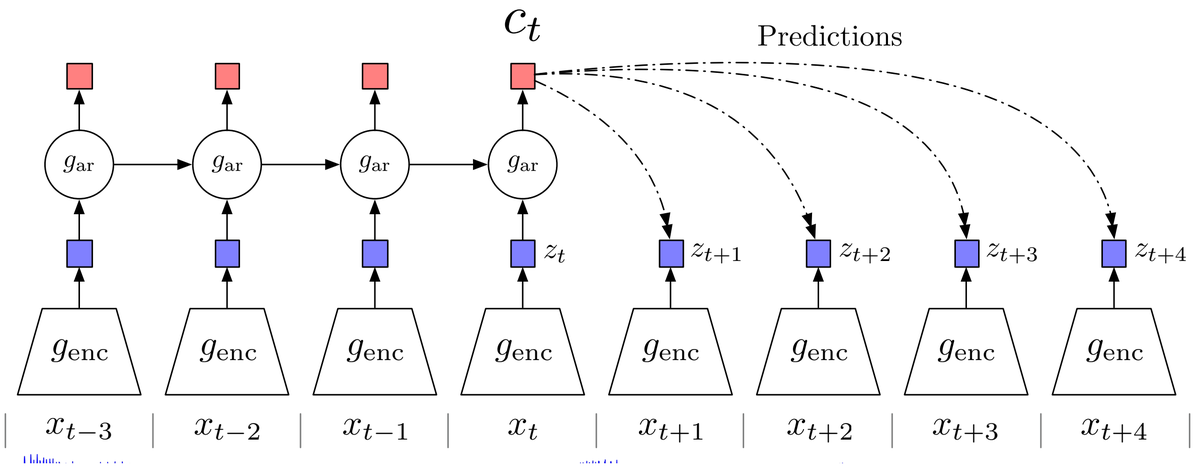
Flow-RGB Correspondence Verification

Ego-motion

Self-supervised Visual Feature Learning with Deep Neural Networks: A Survey

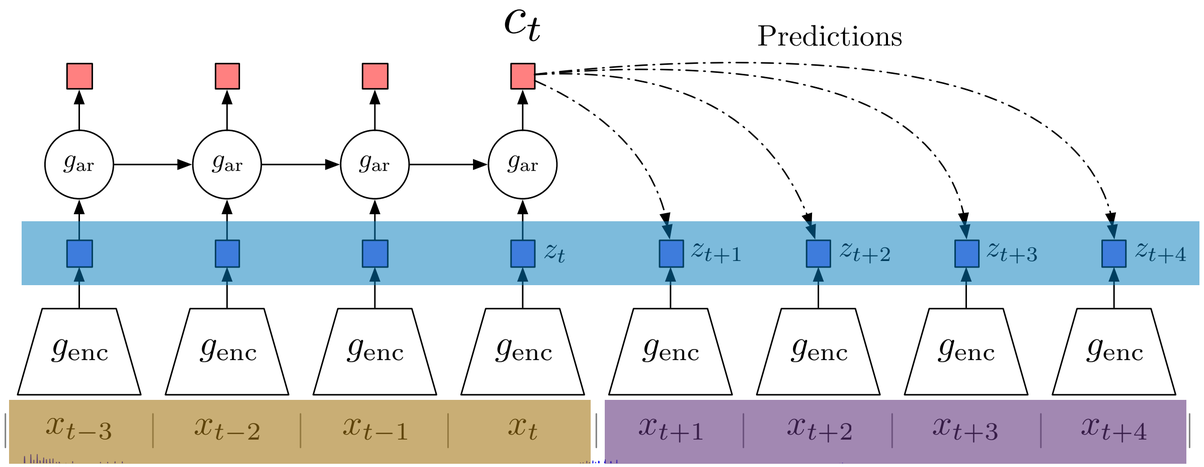
https://arxiv.org/pdf/1902.06162.pdf

Contrastive Predictive Coding



Representation Learning with Contrastive Predictive Coding
https://arxiv.org/pdf/1807.03748.pdf

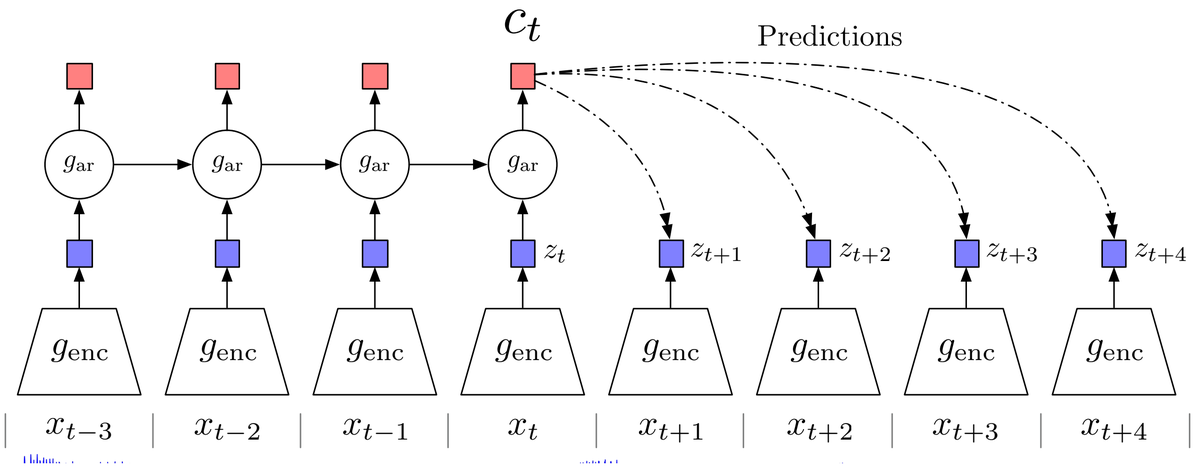
Contrastive Predictive Coding



Predictive Coding: Using features from **upstream observations (x_{t-i})** to predict features in **downstream observations (x_{t+j})**. To avoid unnecessary modeling of low-level features, **compact latent vectors (z_t)** are used instead of raw observations.

