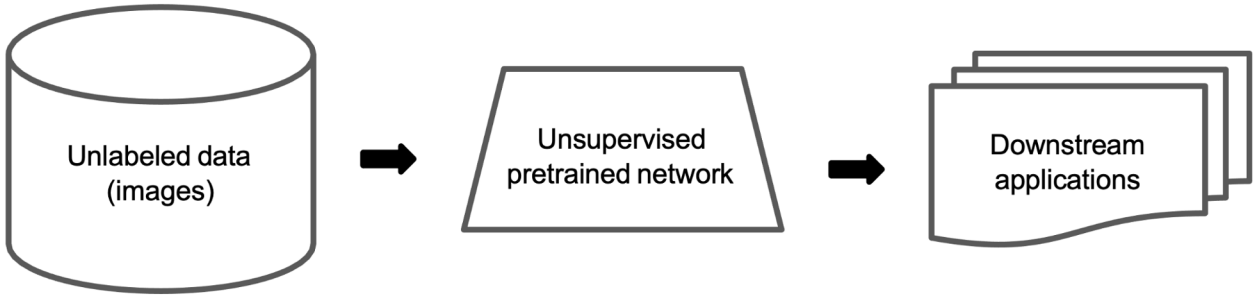
Self-Supervised Learning論文介紹：BYOL

BYOL

BYOL全名是“Bootstrap Your Own Latent: A New Approach to Self-Supervised Learning”，是由DeepMind團隊發表在NIPS2020的論文。

BYOL目標與其他自監督學習一樣，都要利用**無標註的資料**訓練模型成為好的特徵抽取器。而這特徵抽取器可以遷移到其他的電腦視覺任務。



BYOL貢獻

在之前的論文講解中，SimCLR與MoCo都是採用正負樣本的對比來學習圖片的特徵，而為了有大量的負樣本提供對比，其分別使用large batch size與memory queue來達成。

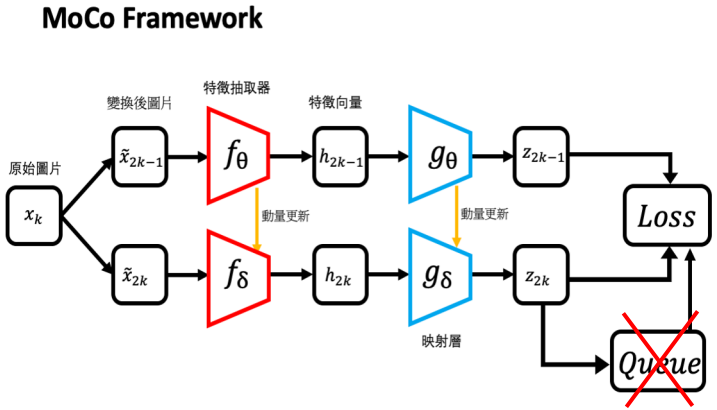
而BYOL提出**不使用負樣本**，只使用正樣本的預測來做圖片特徵學習。

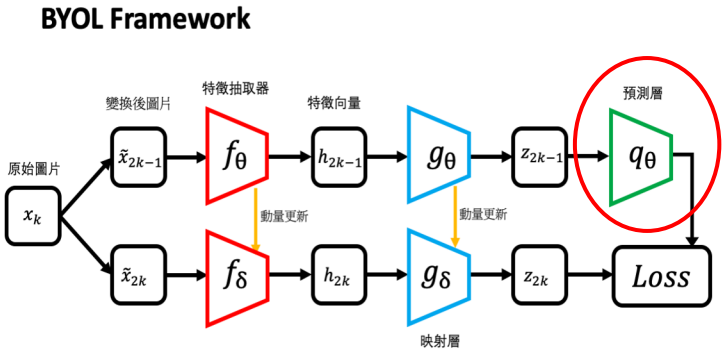
在不使用負樣本的情況下，BYOL宣稱可以在比較小的batch size下維持分類準確率。

BYOL模型架構

架構上與MoCo類似，都是採用非對稱的模型架構，分別有可訓練的特徵抽取器 + 映射層 + 預測層（f_θ、g_θ與q_θ），與動量更新的特徵抽取器 + 映射層（f_δ和g_δ），並且拿掉儲存負樣本的queue。

