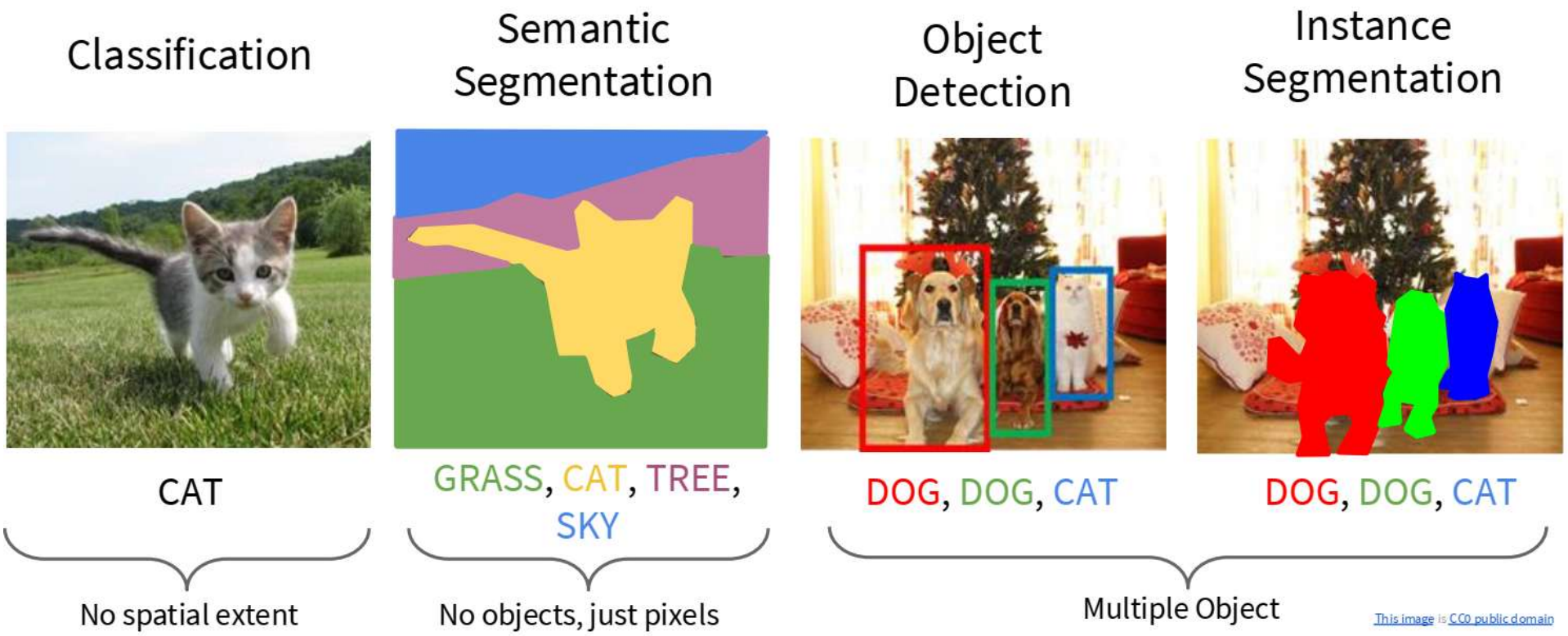
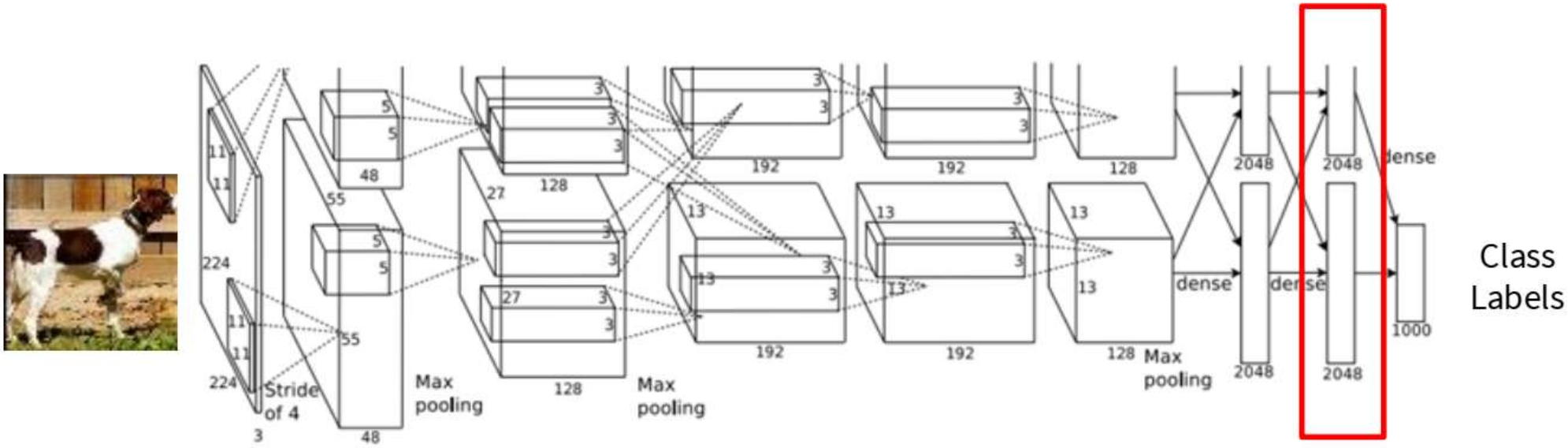
Visión Computacional

Ivan Sipiran

Muchas tareas



Representation Learning

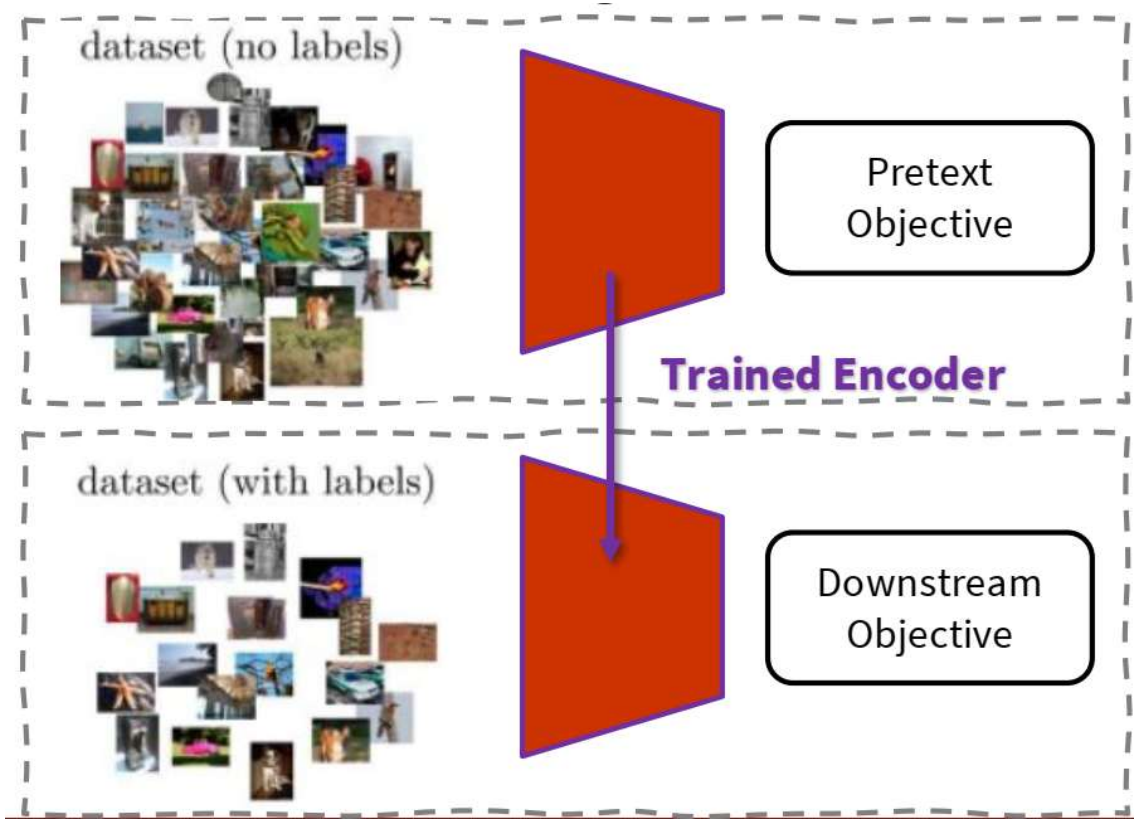


Entrenamiento de gran escala:

- Se necesita mucha data etiquetada

Se podrá entrenar una red neuronal sin data etiquetada?

Self-supervised learning



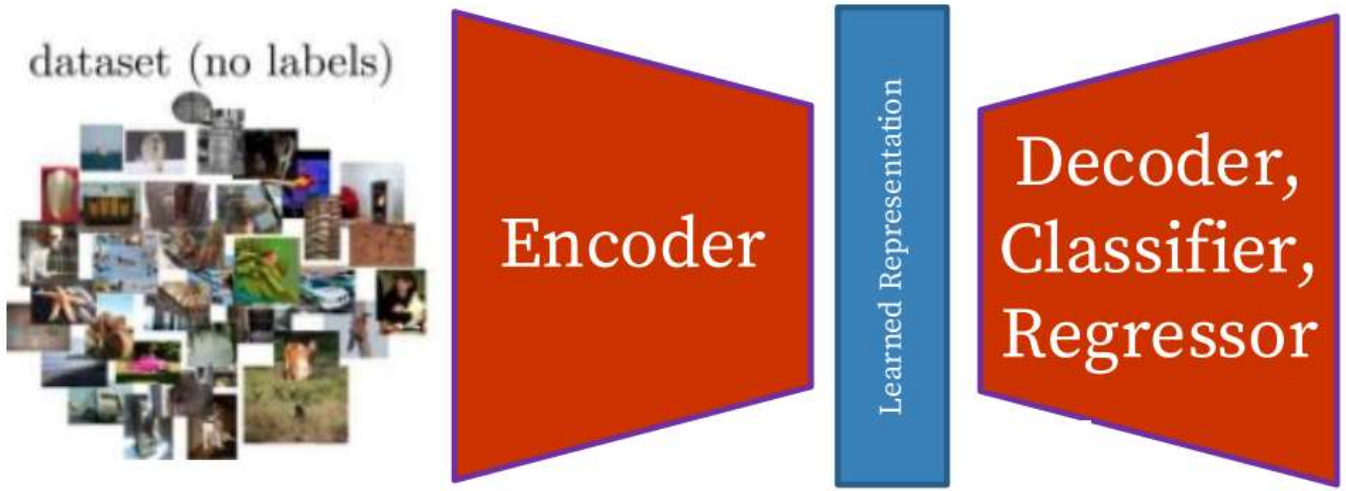
Tarea pretexto

- Definida sobre la data misma
- Sin anotaciones
- No es unsupervised del todo, algo de supervisión todavía aplica