Representation Learning with Contrastive Predictive Coding
https://arxiv.org/pdf/1807.03748.pdf

Contrastive Predictive Coding

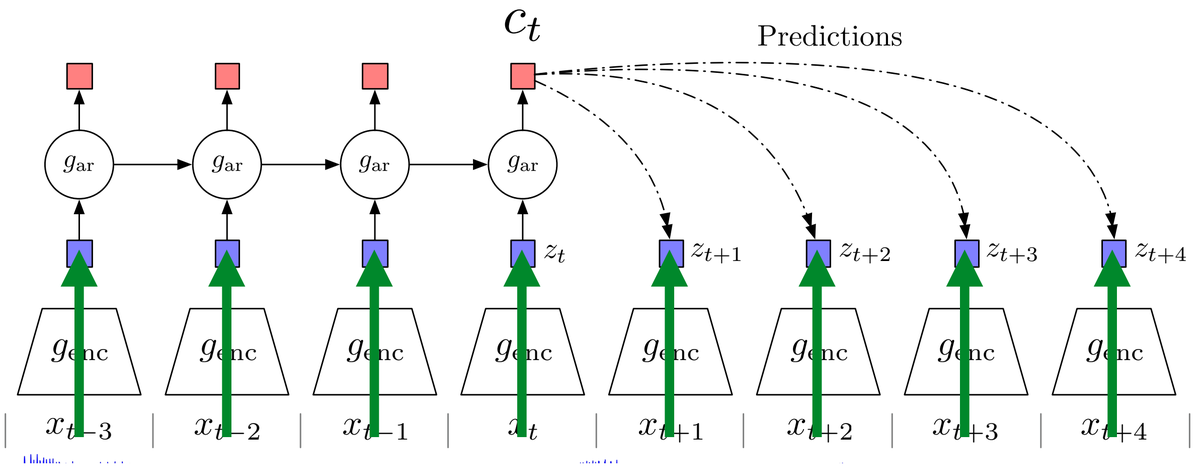


Procedure:

Predictive Coding: Using features from **upstream observations (x_{t-i})** to predict features in **downstream observations (x_{t+j})**. To avoid unnecessary modeling of low-level features, **compact latent vectors (z_t)** are used instead of raw observations.

Representation Learning with Contrastive Predictive Coding
https://arxiv.org/pdf/1807.03748.pdf

Contrastive Predictive Coding



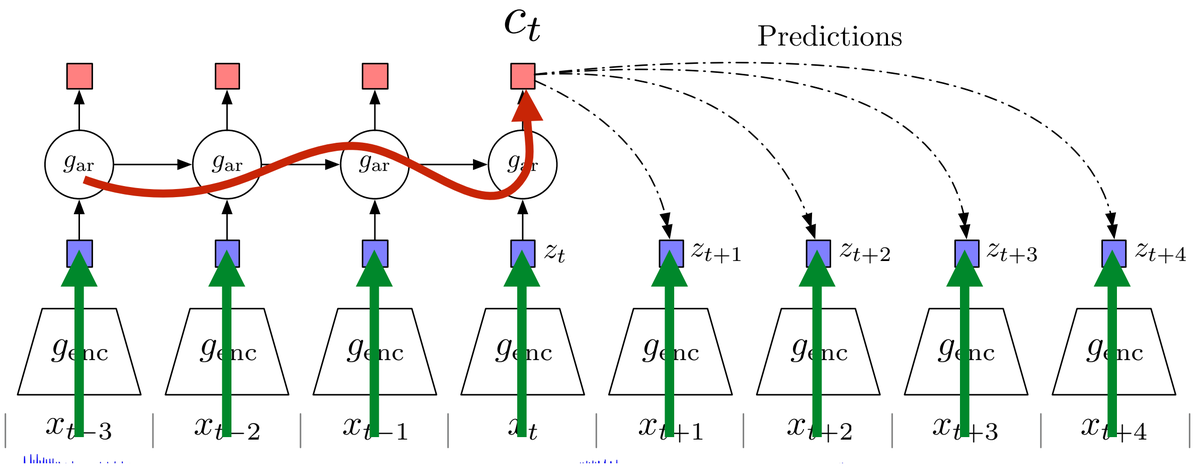
Predictive Coding: Using features from **upstream observations (x_{t-i})** to predict features in **downstream observations (x_{t+j})**. To avoid unnecessary modeling of low-level features, **compact latent vectors (z_t)** are used instead of raw observations.

Procedure:

1. Generate compact representations for each observation **($x_t \rightarrow z_t$ via g_{enc})**.

Representation Learning with Contrastive Predictive Coding
https://arxiv.org/pdf/1807.03748.pdf