MoCo負樣本：queue
BYOL負樣本：不使用

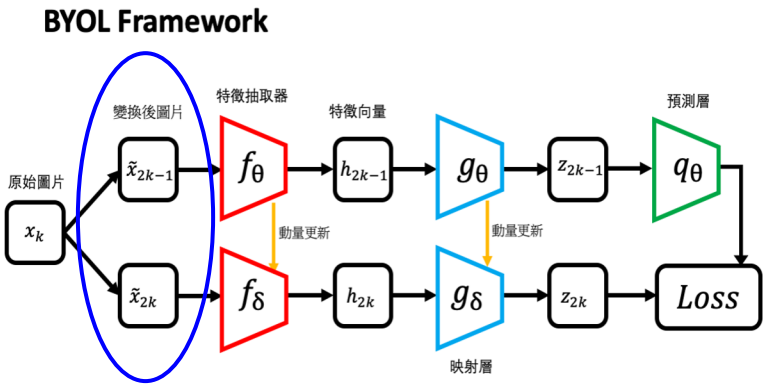




BYOL第一階段的圖片變換

對比學習的第一階段都是圖片變換，要在圖片變換夠多的情況下又保持重要的特徵，讓模型學習圖片不變性。

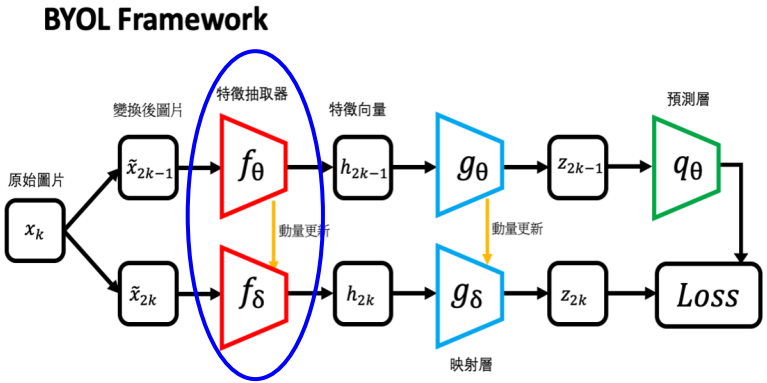
BYOL採取的圖片變換與之前的自監督學習方法（SimCLR, MoCo）一樣，都是以隨機裁切 + 顏色變換為主。



BYOL第二階段的特徵抽取

BYOL使用的特徵抽取器也是採取CNN理念的ResNet架構，需要注意的是上面的f_θ是最後要保留的特徵抽取器，是使用gradients更新，而下面的f_δ是動量更新，不使用gradients。

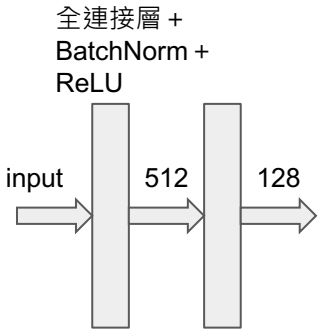
動量更新：f_δ = m*f_δ + (1-m)*f_θ，在BYOL中m設為0.99來模擬緩慢更新的f_δ（與MoCo一樣）。