Tarea downstream

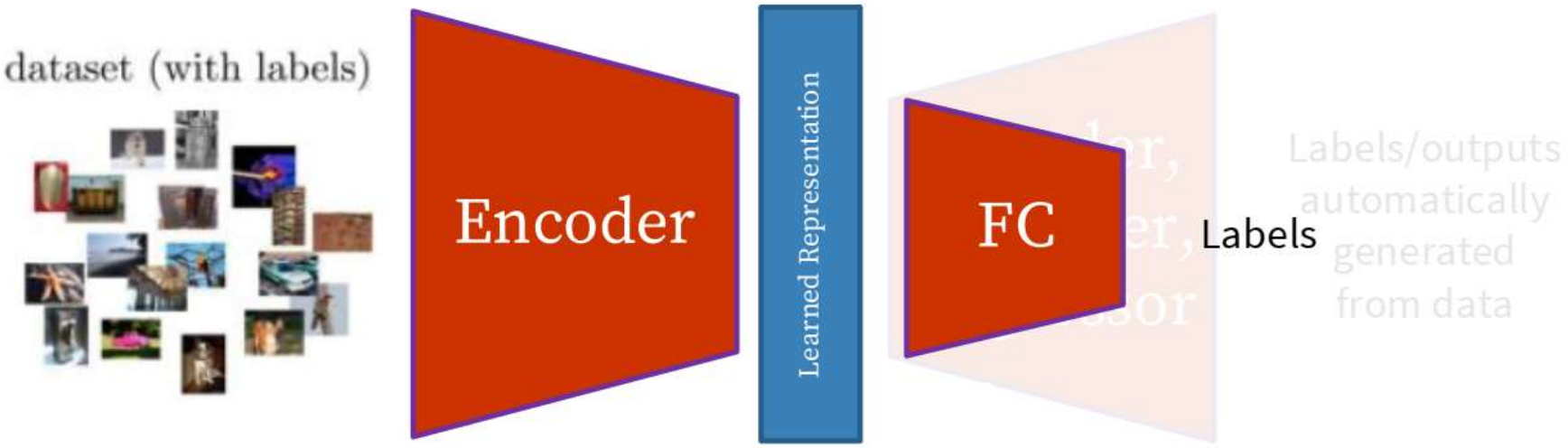
- Aplicación de interés
- No necesitas datasets grandes
- Pero si dataset etiquetado

Self-supervised learning – Tarea pretexto



Etiquetas automáticamente generadas por la data

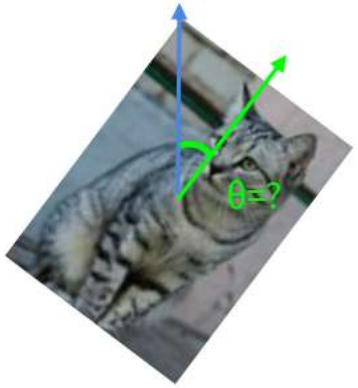
Self-supervised learning – Tarea pretexto



Tarea pretexto

Ejemplo: aprender a predecir transformaciones de imágenes / completar imágenes corruptas

image completion



rotation prediction

“jigsaw puzzle”

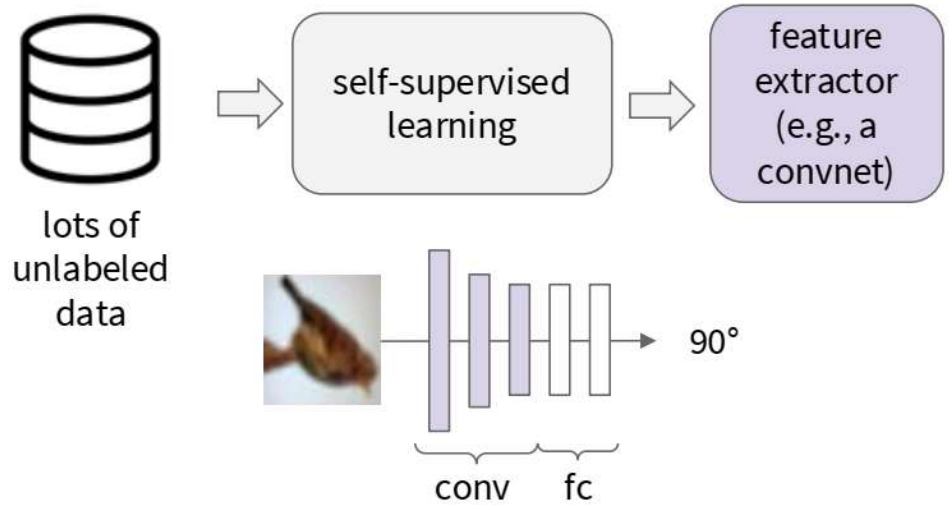
colorization

1. Resolver la tarea de pretexto permite al modelo aprender buenas features
2. Podemos generar automáticamente etiquetas para las tareas de pretexto

Cómo se evalúa?

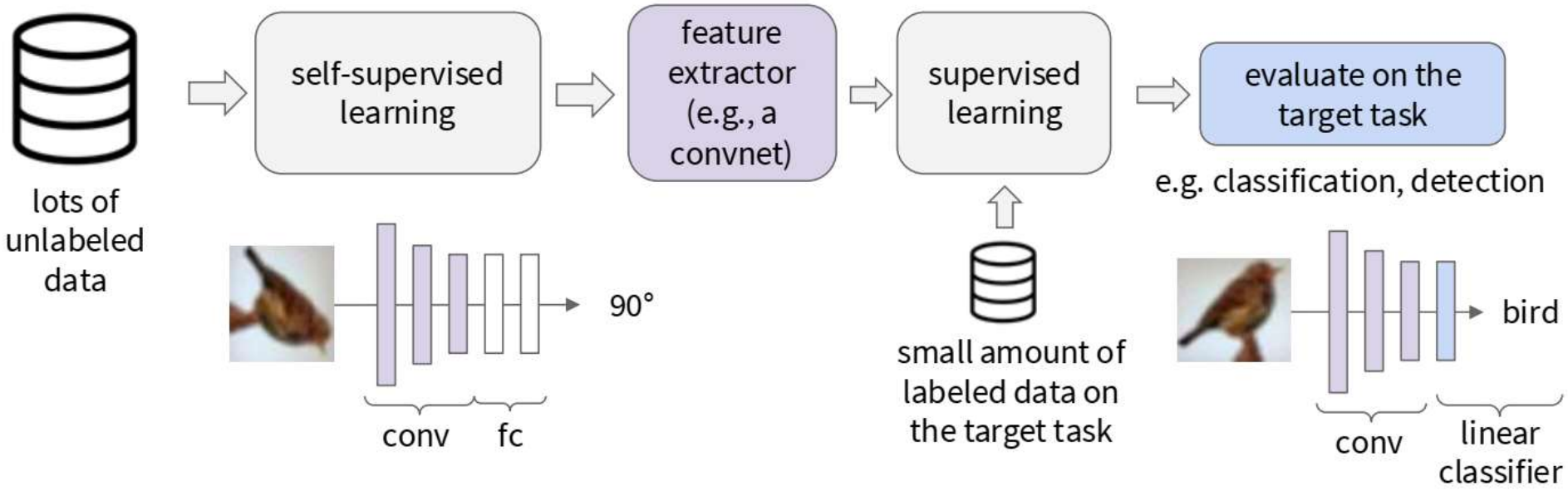
- **Performance de la tarea de pretexto**
  - Medir cuan bien el modelo funciona en la tarea en la que el modelo es entrenado
- **Calidad de la representación**
  - Evaluar la calidad de las features
    - Protocolo de evaluación lineal: entrenar un clasificador lineal sobre las features
    - Clustering: medir performance de clustering
    - T-SNE: visualizar las representaciones para observar separabilidad
- **Robustez y generalización**
  - Testear cuan bien el modelo generaliza a otros datasets
- **Eficiencia computacional**
  - Evaluar la eficiencia del método en términos de tiempo de entrenamiento y recursos usados
- **Transfer learning y performance en downstream**
  - Evaluar la utilidad de las features por transferirlas a una tarea downstream supervisada

Cómo evaluar?



1. Aprender un buen extractor de features desde tareas de pretexto. Por ejemplo: predecir la rotación de una imagen

Cómo evaluar?



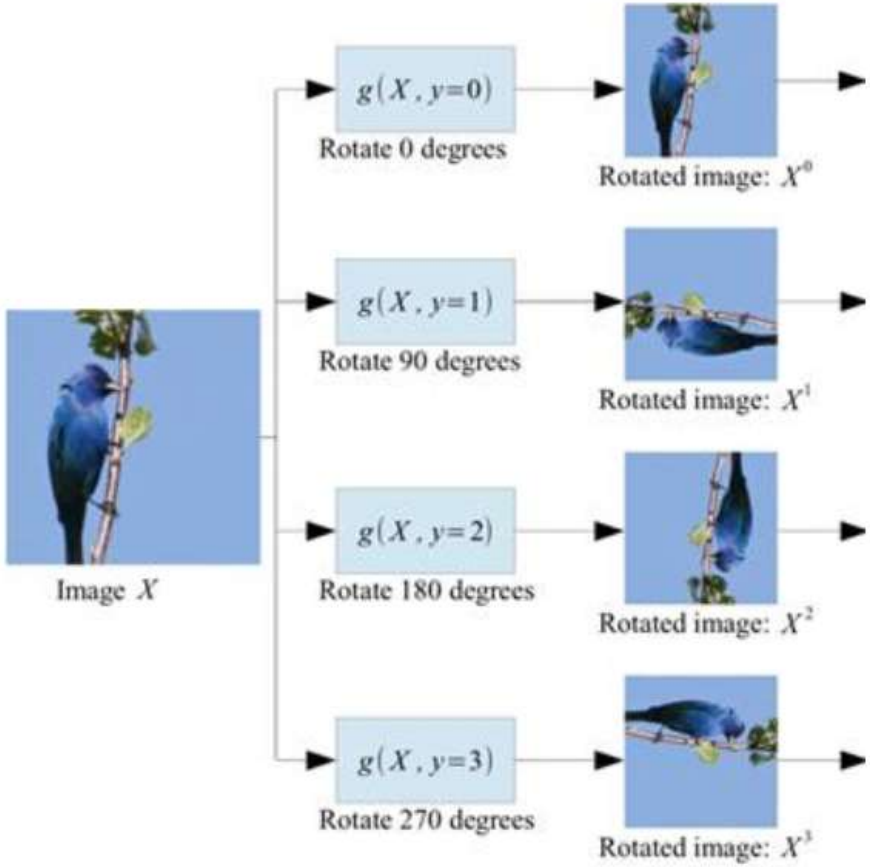
1. Aprender un buen extractor de features desde tareas de pretexto. Por ejemplo: predecir la rotación de una imagen

2. Anexar una red pequeña al feature extractor; entrenar la red en la tarea objetivo con poca data etiquetada

Tarea Pretexto: predecir rotaciones

Hipótesis: un modelo puede reconocer la rotación correcta de un objeto solo si tiene sentido común visual de cómo el objeto debería verse sin rotación.