Contrastive Predictive Coding



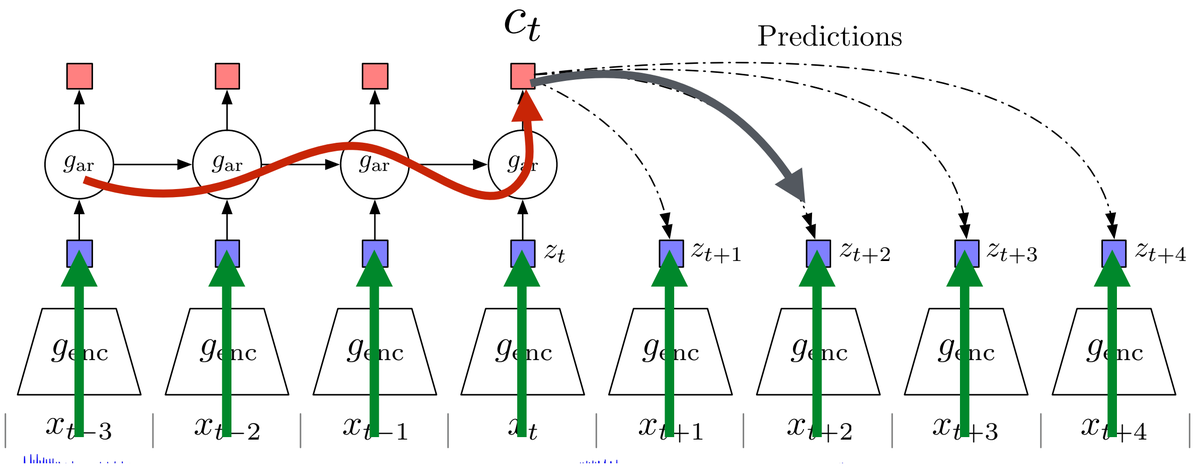
Predictive Coding: Using features from **upstream observations (x_{t-i})** to predict features in **downstream observations (x_{t+j})**. To avoid unnecessary modeling of low-level features, **compact latent vectors (z_t)** are used instead of raw observations.

Procedure:

1. Generate compact representations for each observation **($x_t \rightarrow z_t$ via g_{enc})**.
2. Aggregate compact vectors to generate context **($c_t = g_{ar}(z_{\leq t})$)**.

Representation Learning with Contrastive Predictive Coding
https://arxiv.org/pdf/1807.03748.pdf

Contrastive Predictive Coding



Predictive Coding: Using features from **upstream observations (x_{t-i})** to predict features in **downstream observations (x_{t+j})**. To avoid unnecessary modeling of low-level features, **compact latent vectors (z_t)** are used instead of raw observations.

Procedure:

1. Generate compact representations for each observation **($x_t \rightarrow z_t$ via g_{enc})**.
2. Aggregate compact vectors to generate context **($c_t = g_{ar}(z_{\leq t})$)**.
3. Use context to predict future compact vectors and compare to context-free vector: **$p(z_{t+j} \mid c_t)/p(z_{t+j})$**.

Representation Learning with Contrastive Predictive Coding
https://arxiv.org/pdf/1807.03748.pdf

CPC Loss Function

CPC Loss Function

- **Goal: Preserve Mutual Information between original signal and context.**

$$I(x;c) = \sum_{x,c} p(x,c) \log \frac{p(x|c)}{p(x)}.$$

CPC Loss Function

- **Goal: Preserve Mutual Information between original signal and context.**

$$I(x;c) = \sum_{x,c} p(x,c) \log \frac{p(x|c)}{p(x)}.$$

- **Generate function proportional to MI:**

$$f_k(x_{t+k}, c_t) \propto \frac{p(x_{t+k}|c_t)}{p(x_{t+k})}$$

Example used in CPC papers, many other options exist.

$$f_k(x_{t+k}, c_t) = \exp\left(z_{t+k}^T W_k c_t\right),$$

CPC Loss Function

- **Goal: Preserve Mutual Information between original signal and context.**

$$I(x;c) = \sum_{x,c} p(x,c) \log \frac{p(x|c)}{p(x)}.$$

- **Generate function proportional to MI:**

Example used in CPC papers, many other options exist.

$$f_k(x_{t+k}, c_t) \propto \frac{p(x_{t+k}|c_t)}{p(x_{t+k})} \qquad f_k(x_{t+k}, c_t) = \exp\left(z_{t+k}^T W_k c_t\right),$$

- **Leverage Noise Contrastive Estimation (NCE) to optimize loss:**

$$\mathcal{L}_\mathrm{N} = -\underset{X}{\mathbb{E}}\left[\log \frac{f_k(x_{t+k}, c_t)}{\sum_{x_j \in X} f_k(x_j, c_t)}\right]$$

Categorical cross-entropy loss

$X=\{x_1, \ldots x_N\}$ with 1 "**positive**" sample and N-1 "**negative**" samples.

Positive: from $p(x_{t+k}|c_t)$

Negative: from $p(x_{t+k})$

NCE Loss

$$\mathcal{L}_{\mathrm{N}} = -\mathop{\mathbb{E}}_{X}\left[\log \frac{f_k(x_{t+k}, c_t)}{\sum_{x_j \in X} f_k(x_j, c_t)}\right]$$

$X=\{x_1, \ldots x_N\}$ with 1 "**positive**" sample and N-1 "**negative**" samples.

Positive: from $p(x_{t+k}|c_t)$

Negative: from $p(x_{t+k})$

NCE Loss

$$\mathcal{L}_{\mathrm{N}} = -\mathop{\mathbb{E}}_{X}\left[\log \frac{f_k(x_{t+k}, c_t)}{\sum_{x_j \in X} f_k(x_j, c_t)}\right]$$

$X=\{x_1, \ldots x_N\}$ with 1 "**positive**" sample and N-1 "**negative**" samples.

Positive: from $p(x_{t+k}|c_t)$

Negative: from $p(x_{t+k})$

Let $p(d=i \mid X, c_t)$:= Optimal probability for loss, where $[d=i]$ means x_i is a **positive** sample.

NCE Loss

$$\mathcal{L}_{\mathrm{N}} = -\mathop{\mathbb{E}}_{X}\left[\log \frac{f_k(x_{t+k}, c_t)}{\sum_{x_j \in X} f_k(x_j, c_t)}\right]$$

$X=\{x_1, \ldots x_N\}$ with 1 "**positive**" sample and N-1 "**negative**" samples.

Positive: from $p(x_{t+k}|c_t)$

Negative: from $p(x_{t+k})$

Let $p(d=i \mid X, c_t)$:= Optimal probability for loss, where $[d=i]$ means x_i is a **positive** sample.

