# Impacto del preentrenamiento en la detección de objetos semi-supervisada

Pablo Reina-Jiménez[1], Ana Rodríguez-López[1], Aitor Rodríguez-Dueñas[1], and Manuel Carranza-García[1]

Departamento de Lenguajes y Sistemas Informáticos, Universidad de Sevilla
Av. Reina Mercedes, s/n, Sevilla, 41012, España
mcarranzag@us.es

**Resumen** La detección de objetos en vehículos autónomos es un problema crítico que requiere grandes cantidades de datos etiquetados para entrenar modelos supervisados de manera efectiva. Los enfoques semi-supervisados han demostrado ser una alternativa prometedora al aprovechar datos no etiquetados durante el entrenamiento. Sin embargo, la calidad del punto de partida del extractor de características puede influir significativamente en el rendimiento final del sistema. En este trabajo, investigamos el impacto de distintos métodos de preentrenamiento self-supervised —SwAV, MoCo, y BYOL— como inicialización del backbone ResNet-50 en el modelo de detección semi-supervisado SoftTeacher con Faster R-CNN, evaluado sobre el conjunto de datos real de Waymo. Los experimentos se realizan con un 1 % y un 10 % de datos etiquetados, y con 30k y 180k iteraciones de entrenamiento. Los resultados muestran que el preentrenamiento self-supervised mejora consistentemente el rendimiento respecto a la inicialización supervisada estándar, siendo MoCo v3 y SwAV los métodos con mejor desempeño global. No obstante, las mejoras son más moderadas en la clase de ciclistas, la más minoritaria del conjunto de datos, lo que pone de manifiesto limitaciones persistentes frente al desbalanceo de clases.

**Keywords:** aprendizaje semi-supervisado · aprendizaje self-supervised · detección de objetos · conducción autónoma

## 1. Introducción

En los últimos años, el desarrollo de los sistemas de conducción autónoma ha impulsado una intensa actividad investigadora en el campo de la percepción visual [7]. Estos sistemas deben ser capaces de interpretar su entorno en tiempo real a partir de múltiples fuentes de información, entre las que la visión por computador desempeña un papel central. En concreto, la detección de objetos es un componente crítico de la pila de percepción en cualquier plataforma de conducción autónoma [11].

Los modelos basados en redes neuronales profundas han alcanzado resultados sobresalientes en esta tarea, pero su eficacia está supeditada a la disponibilidad

de grandes volúmenes de imágenes con anotaciones precisas de cuadros delimitadores [19]. Obtener estas anotaciones en el dominio de la conducción autónoma es particularmente exigente ya que requiere la participación de operadores especializados, la gestión de objetos a múltiples escalas y la cobertura de condiciones meteorológicas y lumínicas diversas. Todo ello convierte el proceso de etiquetado en un cuello de botella que dificulta el desarrollo y la actualización continua de los sistemas [26].

Los métodos semi-supervisados surgen como respuesta directa a esta limitación, al aprovechar la abundancia de datos no etiquetados para complementar el aprendizaje a partir de un conjunto reducido de imágenes anotadas. En un trabajo previo [5] evaluamos el modelo SoftTeacher [27] sobre datos reales de Waymo [24], demostrando que con tan solo un 2 % de datos etiquetados es posible acercarse al rendimiento de un modelo completamente supervisado. No obstante, en dicho trabajo se empleó la inicialización estándar de ImageNet supervisada para el backbone ResNet-50.

Paralelamente, el aprendizaje self-supervised ha experimentado un notable avance en los últimos años, dando lugar a métodos capaces de aprender representaciones visuales ricas sin necesidad de anotaciones [15]. Estos métodos obtienen backbones preentrenados que, en muchos casos, superan a la inicialización supervisada estándar en tareas de transferencia. Sin embargo, su impacto sobre la detección semi-supervisada en el dominio específico de los vehículos autónomos no ha sido explorado en profundidad.

En este trabajo investigamos si el preentrenamiento self-supervised del backbone puede mejorar el rendimiento del sistema semi-supervisado de detección de objetos en vehículos autónomos. Específicamente, evaluamos cuatro métodos de preentrenamiento self-supervised —SwAV [3], MoCo v2 [8], MoCo v3 [9] y BYOL [14]— como alternativa a la inicialización supervisada estándar de torchvision [25] en el backbone ResNet-50 del modelo SoftTeacher. Las contribuciones principales de este trabajo son:

- Evaluar distintos métodos de preentrenamiento self-supervised como inicialización del backbone en detección semi-supervisada para vehículos autónomos.
- Analizar su efecto según la cantidad de datos etiquetados y el número de iteraciones de entrenamiento.
- Estudiar sus limitaciones ante el desbalanceo de clases.

El resto del artículo se organiza de la siguiente forma: la Sección 2 presenta los trabajos relacionados; la Sección 3 describe los materiales y la metodología propuesta; la Sección 4 presenta los resultados experimentales; y la Sección 5 recoge las conclusiones.

## 2. Trabajos Relacionados

### 2.1. Detección de objetos

La detección de objetos con aprendizaje profundo ha evolucionado a lo largo de la última década a través de dos líneas de desarrollo paralelas. Por un

lado, las arquitecturas convolucionales han pasado de propuestas iniciales como Faster R-CNN [22] a variantes en cascada como Cascade R-CNN [1], que refinan iterativamente las cajas delimitadoras. Los detectores de una sola etapa como RetinaNet [17], que abordaron el problema del desbalanceo extremo entre fondo y objetos mediante la función de pérdida *focal*, constituyen otra familia ampliamente utilizada. La introducción de las redes piramidales de características (FPN) [16] supuso un avance transversal a ambas familias, al permitir la detección multi-escala con un coste computacional moderado.