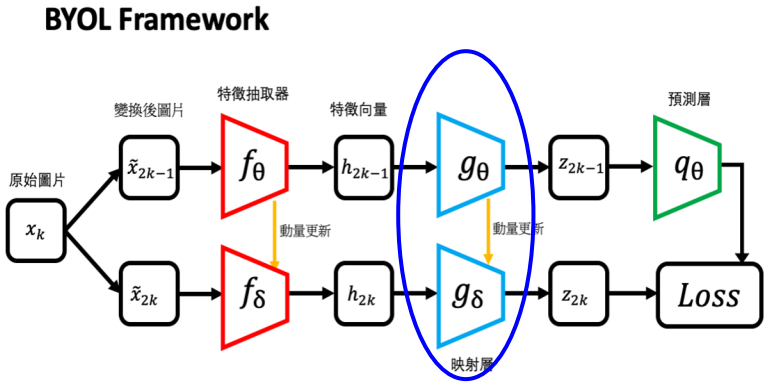


BYOL第三階段的特徵映射

映射層是由全連接層、BatchNorm與ReLU所組成，中間會有一層512維度的隱藏層，而輸出維度會降低成128維。

映射層g_θ是用gradients更新，而映射層g_δ是採取動量更新，g_δ = m*g_δ + (1-m)*g_θ。

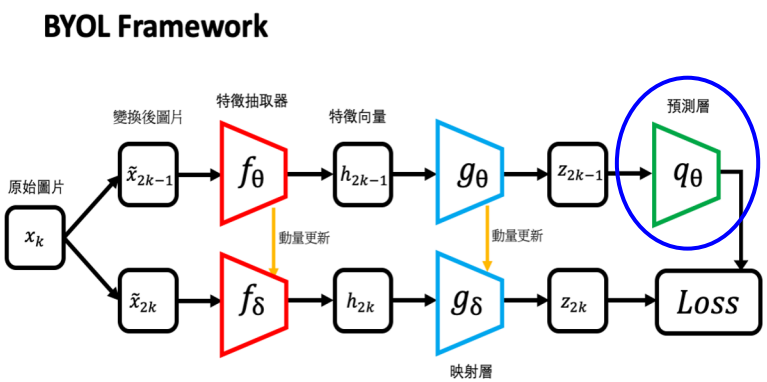




BYOL第四階段的預測層

預測層與映射層是一樣的模型，都是由全連接層加上BatchNorm與ReLU組成。

q_θ輸出128維度向量，並且與g_δ的輸出結果做loss計算。



BYOL第五階段的Loss計算

把預測層q_θ與映射層g_δ的輸出向量做normalize後，再去做L2 loss的計算（實際上等於2-2*cosine_similarity）。

實際上就是算向量的cosine similarity

normalize 預測層的輸出

normalize 映射層的輸出

$$\mathcal{L}_{\theta,\xi} \triangleq \left\| \overline{q_\theta}(z_\theta) - \overline{z}'_\xi \right\|_2^2 = 2 - 2 \cdot \frac{\langle q_\theta(z_\theta), z'_\xi \rangle}{\left\| q_\theta(z_\theta) \right\|_2 \cdot \left\| z'_\xi \right\|_2}.$$

BYOL演算法流程

Algorithm 1: BYOL: **B**ootstrap **Y**our **O**wn **L**atent

Inputs :

$\mathcal{D}$, $\mathcal{T}$, and $\mathcal{T}'$	set of images and distributions of transformations
θ, f_θ, g_θ, and q_θ	initial online parameters, encoder, projector, and predictor
ξ, f_ξ, g_ξ	initial target parameters, target encoder, and target projector
optimizer	optimizer, updates online parameters using the loss gradient
K and N	total number of optimization steps and batch size
$\{\tau_k\}_{k=1}^{K}$ and $\{\eta_k\}_{k=1}^{K}$	target network update schedule and learning rate schedule

for $k = 1$ **to** K **do**

  $\mathcal{B} \leftarrow \{x_i \sim \mathcal{D}\}_{i=1}^{N}$ // sample a batch of N images

  for $x_i \in \mathcal{B}$ **do**

    $t \sim \mathcal{T}$ and $t' \sim \mathcal{T}'$ // sample image transformations

    $z_1 \leftarrow g_\theta(f_\theta(t(x_i)))$ and $z_2 \leftarrow g_\theta(f_\theta(t'(x_i)))$ // compute projections

    $z'_1 \leftarrow g_\xi(f_\xi(t'(x_i)))$ and $z'_2 \leftarrow g_\xi(f_\xi(t(x_i)))$ // compute target projections

    $l_i \leftarrow -2 \cdot \left(\frac{\langle q_\theta(z_1), z'_1 \rangle}{\|q_\theta(z_1)\|_2 \cdot \|z'_1\|_2} + \frac{\langle q_\theta(z_2), z'_2 \rangle}{\|q_\theta(z_2)\|_2 \cdot \|z'_2\|_2} \right)$ // compute the loss for x_i

  end

  $\delta\theta \leftarrow \frac{1}{N} \sum_{i=1}^{N} \partial_\theta l_i$ // compute the total loss gradient w.r.t. θ

  $\theta \leftarrow \text{optimizer}(\theta, \delta\theta, \eta_k)$ // update online parameters

  $\xi \leftarrow \tau_k \xi + (1 - \tau_k)\theta$ // update target parameters

end

Output : encoder f_θ

評估模型效果的方式(same as SimCLR)

在Self-Supervised Learning訓練中，因為不使用標註資料，所以訓練過程無法知道準確率如何，只能觀察loss是否下降，那要怎樣評估模型的好壞呢？

Linear Evaluation Protocol：在SSL訓練結束後，我們會把**模型f的參數凍結**，並在模型f後面接上全連接層做分類任務，這個階段我們會用**有標註**的資料訓練全連接層，這樣的方式可以評估模型f的特徵抽取能力好不好。

Fine-tune：**模型f的參數可以一起更新**，但通常這樣的評估方式會只使用小部分的標註資料，可能10%或1%訓練資料。

Transfer Pretrained Model：把在A資料集訓練好的模型f當作起始參數用在其他的資料集B上。

BYOL實驗：Linear Evaluation

在論文中，BYOL在ImageNet上面訓練了1000 epochs後，做linear evaluation來測試圖像分類的準確率，並且與其他自監督學習演算法做比較。

普通的監督式學習的話準確率為76.5%，而由下圖可以看到BYOL比其他的方法好之外，只比普通監督式學習低2.2%。

Method	Top-1	Top-5
Local Agg.	60.2	-
PIRL [35]	63.6	-
CPC v2 [32]	63.8	85.3
CMC [11]	66.2	87.0
SimCLR [8]	69.3	89.0
MoCo v2 [37]	71.1	-
InfoMin Aug. [12]	73.0	91.1
BYOL (ours)	**74.3**	**91.6**

(a) ResNet-50 encoder.