Thus, probability that x_i is from **positive** sample instead of **negative** is:

$$p(d=i|X,c_t) = \frac{p(x_i|c_t)\prod_{l\neq i} p(x_l)}{\sum_{j=1}^{N} p(x_j|c_t)\prod_{l\neq j} p(x_l)}$$

$$= \frac{\frac{p(x_i|c_t)}{p(x_i)}}{\sum_{j=1}^{N} \frac{p(x_j|c_t)}{p(x_j)}}.$$

NCE Loss

$$\mathcal{L}_{\mathrm{N}} = -\mathop{\mathbb{E}}_{X}\left[\log \frac{f_k(x_{t+k}, c_t)}{\sum_{x_j \in X} f_k(x_j, c_t)}\right]$$

$X=\{x_1, \ldots x_N\}$ with 1 "**positive**" sample and N-1 "**negative**" samples.

Positive: from $p(x_{t+k}|c_t)$

Negative: from $p(x_{t+k})$

Let $p(d=i \mid X, c_t)$:= Optimal probability for loss, where $[d=i]$ means x_i is a **positive** sample.

Thus, probability that x_i is from **positive** sample instead of **negative** is:

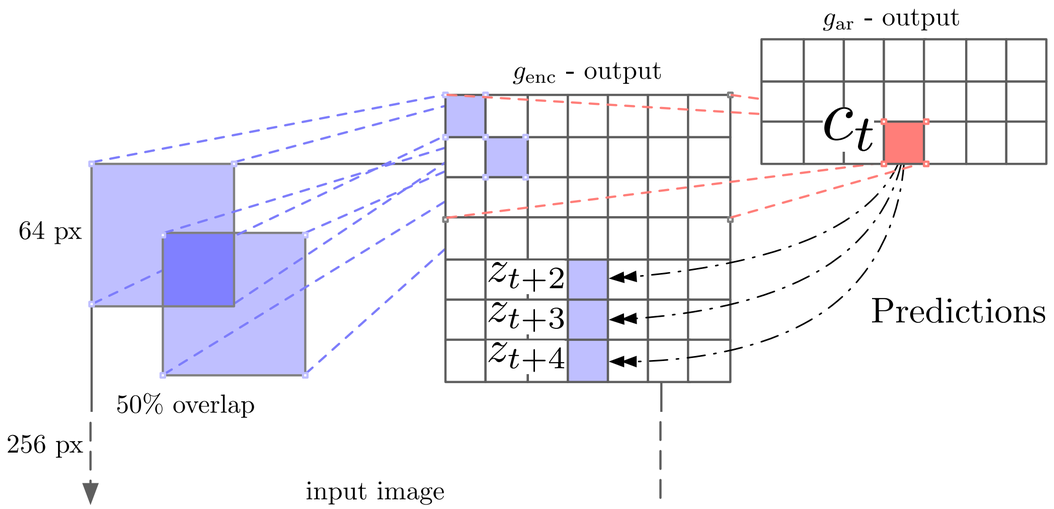
$$p(d=i|X,c_t) = \frac{p(x_i|c_t)\prod_{l\neq i} p(x_l)}{\sum_{j=1}^{N} p(x_j|c_t)\prod_{l\neq j} p(x_l)}$$

$$= \frac{\frac{p(x_i|c_t)}{p(x_i)}}{\sum_{j=1}^{N}\frac{p(x_j|c_t)}{p(x_j)}}. \quad \propto \frac{p(x_{t+k}|c_t)}{p(x_{t+k})} \quad \textit{(MI Goal)}$$

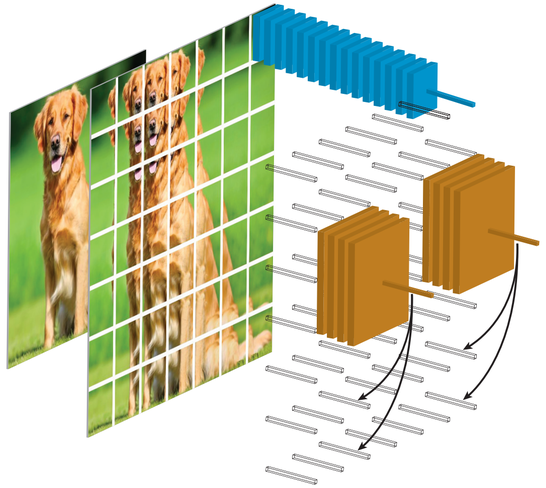
NCE Loss (detailed)

$$
\begin{aligned}
\mathcal{L}_{\mathrm{N}}^{\mathrm{opt}} &= -\mathop{\mathbb{E}}_{X} \log \left[\frac{\frac{p(x_{t+k}|c_t)}{p(x_{t+k})}}{\frac{p(x_{t+k}|c_t)}{p(x_{t+k})} + \sum_{x_j \in X_{\mathrm{neg}}} \frac{p(x_j|c_t)}{p(x_j)}} \right] \\
&= \mathop{\mathbb{E}}_{X} \log \left[1 + \frac{p(x_{t+k})}{p(x_{t+k}|c_t)} \sum_{x_j \in X_{\mathrm{neg}}} \frac{p(x_j|c_t)}{p(x_j)} \right] \\
&\approx \mathop{\mathbb{E}}_{X} \log \left[1 + \frac{p(x_{t+k})}{p(x_{t+k}|c_t)} (N-1) \mathop{\mathbb{E}}_{x_j} \frac{p(x_j|c_t)}{p(x_j)} \right] \\
&= \mathop{\mathbb{E}}_{X} \log \left[1 + \frac{p(x_{t+k})}{p(x_{t+k}|c_t)} (N-1) \right] \\
&\geq \mathop{\mathbb{E}}_{X} \log \left[\frac{p(x_{t+k})}{p(x_{t+k}|c_t)} N \right] \\
&= -I(x_{t+k}, c_t) + \log(N),
\end{aligned}
$$