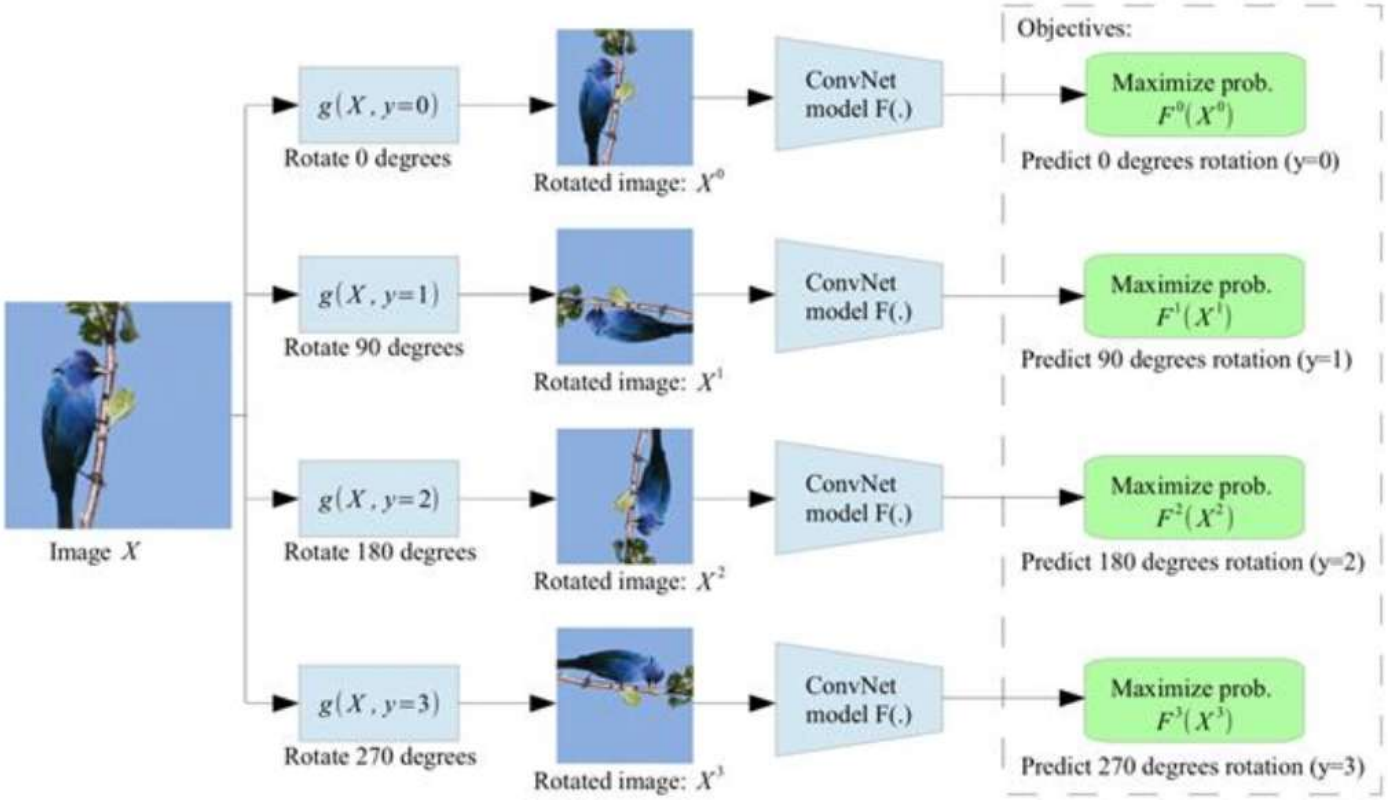
Tarea Pretexto: predecir rotaciones



Auto-supervisado por rotar las imágenes de entrada

El modelo aprende a predecir qué rotación se aplica (una clasificación de 4 clases)

Tarea Pretexto: predecir rotaciones



Auto-supervisado por rotar las imágenes de entrada

El modelo aprende a predecir qué rotación se aplica (una clasificación de 4 clases)

Tarea Pretexto: predecir rotaciones

	Classification (%mAP)		Detection (%mAP)	Segmentation (%mIoU)
Trained layers	fc6-8	all	all	all
ImageNet labels	78.9	79.9	56.8	48.0
Random		53.3	43.4	19.8
Random rescaled Krähenbühl et al. (2015)	39.2	56.6	45.6	32.6
Egomotion (Agrawal et al., 2015)	31.0	54.2	43.9	
Context Encoders (Pathak et al., 2016b)	34.6	56.5	44.5	29.7
Tracking (Wang & Gupta, 2015)	55.6	63.1	47.4	
Context (Doersch et al., 2015)	55.1	65.3	51.1	
Colorization (Zhang et al., 2016a)	61.5	65.6	46.9	35.6
BIGAN (Donahue et al., 2016)	52.3	60.1	46.9	34.9
Jigsaw Puzzles (Noroozi & Favaro, 2016)	-	67.6	53.2	37.6
NAT (Bojanowski & Joulin, 2017)	56.7	65.3	49.4	
Split-Brain (Zhang et al., 2016b)	63.0	67.1	46.7	36.0
ColorProxy (Larsson et al., 2017)		65.9		38.4
Counting (Noroozi et al., 2017)	-	67.7	51.4	36.6
(Ours) RotNet	**70.87**	**72.97**	**54.4**	**39.1**

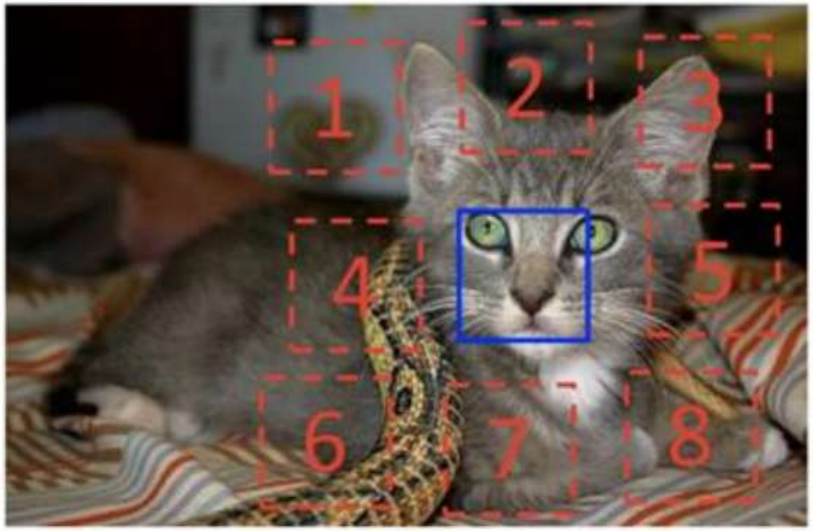
Pretrained with full ImageNet supervision

No pretraining

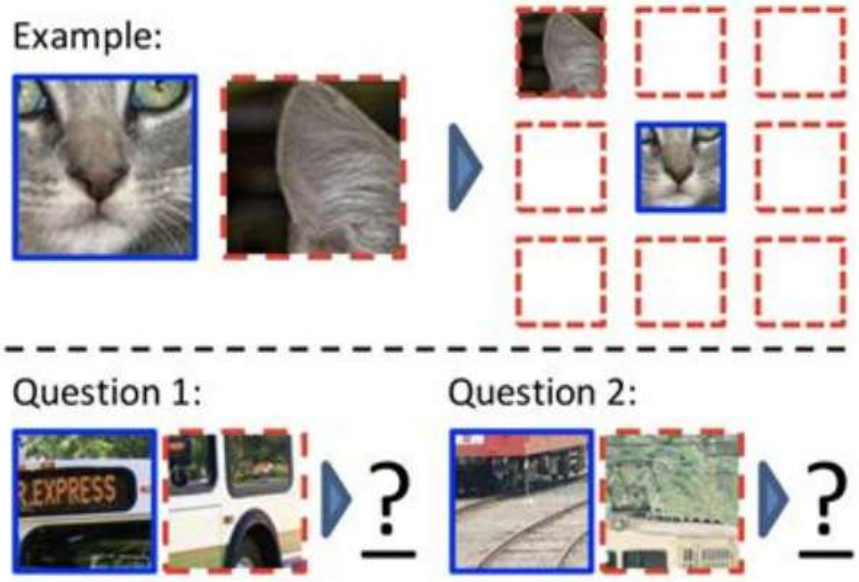
Self-supervised learning on ImageNet (entire training set) with AlexNet.

Finetune on labeled data from Pascal VOC 2007.

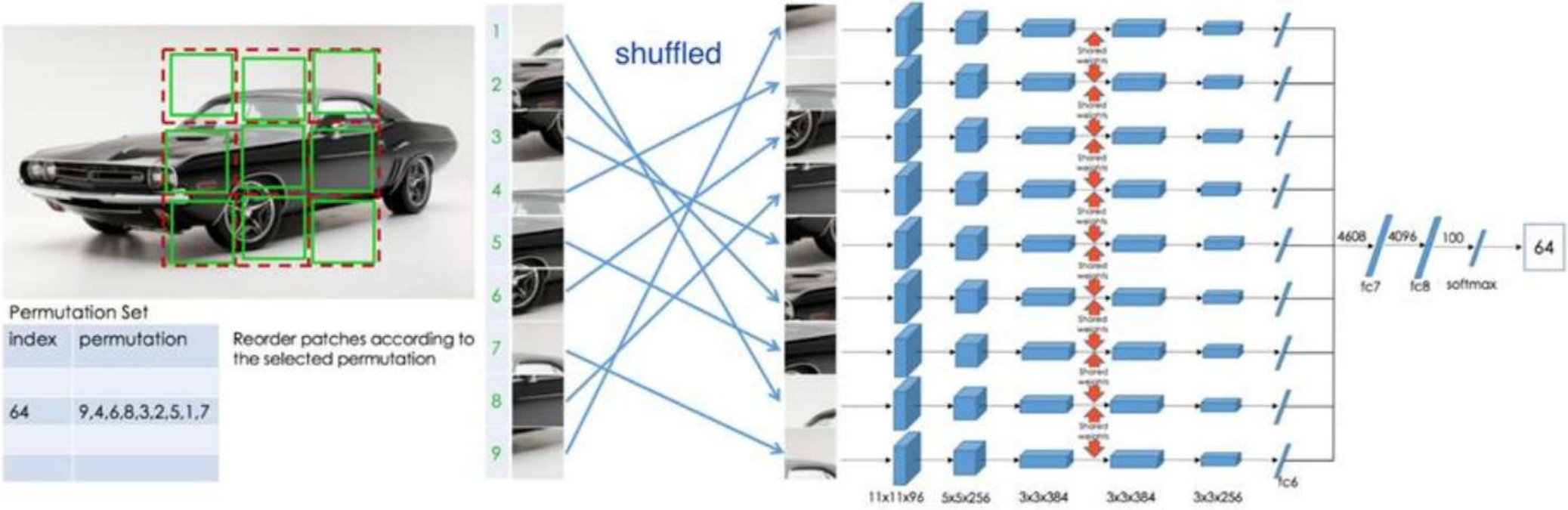
Tarea Pretexto: predecir ubicaciones de parches