BYOL實驗：Linear Evaluation

如果使用更大的模型當作特徵抽取器，BYOL可以再近一步提升準確率，在不同大小的模型訓練下，BYOL也比其他的自監督演算法好（通常自監督學習比監督式學習更能受惠於複雜的模型）。

Method	Architecture	Param.	Top-1	Top-5
SimCLR [8]	ResNet-50 (2×)	94M	74.2	92.0
CMC [11]	ResNet-50 (2×)	94M	70.6	89.7
BYOL (ours)	ResNet-50 (2×)	94M	77.4	93.6
CPC v2 [32]	ResNet-161	305M	71.5	90.1
MoCo [9]	ResNet-50 (4×)	375M	68.6	-
SimCLR [8]	ResNet-50 (4×)	375M	76.5	93.2
BYOL (ours)	ResNet-50 (4×)	375M	78.6	94.2
BYOL (ours)	ResNet-200 (2×)	250M	**79.6**	**94.8**

(b) Other ResNet encoder architectures.

BYOL實驗：Fine Tune

在fine tune階段，我們模擬少量標註資料的情況，只使用10%, 1%的ImageNet有標註資料來更新模型，可以看到在1%的情況下可以遠超過監督式學習的準確率。

而BYOL不管在小模型或大模型下，都是表現最好的自監督演算法之一。

Method	Top-1		Top-5	
	1%	10%	1%	10%
Supervised [77]	25.4	56.4	48.4	80.4
InstDisc	-	-	39.2	77.4
PIRL [35]	-	-	57.2	83.8
SimCLR [8]	48.3	65.6	75.5	87.8
BYOL (ours)	**53.2**	**68.8**	**78.4**	**89.0**

(a) ResNet-50 encoder.

Method	Architecture	Param.	Top-1		Top-5	
			1%	10%	1%	10%
CPC v2 [32]	ResNet-161	305M	-	-	77.9	91.2
SimCLR [8]	ResNet-50 (2×)	94M	58.5	71.7	83.0	91.2
BYOL (ours)	ResNet-50 (2×)	94M	62.2	73.5	84.1	91.7
SimCLR [8]	ResNet-50 (4×)	375M	63.0	74.4	85.8	92.6
BYOL (ours)	ResNet-50 (4×)	375M	69.1	75.7	87.9	92.5
BYOL (ours)	ResNet-200 (2×)	250M	**71.2**	**77.7**	**89.5**	**93.7**

(b) Other ResNet encoder architectures.

BYOL實驗：Transfer Learning

在這一階段，把BYOL在ImageNet上面做完自監督學習的模型遷移到其他的圖像分類資料集上，並且使用linear evaluation跟fine tune兩種方式測試模型的遷移性。

由下圖的實驗結果可以看到，BYOL學習到的圖片特徵是通用的，並非只適用在ImageNet上面，在其他的圖像分類資料集都仍有不錯的結果。

Method	Food101	CIFAR10	CIFAR100	Birdsnap	SUN397	Cars	Aircraft	VOC2007	DTD	Pets	Caltech-101	Flowers
Linear evaluation:												
BYOL (ours)	**75.3**	91.3	**78.4**	**57.2**	**62.2**	**67.8**	60.6	82.5	75.5	90.4	94.2	**96.1**
SimCLR (repro)	72.8	90.5	74.4	42.4	60.6	49.3	49.8	81.4	**75.7**	84.6	89.3	92.6
SimCLR [8]	68.4	90.6	71.6	37.4	58.8	50.3	50.3	80.5	74.5	83.6	90.3	91.2
Supervised-IN [8]	72.3	**93.6**	78.3	53.7	61.9	66.7	**61.0**	**82.8**	74.9	**91.5**	**94.5**	94.7
Fine-tuned:												
BYOL (ours)	**88.5**	**97.8**	86.1	**76.3**	63.7	91.6	**88.1**	**85.4**	**76.2**	91.7	**93.8**	97.0
SimCLR (repro)	87.5	97.4	85.3	75.0	63.9	91.4	87.6	84.5	75.4	89.4	91.7	96.6
SimCLR [8]	88.2	97.7	85.9	75.9	63.5	91.3	88.1	84.1	73.2	89.2	92.1	97.0
Supervised-IN [8]	88.3	97.5	**86.4**	75.8	**64.3**	**92.1**	86.0	85.0	74.6	**92.1**	93.3	**97.6**
Random init [8]	86.9	95.9	80.2	76.1	53.6	91.4	85.9	67.3	64.8	81.5	72.6	92.0

Table 3: Transfer learning results from ImageNet (IN) with the standard ResNet-50 architecture.