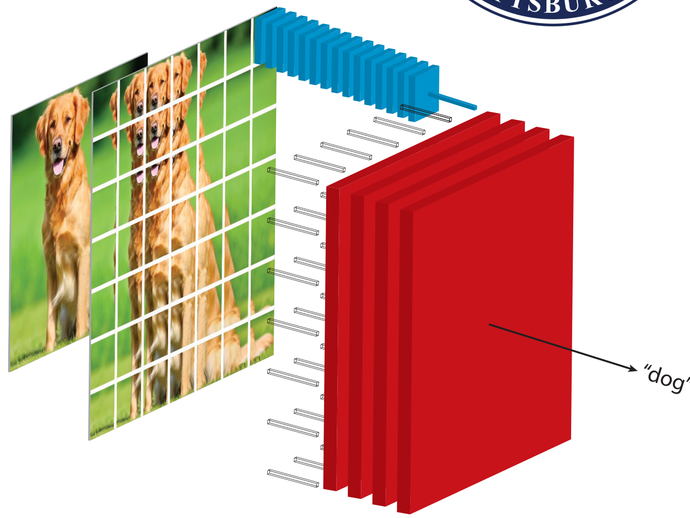
CPC For Images



Unsupervised pre-training



Supervised fine-tuning

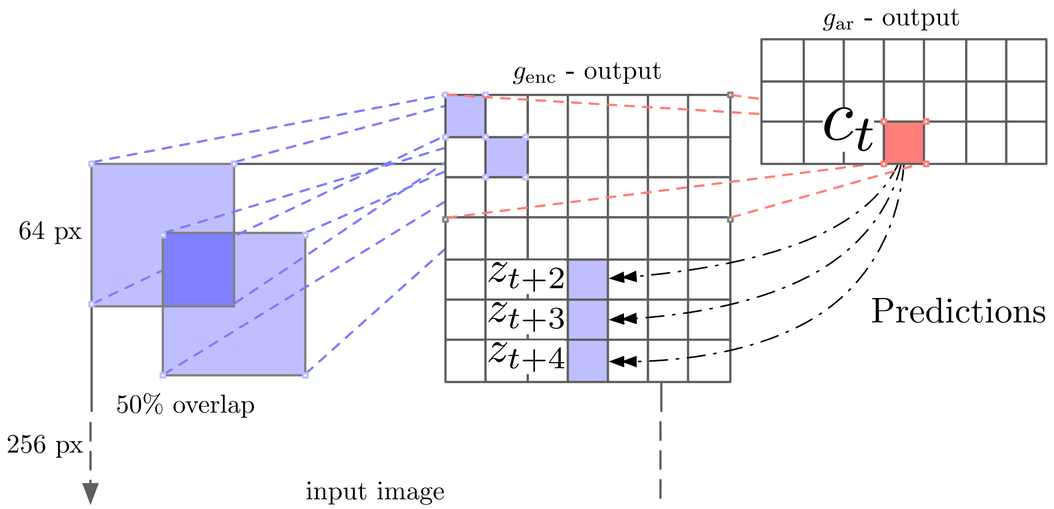


Maxpool z_t's into single vector of size 2048 (4096) for input into downstream task.

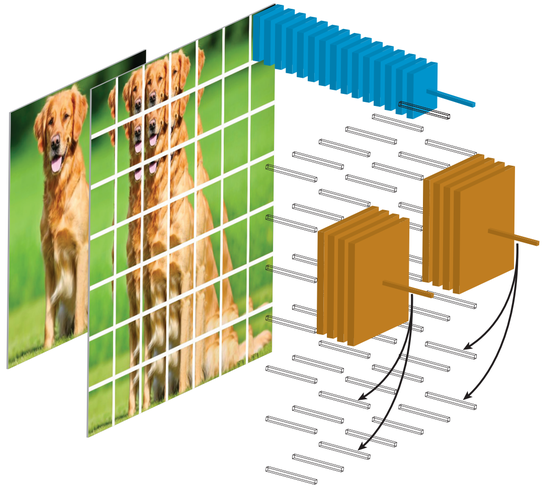
Representation Learning with Contrastive Predictive Coding
https://arxiv.org/pdf/1807.03748.pdf

Data-Efficient Image Recognition with Contrastive Predictive Coding
https://arxiv.org/pdf/1905.09272.pdf

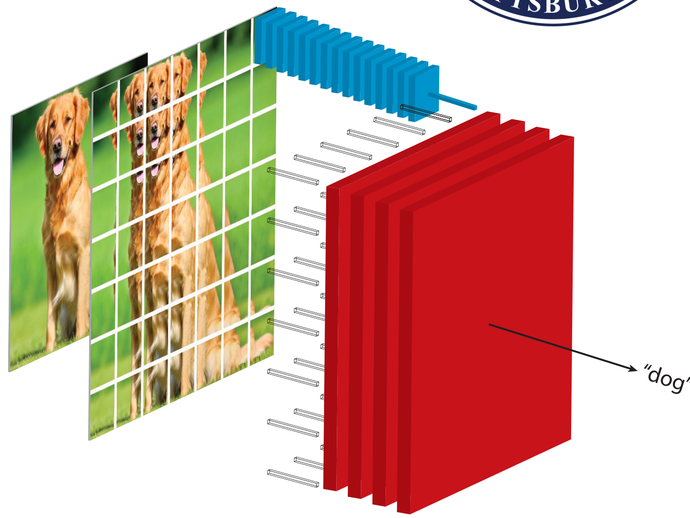
CPC For Images



Unsupervised pre-training



Supervised fine-tuning



Maxpool z_t's into single vector of size 2048 (4096) for input into downstream task.