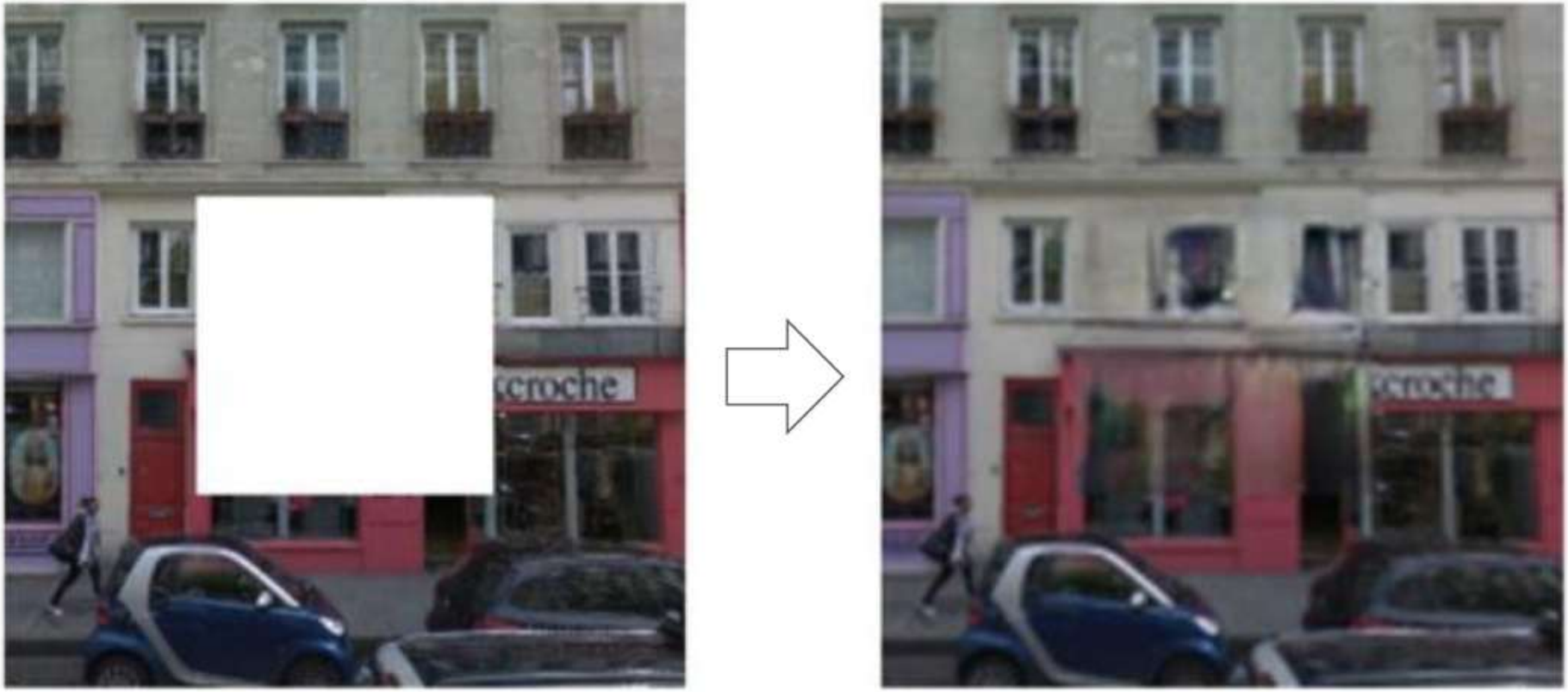
X = (,); Y = 3



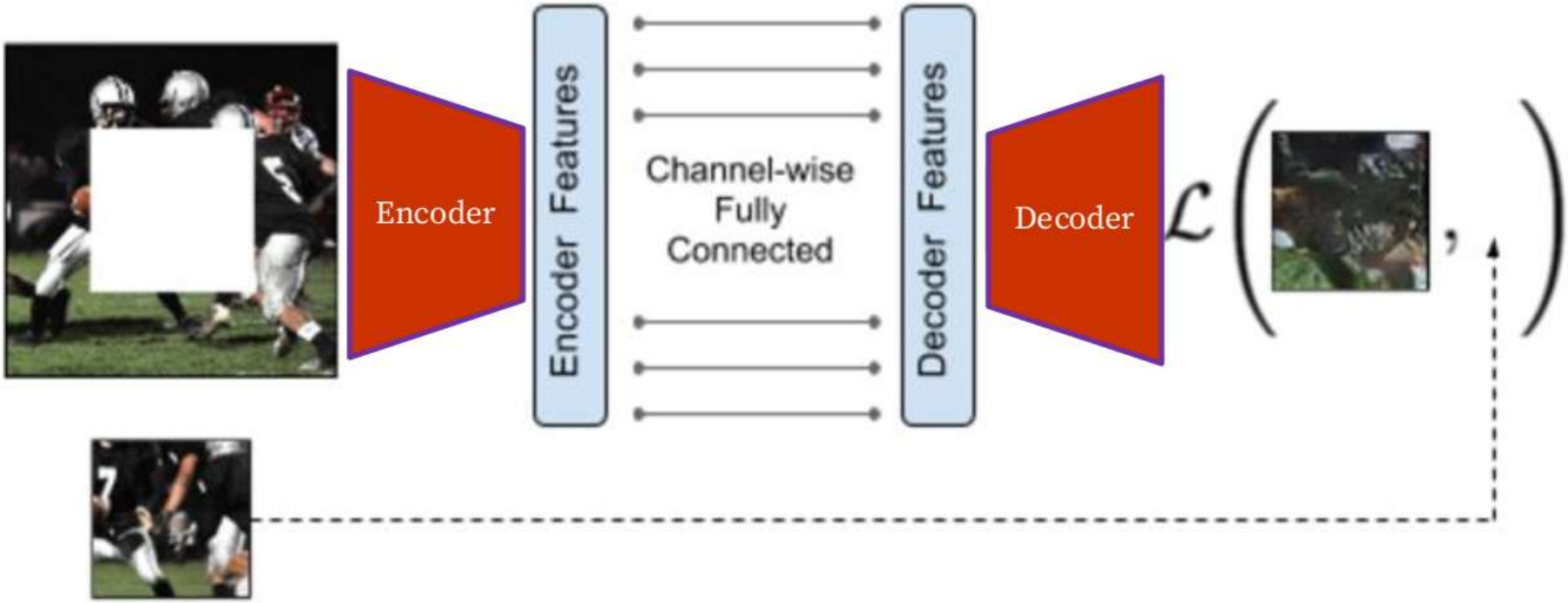
Tarea Pretexto: resolver rompecabezas



Tarea Pretexto: predecir pixels faltantes



Tarea Pretexto: inpainting por reconstrucción



Tarea Pretexto: colorization

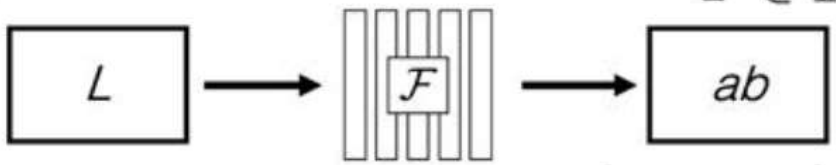
$\mathcal{F}$

Grayscale image: *L* channel

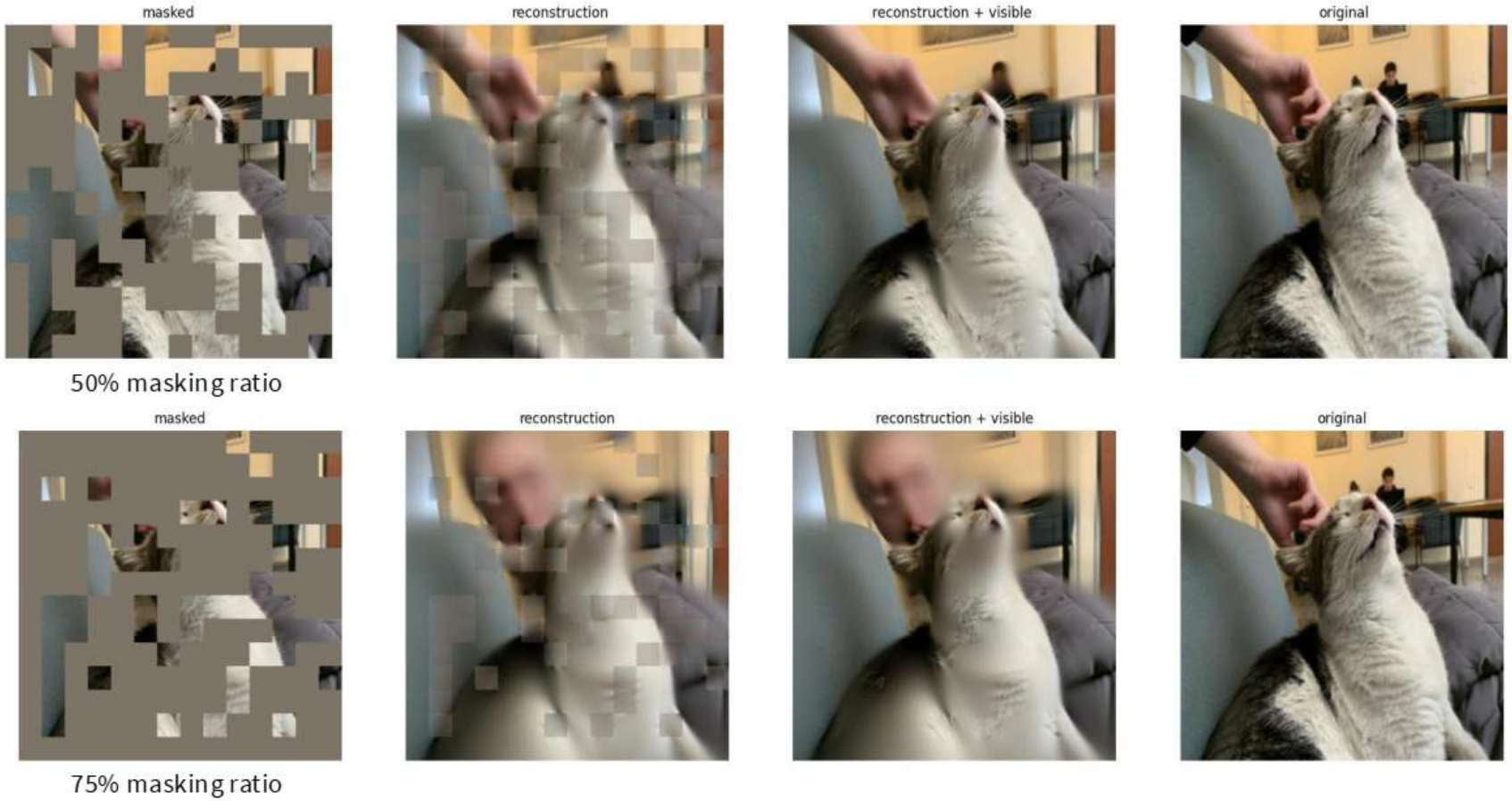
$\mathbf{X} \in \mathbb{R}^{H \times W \times 1}$