BYOL實驗：Transfer Learning

在這一階段，把BYOL在ImageNet上面做完自監督學習的模型遷移到其他的電腦視覺任務上面（VOC2012），像是影像分割和物件偵測，以此顯示BYOL學習到的特徵不僅可以用在圖像分類，也可以用在其他需要圖片特徵的電腦視覺任務上。

結果顯示，模型的遷移能力比Supervised, SimCLR和MoCo還好。

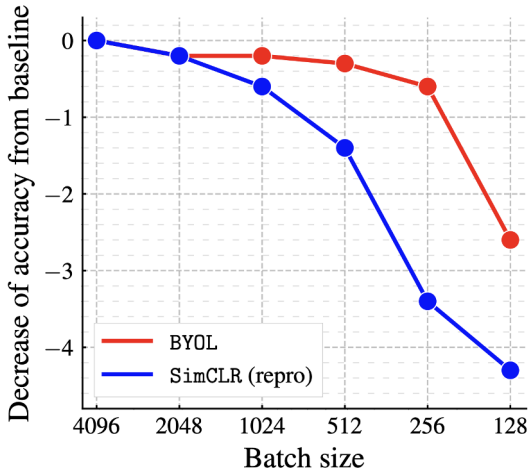
Method	AP_{50}	mIoU
Supervised-IN [9]	74.4	74.4
MoCo [9]	74.9	72.5
SimCLR (repro)	75.2	75.2
BYOL (ours)	**77.5**	**76.3**

(a) Transfer results in semantic segmentation and object detection.

BYOL實驗：Batch Size

前面有提到BYOL宣稱它的演算法可以在小batch size情況下，維持較好的準確率。

從下圖可以看到，BYOL確實在batch size=2048/1024/512的情況下都沒有明顯的衰退，相較之下SimCLR的衰退就明顯許多。



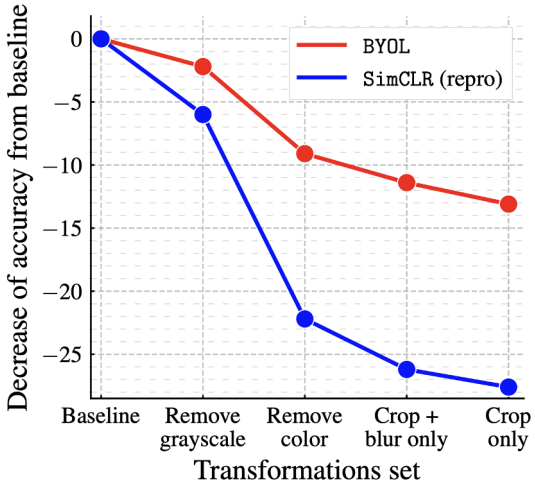
(a) Impact of batch size

BYOL：圖片變換的依賴程度

我們知道自監督學習的圖片變換最好是複數種加在一起組成高強度的圖片變換，且對於SimCLR來說裁切（Crop）尤其重要。

那BYOL對於圖片變換的依賴程度呢？會不會像SimCLR一樣少了某種圖片變換效果就大幅降低效果？

=>BYOL降低幅度較少！



(b) Impact of progressively removing transformations

BYOL：動量更新超參數

BYOL與MoCo都有使用動量更新的模型（f_δ = m*f_δ + (1-m)*f_θ），在MoCo原論文中實驗顯示m設為0.99有最好的效果，在BYOL中做的實驗也顯示0.99是最適合的更新量（這邊的τ就是m參數）。

Target	τ_{base}	Top-1
Constant random network	1	$18.8_{\pm 0.7}$
Moving average of online	0.999	69.8
Moving average of online	0.99	**72.5**
Moving average of online	0.9	68.4
Stop gradient of online†	0	0.3

(a) Results for different target modes. †In the *stop gradient of online*, $\tau = \tau_{\text{base}} = 0$ is kept constant throughout training.

BYOL：結論

BYOL提出另一種自監督學習的想法，在不使用負樣本做對比的情況下，只用正樣本互相預測，也是可以達到很好的準確率，且在比較小的batch size表現上可以比之前的SimCLR框架還要好。