Results (c.2018)

Method	Top-1 ACC
Using AlexNet conv5	
Video [28]	29.8
Relative Position [11]	30.4
BiGan [35]	34.8
Colorization [10]	35.2
Jigsaw [29] *	38.1
Using ResNet-V2	
Motion Segmentation [36]	27.6
Exemplar [36]	31.5
Relative Position [36]	36.2
Colorization [36]	39.6
CPC	**48.7**

Table 3: ImageNet top-1 unsupervised classification results. *Jigsaw is not directly comparable to the other AlexNet results because of architectural differences.

Method	Top-5 ACC
Motion Segmentation (MS)	48.3
Exemplar (Ex)	53.1
Relative Position (RP)	59.2
Colorization (Col)	62.5
Combination of MS + Ex + RP + Col	69.3
CPC	**73.6**

Table 4: ImageNet top-5 unsupervised classification results. Previous results with MS, Ex, RP and Col were taken from [36] and are the best reported results on this task.

Representation Learning with Contrastive Predictive Coding
https://arxiv.org/pdf/1807.03748.pdf

Data-Efficient Image Recognition with Contrastive Predictive Coding
https://arxiv.org/pdf/1905.09272.pdf

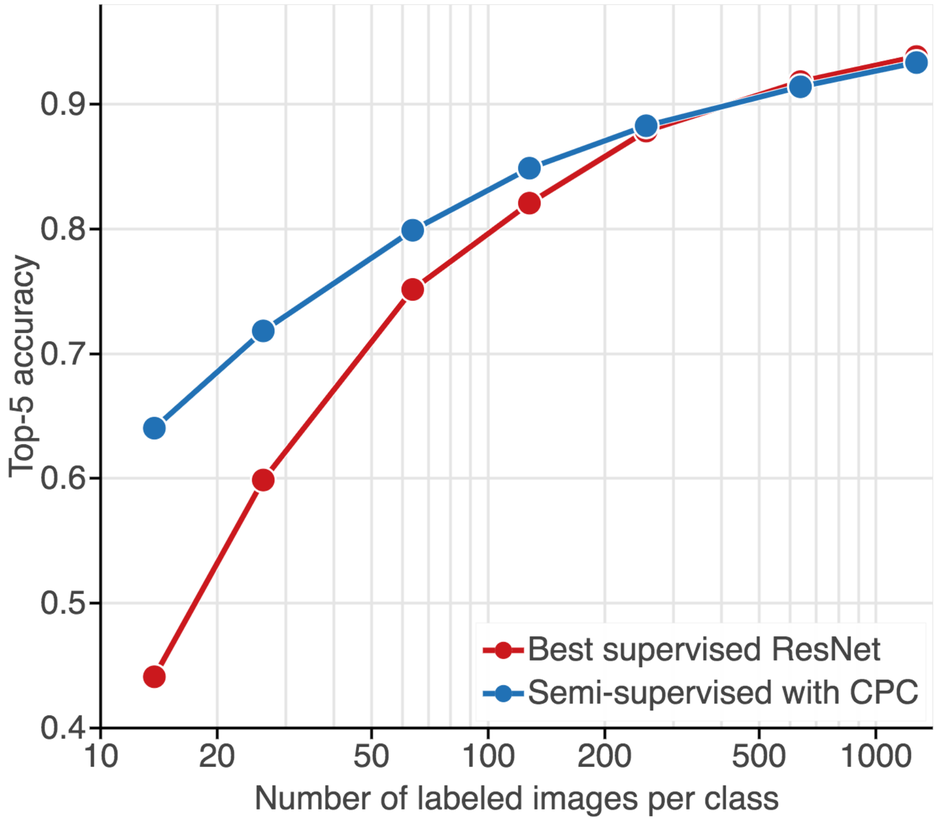
CPC and Data Efficiency

★ **DeepMind has improved upon initial results with CPC:**

- g_{enc} has grown from ResNet 101 to ResNet 170.
- Layer normalization instead of BatchNorm.
- More tasks: Upper patches predict lower and vice versa.
  - Original CPC only had upper predict lower.
- More data augmentation (including color dropping).