Color information: *ab* channels

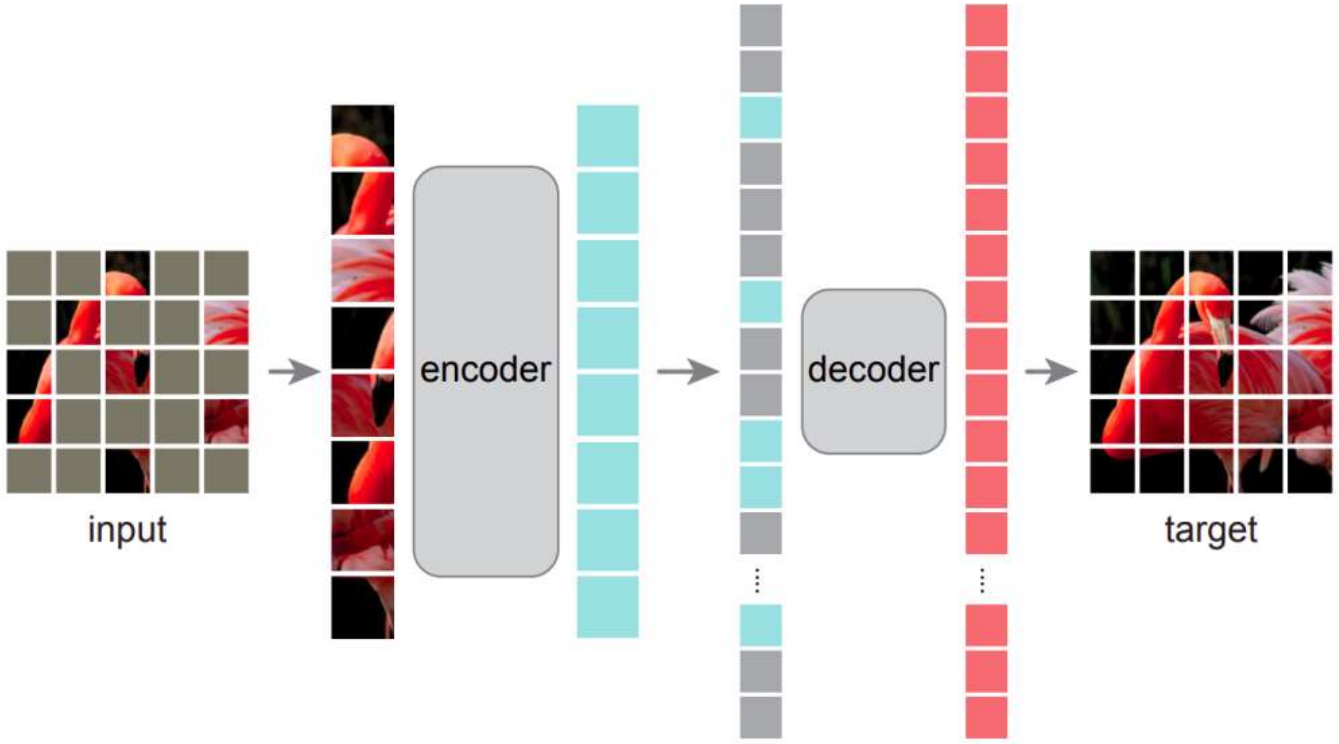
$\widehat{\mathbf{Y}} \in \mathbb{R}^{H \times W \times 2}$



Masked Auto-encoders



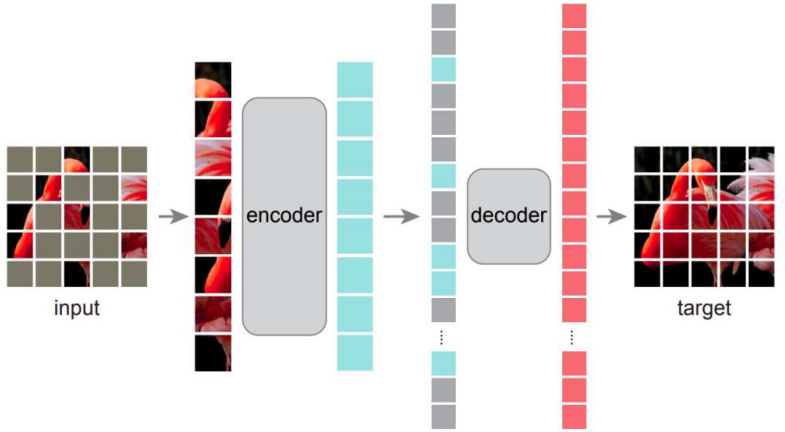
Masked Auto-encoders



He et al., 2021 Masked Autoencoders
Are Scalable Vision Learners

Masked Auto-encoders

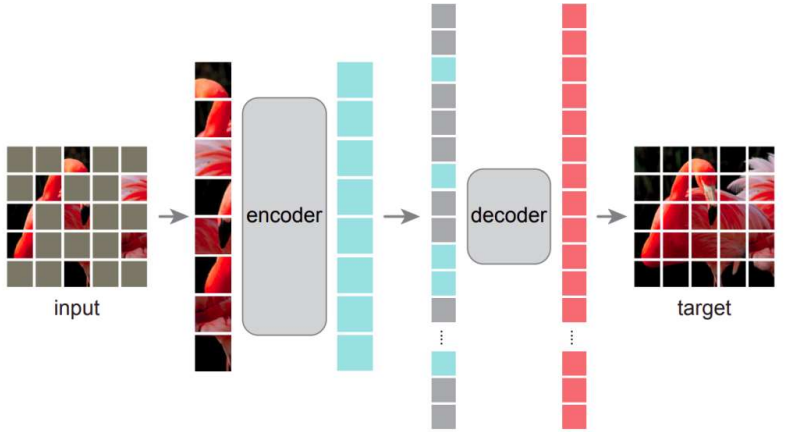
- Similar al ViT original, dividir la entrada en parches sin overlap
- Uniformemente samplear un porcentaje alto (75%) de parches y enmascarar
- Usar un porcentaje alto hace que la tarea sea difícil y desafiante
- El encoder tiene que ser muy grande



He et al., 2021 Masked Autoencoders Are Scalable Vision Learners

MAE Encoder

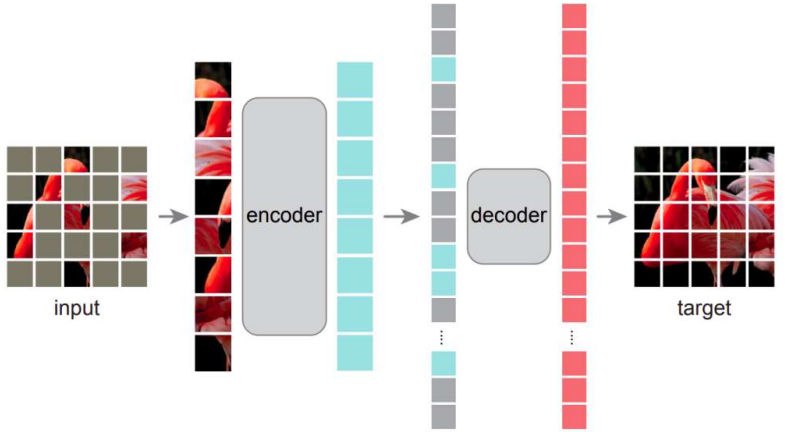
- El encoder solo opera sobre parches sin máscara
- Convierte los parches con linear embedding y positional encoding
- Usa bloques Transformer
- Desde que los parches de entrada son pocos, se usa un encoder muy grande



He et al., 2021 Masked Autoencoders Are Scalable Vision Learners

MAE Decoder

- Combina la salida del encoder con los tokens enmascarados, agregando positional encodings
- Usa bloques Transformer, seguidos por proyección lineal que termina en reconstrucción de pixels.
- Solo se encarga de la reconstrucción, por lo que no se usa después del entrenamiento.
- Diseño de autoencoder asimétrico



He et al., 2021 Masked Autoencoders Are Scalable Vision Learners

Reconstrucción

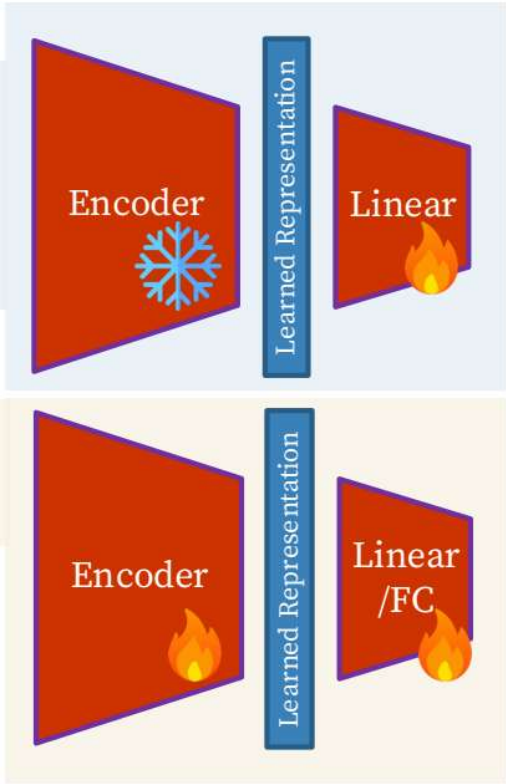
- MSE (mean squared error) en el espacio de pixels entre la imagen de entrada y la imagen reconstruida
- Loss solo se computa para parches enmascarados

Linear Probing vs Full fine-tuning

- En linear probing, el modelo pre-entrenado se congela y solo se agrega una capa lineal para predecir etiquetas. El método es usado para evaluar la calidad de las representaciones del modelo de extracción de features.

- En finetuning, el modelo pre-entrenado se entrena junto con una o más capas agregadas, posiblemente con no-linealidades.

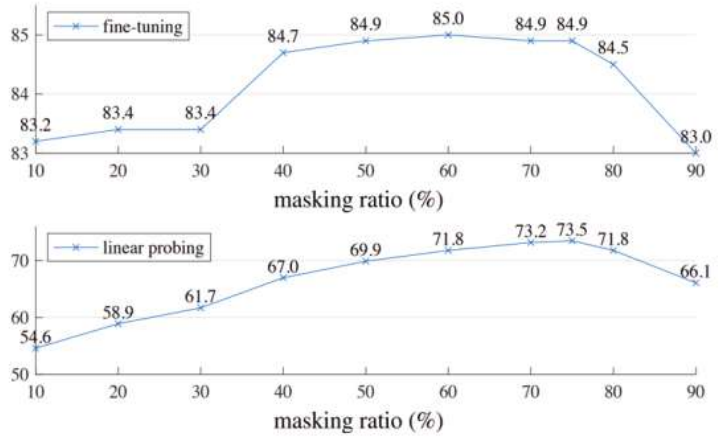
- LP: provee una medida de calidad de la representación
- FT: explota el modelo para adoptar nuevas tareas



Ablation

Muchas decisiones de hiperparámetros

- Porcentaje de masking
- Prof. De encoder
- Prof. De decoder
- Uso de mask token
- Target de reconstrucción
- Data augmentation
- Método de sampling
- Schedule de training



case	ratio	ft	lin
random	75	**84.9**	**73.5**
block	50	83.9	72.3
block	75	82.8	63.9
grid	75	84.0	66.0