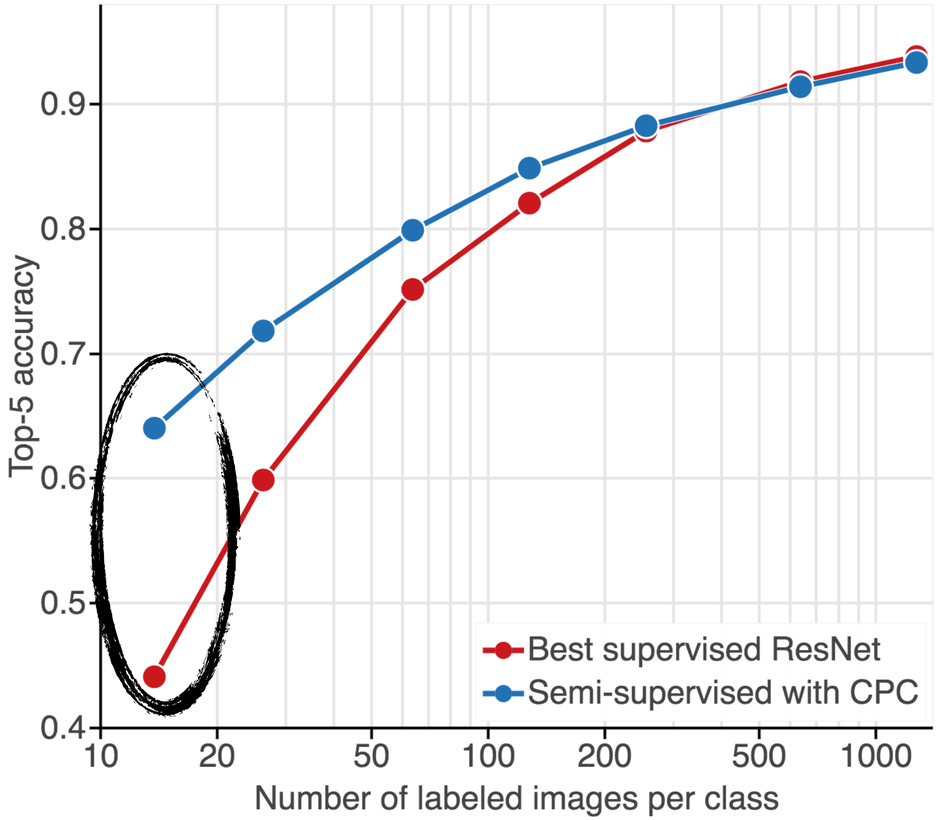
CPC and Data Efficiency

- ★ **DeepMind has improved upon initial results with CPC:**
  - g_{enc} has grown from ResNet 101 to ResNet 170.
  - Layer normalization instead of BatchNorm.
  - More tasks: Upper patches predict lower and vice versa.
    - Original CPC only had upper predict lower.
  - More data augmentation (including color dropping).



CPC and Data Efficiency

- ★ **DeepMind has improved upon initial results with CPC:**
  - g_{enc} has grown from ResNet 101 to ResNet 170.
  - Layer normalization instead of BatchNorm.
  - More tasks: Upper patches predict lower and vice versa.
    - Original CPC only had upper predict lower.
  - More data augmentation (including color dropping).



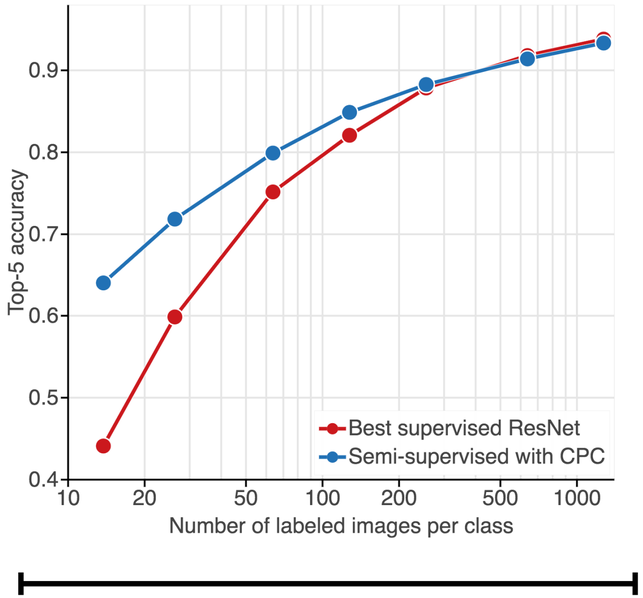
Results: Linear Separability

Linear separability using the max-pooled feature vectors generated by the CPC network. Each image becomes a 4096 dim vector.

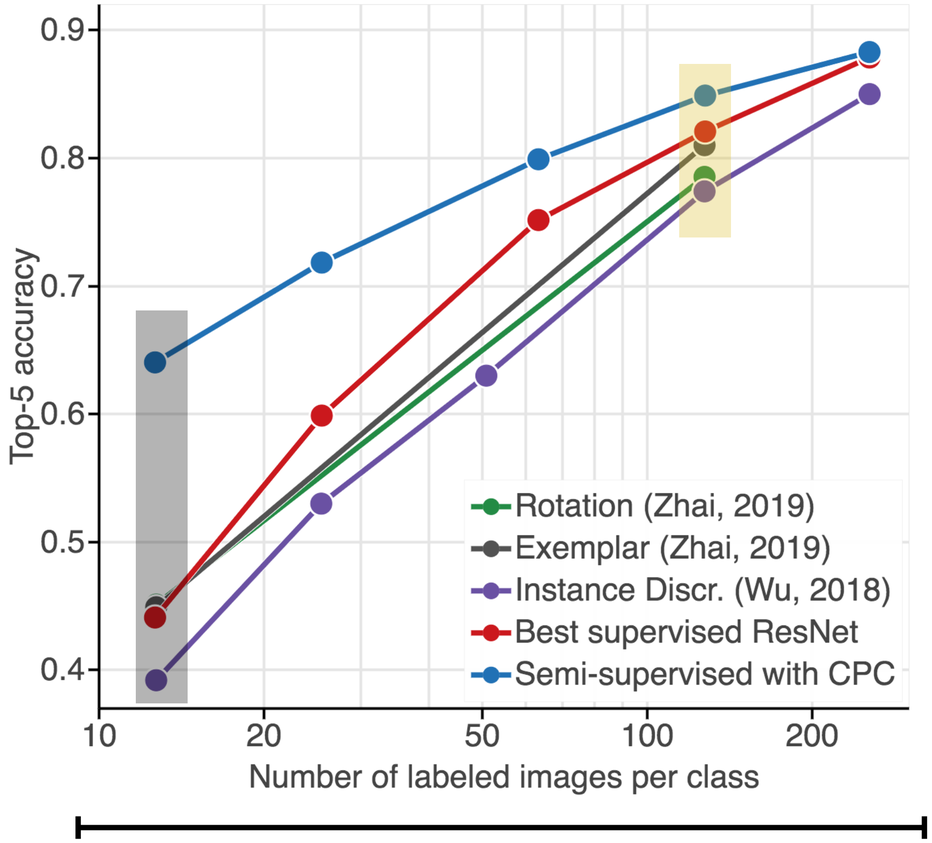
Method	Top-1	Top-5
Motion Segmentation (MS) [50]	27.6	48.3
Exemplar (Ex) [17]	31.5	53.1
Relative Position (RP) [14]	36.2	59.2
Colorization (Col) [69]	39.6	62.5
Combination of MS + Ex + RP + Col [15]	-	69.3
CPC [49]	48.7	73.6
Rotation + RevNet [36]	55.4	-
CPC (ours)	**61.0**	**83.0**
AlexNet	**59.3**	**81.8**

Table 1. Comparison to linear separability of other self-supervised methods. In all cases a feature extractor is optimized in an unsupervised manner, and a linear classifier is trained using all labels in the ImageNet dataset.

Results: Low-data



1% to 100% of labeled images



1% to ~20% of labeled images

Labeled data **Method**	1% **Top-5 accuracy**	10%
Supervised baseline	44.10	82.08
Methods using label-propagation:		
Pseudolabeling [68]	51.56	82.41
VAT [68]	44.05	82.78
VAT + Entropy Minimization [68]	46.96	83.39
Unsup. Data Augmentation [65]	-	88.52
Rotation + VAT + Ent. Min. [68]	-	**91.23**
Methods only using representation learning:		
Instance Discrimination [64]	39.20	77.40
Exemplar [68]	44.90	81.01
Exemplar (joint training) [68]	47.02	83.72
Rotation [68]	45.11	78.53
Rotation (joint training) [68]	53.37	83.82
CPC (ours)	**64.03**	**84.88**

Results: Transfer PASCAL

Method	mAP
Transfer from labeled ImageNet:	
Supervised - ResNet-152	74.7
Transfer from unlabeled ImageNet:	
Exemplar (Ex) [17]	60.9
Motion Segmentation (MS) [50]	61.1
Colorization (Col) [69]	65.5
Relative Position (RP) [14]	66.8
Combination of Ex + MS + Col + RP [15]	70.5
Deep Cluster [8]	65.9
Deeper Cluster [9]	67.8
CPC - ResNet-101	70.6
CPC - ResNet-170	**72.1**

Table 3. Comparison of PASCAL 2007 image detection accuracy to other transfer methods. The first class of methods learn from unlabeled ImageNet data and fine-tune for PASCAL detection. The second class learns from the entire labeled ImageNet dataset before transferring. All results are reported in terms of mean average precision (mAP).

Tuning and Analysis

Frozen vs. Fine-tuning

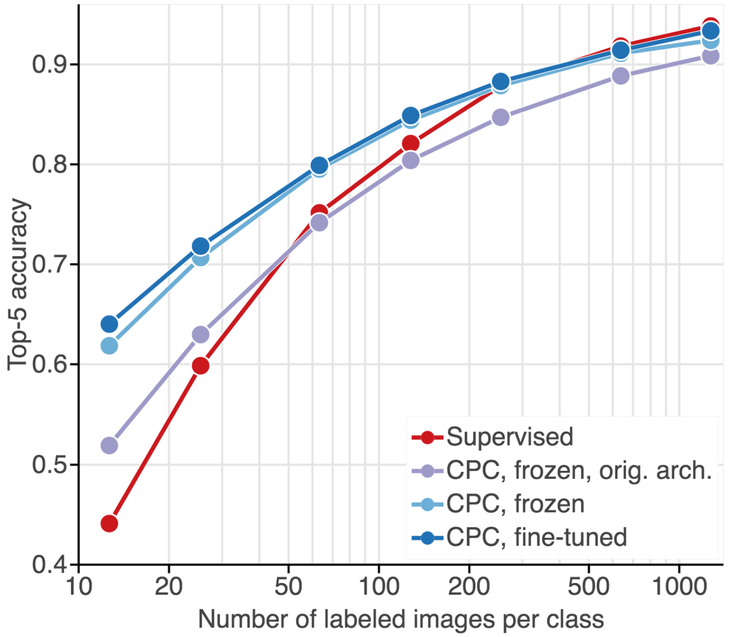


Figure 4. Contribution of unsupervised learning and fine-tuning to recognition performance. Light blue: classification performance of an frozen feature extractor followed by a supervised classifier. Purple: similarly, but with the original CPC architecture. Dark blue: classification performance of the fine-tuned model. Red: fully supervised baseline.

Number of Iterations

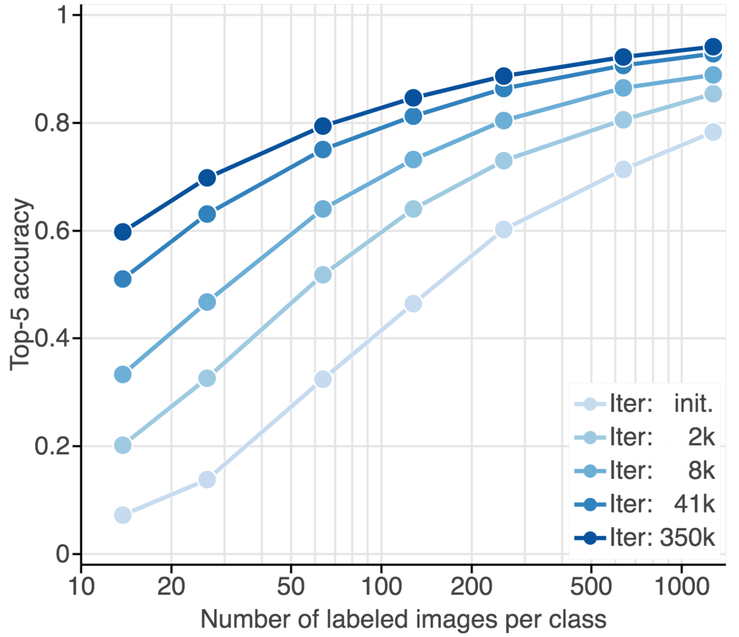


Figure 5. Image recognition accuracy over the course of CPC training. Without training, the ResNet-170 architecture achieves very low performance across data regimes. Over the course of training, this performance increases rapidly, reaching our final result after 350k iterations.