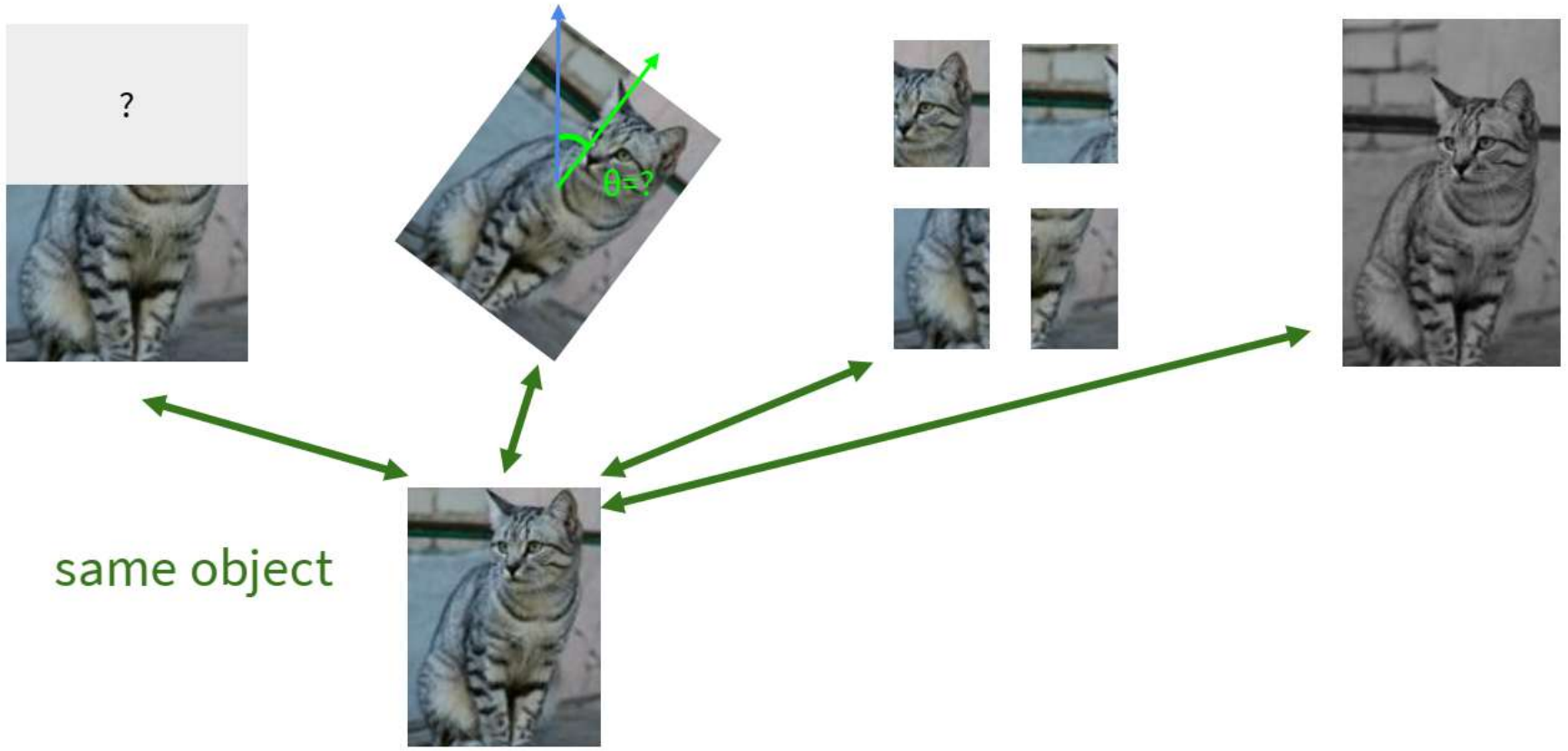
MAE - Comparación

method	pre-train data	ViT-B	ViT-L	ViT-H	ViT-H_{448}
scratch, our impl.	-	82.3	82.6	83.1	-
DINO [5]	IN1K	82.8	-	-	-
MoCo v3 [9]	IN1K	83.2	84.1	-	-
BEiT [2]	IN1K+DALLE	83.2	85.2	-	-
MAE	IN1K	83.6	85.9	86.9	**87.8**

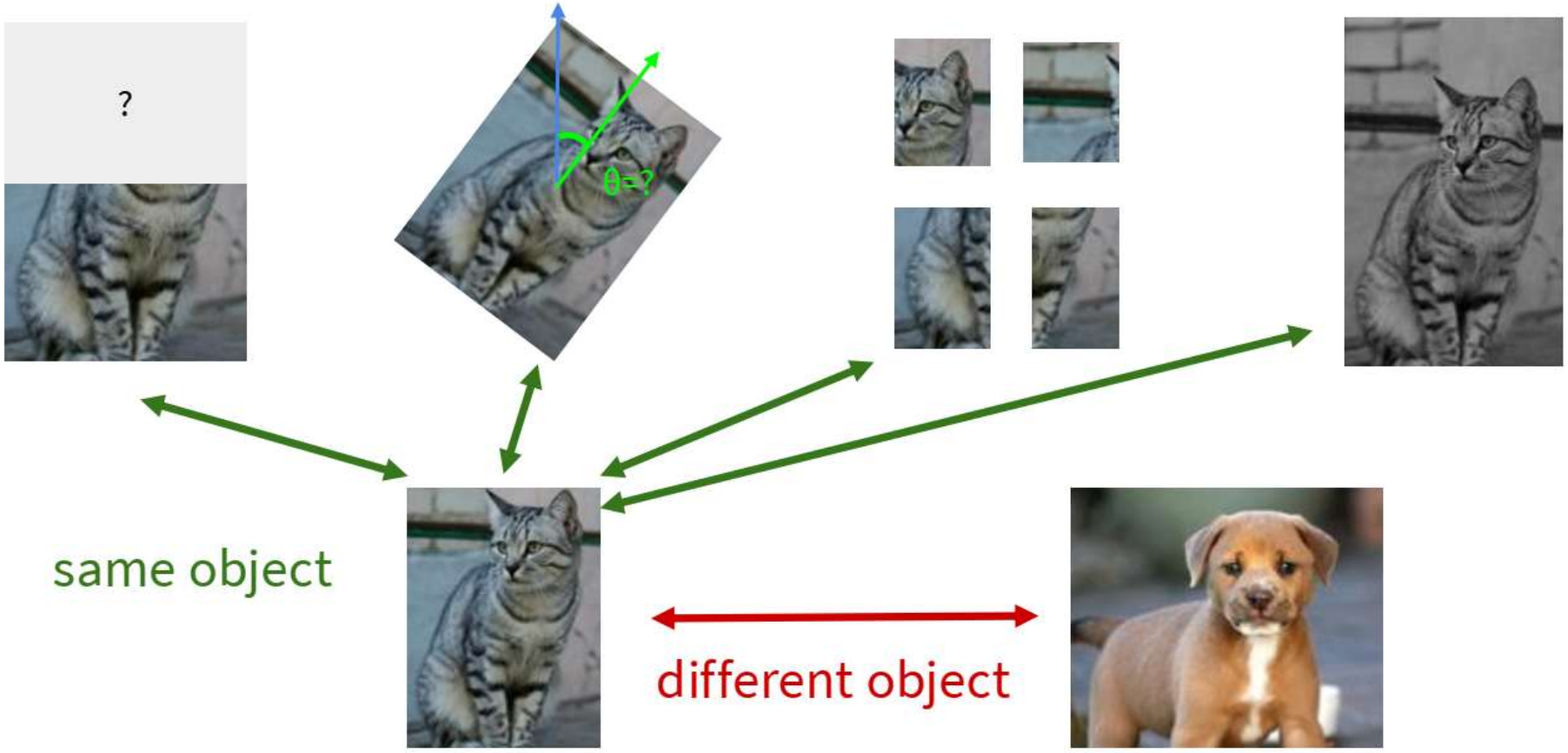
Tarea Pretexto

- Se enfoca en el sentido común visual: rotaciones, inpainting, colorización, etc
- No se toma en cuenta el performance de las tareas de pretexto, sino cuán útiles son las features
- Problemas:
  - Pensar en tareas de pretexto individuales es tedioso
  - Feaures podrían no ser generales

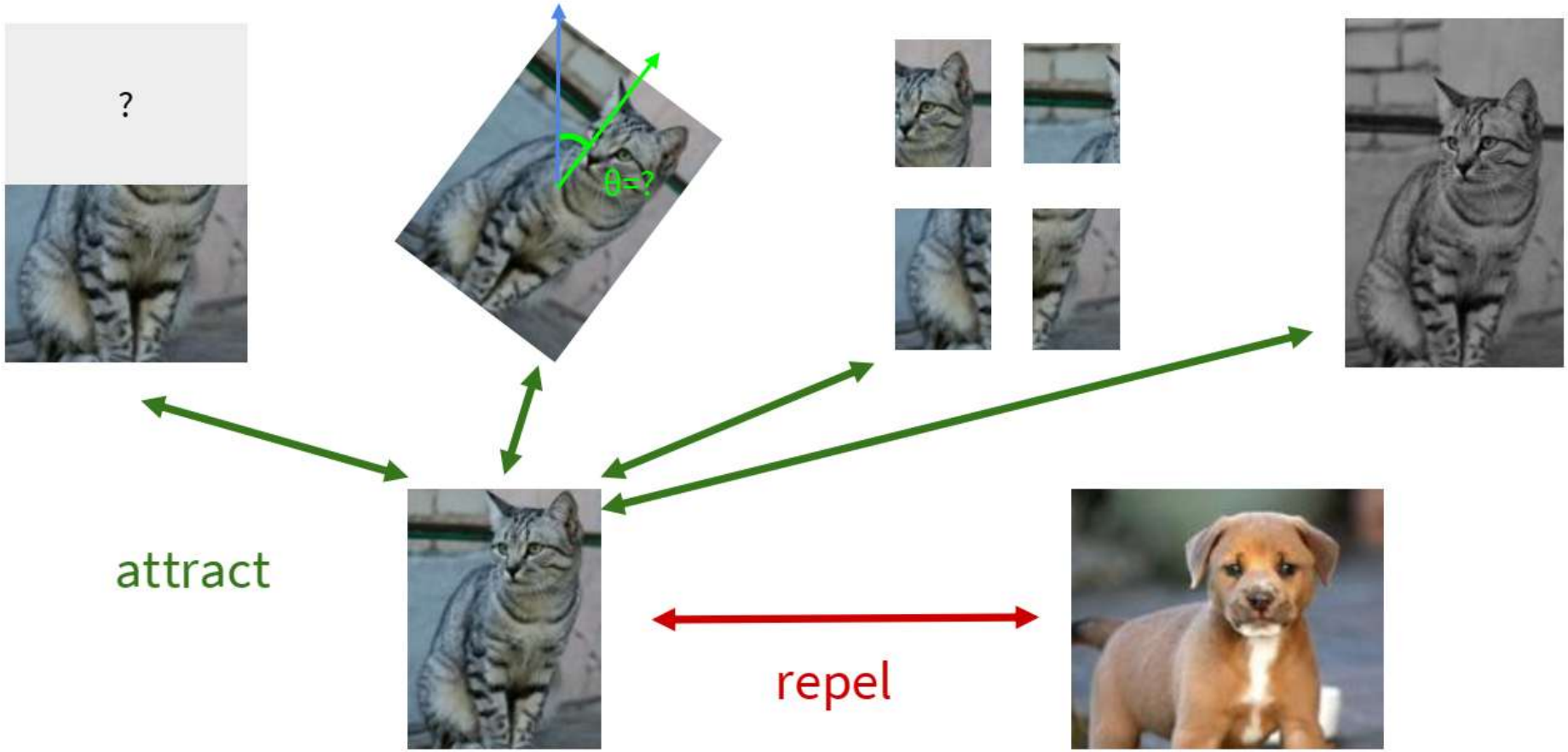
Tarea de pretexto más general



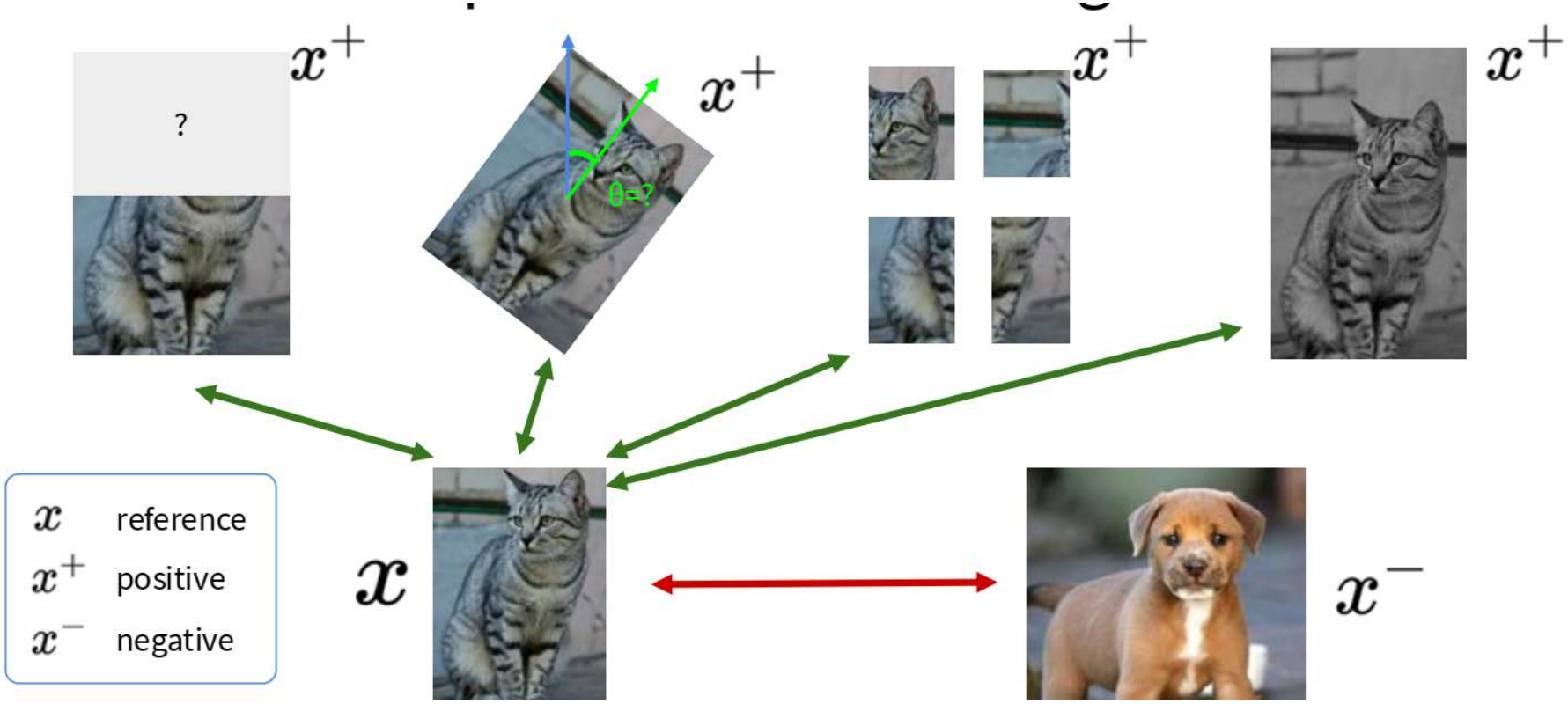
Tarea de pretexto más general



Aprendizaje contrastivo



Aprendizaje contrastivo



Formulación de aprendizaje contrastivo

Lo que buscamos

$$\text{score}(f(x), f(x^+)) >> \text{score}(f(x), f(x^-))$$

Dada una función de score, queremos aprender una función encoder que compute un alto score para pares positivos y bajo score para pares negativos.

Formulación de aprendizaje contrastivo

Función de Loss dado un sample positivo y N-1 simples negativos

$$L = -\mathbb{E}_X \left[\log \frac{\exp(s(f(x), f(x^+))}{\exp(s(f(x), f(x^+)) + \sum_{j=1}^{N-1} \exp(s(f(x), f(x_j^-))} \right]$$

Formulación de aprendizaje contrastivo

Función de Loss dado un sample positivo y N-1 simples negativos

$$L = -\mathbb{E}_X \left[\log \frac{\exp(s(f(x), f(x^+))}{\exp(s(f(x), f(x^+)) + \sum_{j=1}^{N-1} \exp(s(f(x), f(x_j^-))} \right]$$

x

x^+

x

x_1^-

x_2^-

x_3^-

...

Formulación de aprendizaje contrastivo

Función de Loss dado un sample positivo y N-1 simples negativos

$$L = -\mathbb{E}_X \left[\log \frac{\exp(s(f(x), f(x^+))}{\exp(s(f(x), f(x^+)) + \sum_{j=1}^{N-1} \exp(s(f(x), f(x_j^-))} \right]$$

score for the positive pair

score for the N-1 negative pairs

Formulación de aprendizaje contrastivo

Función de Loss dado un sample positivo y N-1 simples negativos

$$L = -\mathbb{E}_X \left[\log \frac{\exp(s(f(x), f(x^+))}{\exp(s(f(x), f(x^+)) + \sum_{j=1}^{N-1} \exp(s(f(x), f(x_j^-))} \right]$$

score for the positive pair

score for the N-1 negative pairs

Es Cross-entropy Loss para un clasificador de N clases!!

Aprende a encontrar el sample positivo desde los N samples!!

Formulación de aprendizaje contrastivo

Función de Loss dado un sample positivo y N-1 simples negativos

$$L = -\mathbb{E}_X \left[\log \frac{\exp(s(f(x), f(x^+))}{\exp(s(f(x), f(x^+)) + \sum_{j=1}^{N-1} \exp(s(f(x), f(x_j^-))} \right]$$

Conocida como la InfoNCE los

Una cota inferior sobre la información mutual entre f(x) y f(x+)

$$MI[f(x), f(x^+)] - \log(N) \geq -L$$

Mientras más grande N, más justa la cota

SimCLR: Simple aprendizaje contrastivo

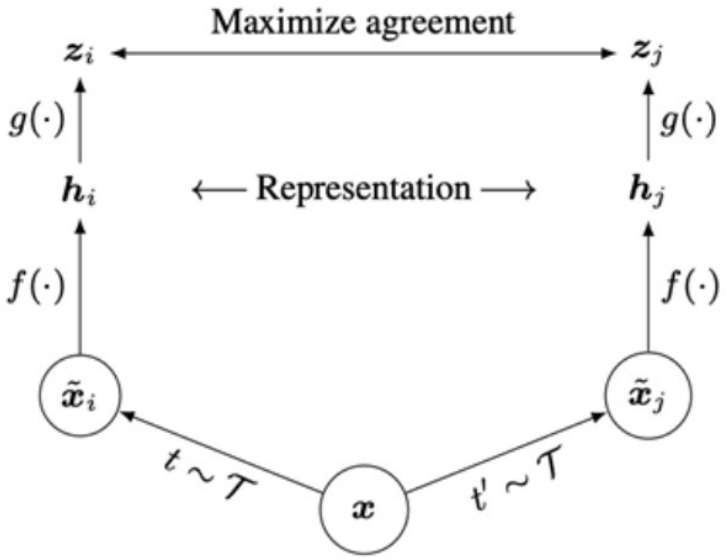
Similitud coseno como función de score

$$s(u, v) = \frac{u^T v}{||u||||v||}$$

Usar una red de proyección $g(\cdot)$ para proyectar features a un espacio en donde el aprendizaje contrastivo ocurre

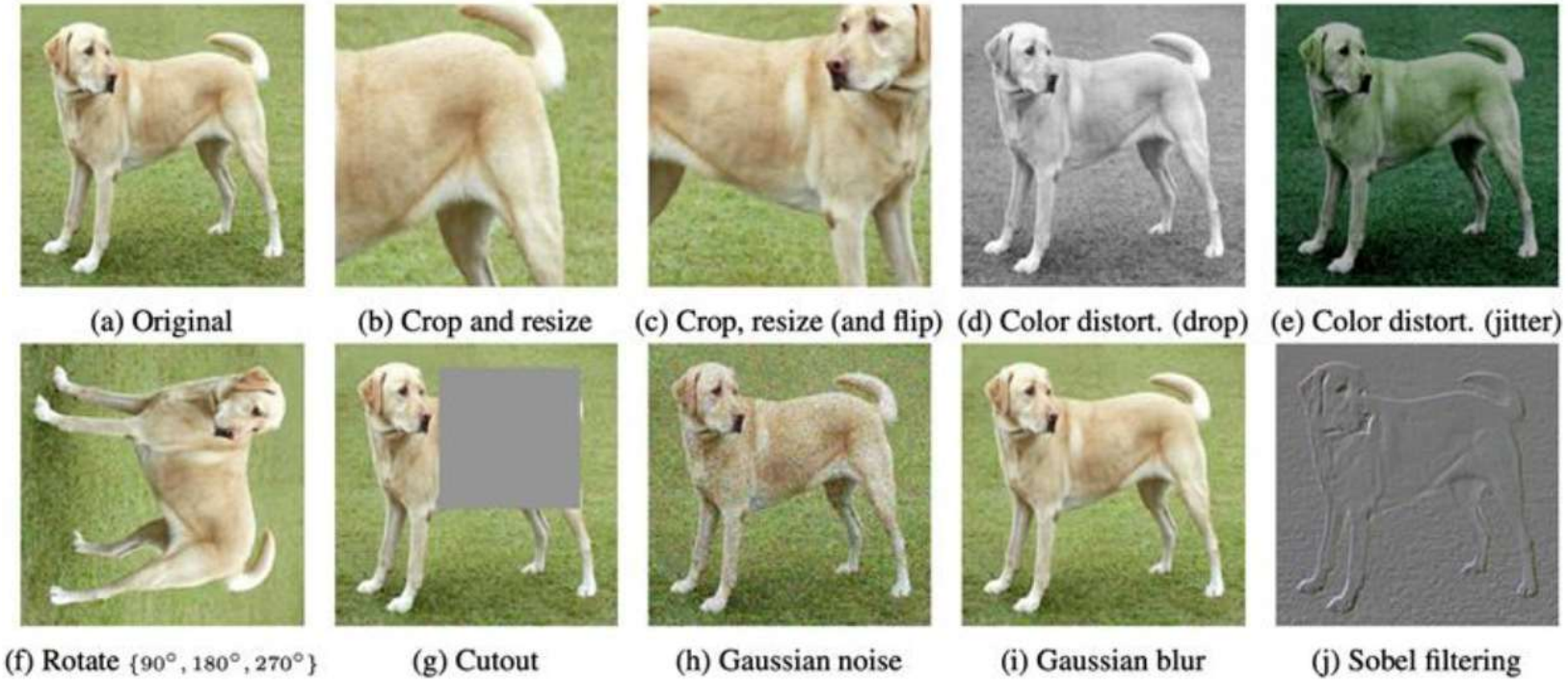
Generar samples positivos a través de data augmentation

- Random cropping, random color distortion, random blur



SimCLR: Simple aprendizaje contrastivo

Generar samples positivos desde data augmentation

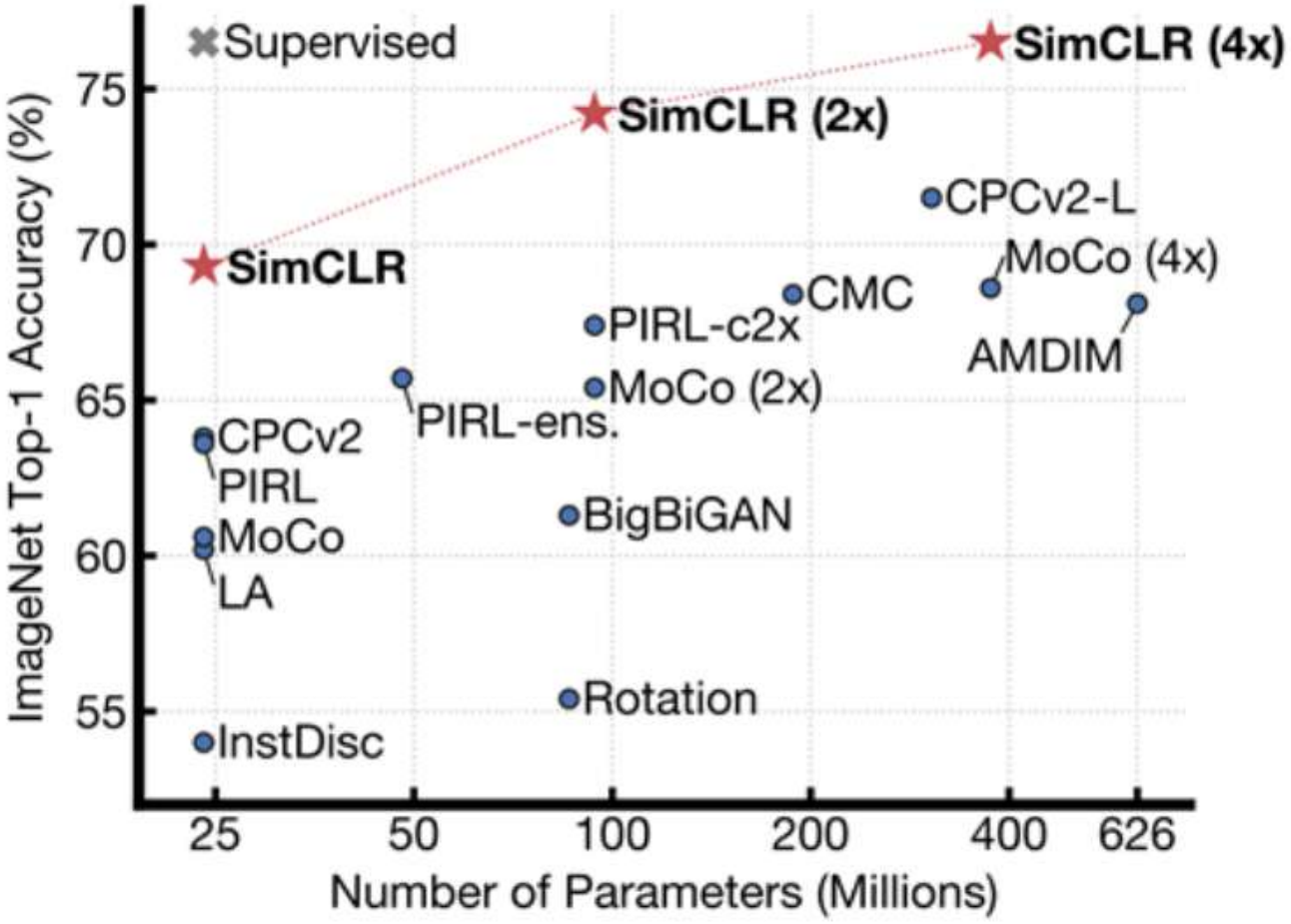


SimCLR

Generate a positive pair by sampling data augmentation functions

Algorithm 1 SimCLR's main learning algorithm.

input: batch size N, constant τ, structure of $f, g, \mathcal{T}$.
for sampled minibatch $\{\boldsymbol{x}_k\}_{k=1}^{N}$ **do**
  for all $k \in \{1, \ldots, N\}$ **do**
    draw two augmentation functions $t \sim \mathcal{T}, t' \sim \mathcal{T}$
    # the first augmentation
    $\tilde{\boldsymbol{x}}_{2k-1} = t(\boldsymbol{x}_k)$
    $\boldsymbol{h}_{2k-1} = f(\tilde{\boldsymbol{x}}_{2k-1})$ # representation
    $\boldsymbol{z}_{2k-1} = g(\boldsymbol{h}_{2k-1})$ # projection
    # the second augmentation
    $\tilde{\boldsymbol{x}}_{2k} = t'(\boldsymbol{x}_k)$
    $\boldsymbol{h}_{2k} = f(\tilde{\boldsymbol{x}}_{2k})$ # representation
    $\boldsymbol{z}_{2k} = g(\boldsymbol{h}_{2k})$ # projection
  end for
  for all $i \in \{1, \ldots, 2N\}$ and $j \in \{1, \ldots, 2N\}$ **do**
    $s_{i,j} = \boldsymbol{z}_i^\top \boldsymbol{z}_j / (\|\boldsymbol{z}_i\| \|\boldsymbol{z}_j\|)$ # pairwise similarity
  end for
  define $\ell(i, j)$ **as** $\ell(i, j) = -\log \frac{\exp(s_{i,j}/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(s_{i,k}/\tau)}$
  $\mathcal{L} = \frac{1}{2N} \sum_{k=1}^{N} [\ell(2k-1, 2k) + \ell(2k, 2k-1)]$
  update networks f and g to minimize $\mathcal{L}$
end for
return encoder network $f(\cdot)$, and throw away $g(\cdot)$



Entrenar el encoder sobre ImageNet

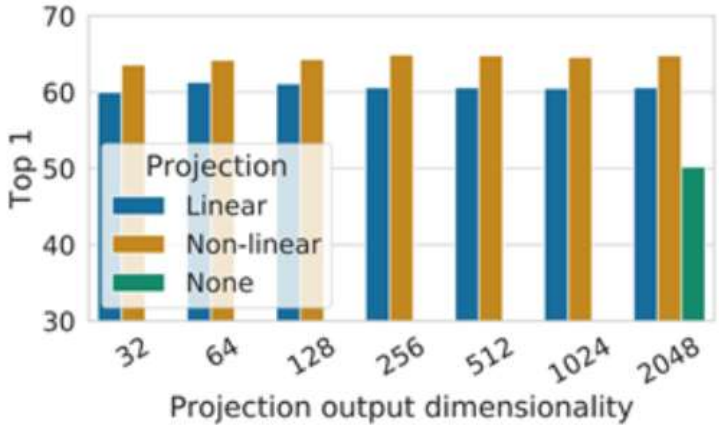
Congelar el feature encoder, entrenar un clasificador lineal con data etiquetada

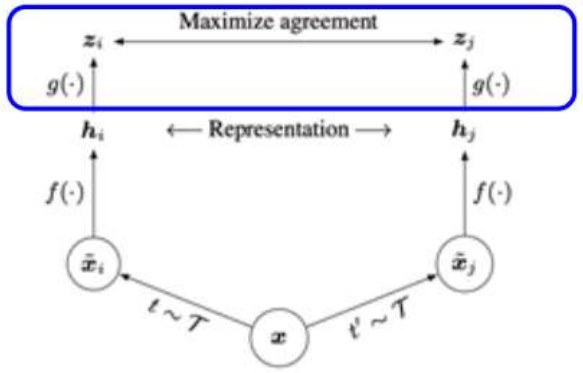
Method	Architecture	Label fraction (Top 5) 1%	10%
Supervised baseline	ResNet-50	48.4	80.4
Methods using other label-propagation:			
Pseudo-label	ResNet-50	51.6	82.4
VAT+Entropy Min.	ResNet-50	47.0	83.4
UDA (w. RandAug)	ResNet-50	-	88.5
FixMatch (w. RandAug)	ResNet-50	-	89.1
S4L (Rot+VAT+En. M.)	ResNet-50 (4×)	-	91.2
Methods using representation learning only:			
InstDisc	ResNet-50	39.2	77.4
BigBiGAN	RevNet-50 (4×)	55.2	78.8
PIRL	ResNet-50	57.2	83.8
CPC v2	ResNet-161(*)	77.9	91.2
SimCLR (ours)	ResNet-50	75.5	87.8
SimCLR (ours)	ResNet-50 (2×)	83.0	91.2
SimCLR (ours)	ResNet-50 (4×)	**85.8**	**92.6**

Entrenar el encoder sobre ImageNet

Finetuning con 1%/10% de data etiquetada.

SimCLR: diseño



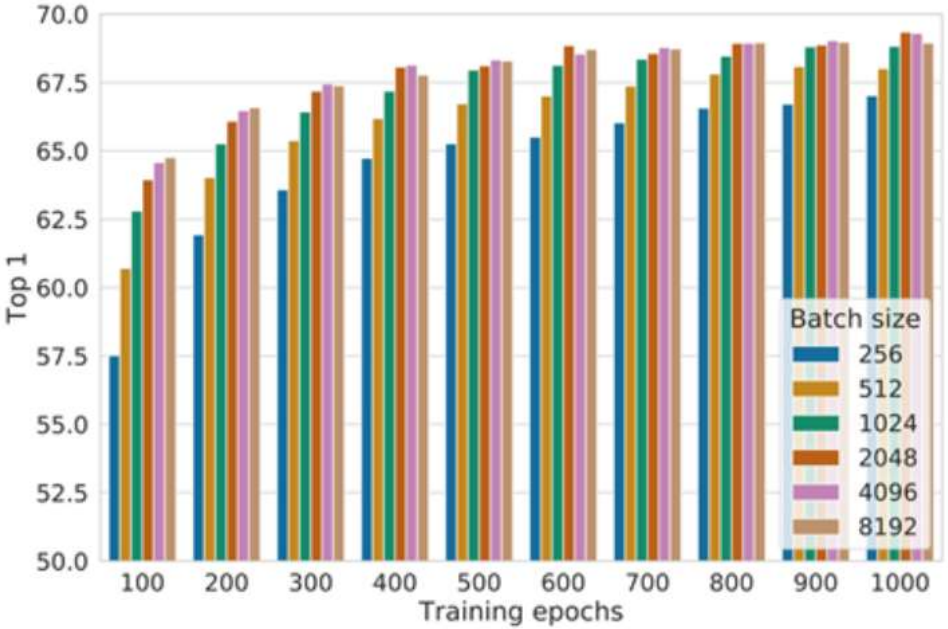


Proyección lineal/no lineal mejoran el aprendizaje

Explicación:

- Objetivo contrastivo puede descartar información útil para tareas downstream
- Espacio z es invariante a transformaciones

SimCLR: diseño



Batch size grande es crucial!

Requiere entrenamiento distribuido en TPU