Por otro lado, los modelos basados en transformers como DETR [2], reformulan la detección como un problema de predicción de conjuntos y elimina componentes manuales como la supresión no máxima. Desde entonces, trabajos como RT-DETR [28] han demostrado que las arquitecturas basadas en transformers pueden operar en tiempo real manteniendo precisiones superiores a las de los detectores convolucionales equivalentes.

### 2.2. Aprendizaje semi-supervisado para detección

El aprendizaje semi-supervisado para detección de objetos (SSOD) busca reducir la dependencia de anotaciones masivas combinando un conjunto reducido de datos etiquetados con grandes volúmenes de datos sin etiquetar. En términos generales, los métodos SSOD se articulan en torno a dos paradigmas: el pseudo-etiquetado iterativo y la regularización por consistencia.

En el paradigma de pseudo-etiquetado, el modelo genera predicciones sobre los datos no etiquetados que se incorporan al entrenamiento como supervisión adicional. STAC [23] es una de las propuestas fundacionales en este ámbito: entrena un detector supervisado en la primera etapa y usa sus predicciones como pseudo-etiquetas para una segunda etapa con aumentación fuerte. SoftTeacher [27] mejora este esquema en un entrenamiento de extremo a extremo, empleando la cabeza de regresión del propio detector para filtrar las pseudo-etiquetas poco fiables sin necesidad de un segundo ciclo de entrenamiento. Unbiased Teacher v2 [20] generaliza la familia de métodos *teacher-student* a detectores sin anclas, introduciendo además un mecanismo de selección de pseudo-etiquetas basado en la incertidumbre relativa entre los modelos *teacher* y *student*. DenseTeacher [29] traslada estas ideas a los detectores de una etapa, sustituyendo las pseudo-etiquetas escasas de instancia por mapas de predicción densos que proporcionan una señal de supervisión más rica. MixPL [10] propone una solución genérica que equilibra el aprendizaje entre objetos de distinto tamaño y es aplicable a múltiples meta-arquitecturas.

La aplicación del SSOD al dominio específico de la conducción autónoma añade dificultades propias: objetos de pequeño tamaño, gran desbalanceo entre clases y distribución de escenas muy diferente a los conjuntos de referencia como COCO [18] o Pascal VOC [13]. En [5], analizamos este problema sobre datos reales de Waymo con SoftTeacher, concluyendo que con tan solo un 2 % de etiquetas se puede alcanzar un rendimiento próximo al supervisado completo, aunque las clases minoritarias siguen siendo un cuello de botella crítico.

### 2.3. Aprendizaje self-supervised para representaciones visuales

El aprendizaje self-supervised (SSL) ha protagonizado uno de los avances más relevantes de la visión por computador en los últimos años, al permitir aprender representaciones generalizables directamente a partir de datos sin etiquetar. Los métodos contrastivos como MoCo [15] y su evolución MoCo v2 [8] establecieron que una representación entrenada para distinguir vistas de la misma imagen de vistas de otras imágenes puede transferirse competitivamente a tareas supervisadas. BYOL [14] demostró que los ejemplos negativos no son imprescindibles: un mecanismo con dos redes asimétricas, donde una se actualiza mediante EMA de la otra, es suficiente para evitar el colapso y produce representaciones de alta calidad. SwAV [3] introduce una perspectiva diferente al apoyarse en asignaciones de prototipos en lugar de comparaciones por pares, lo que mejora la eficiencia y la estabilidad del entrenamiento. MoCo v3 [9] adapta el aprendizaje contrastivo a los Vision Transformers (ViT) [12], abriendo la puerta a representaciones con mayor capacidad de captura de contexto global. DINO [4] muestra que el entrenamiento self-supervised con ViT produce mapas de atención que se corresponden con regiones semánticamente significativas, lo que resulta especialmente prometedor para tareas de detección densa.

Sin embargo, el impacto de estos preentrenamientos sobre la detección semi-supervisada en el dominio específico de la conducción autónoma, con sus características particulares de desbalanceo de clases y objetos de pequeño tamaño, no ha sido analizado en detalle en la literatura existente.

## 3. Materiales y Métodos

La Figura 1 resume el flujo general del estudio, desde el conjunto de datos empleado hasta la tarea de detección semi-supervisada, y la comparación entre inicializaciones supervisadas y self-supervised del backbone.

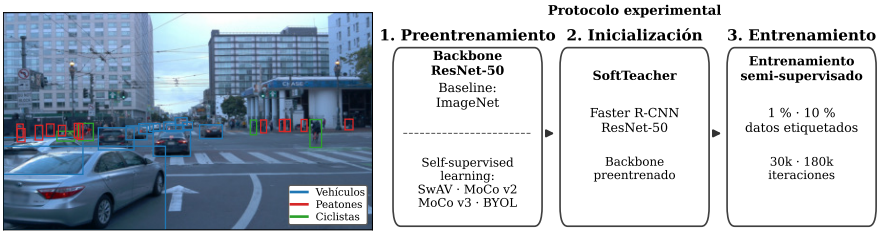



**Figura 1.** Esquema general del estudio: uso de imágenes RGB del conjunto Waymo para la detección de vehículos, peatones y ciclistas mediante SoftTeacher con Faster R-CNN, comparando la inicialización supervisada de ImageNet con distintos preentrenamientos self-supervised del backbone.

### 3.1. Waymo Open Dataset

Para este estudio utilizamos el conjunto de datos abierto Waymo Open Dataset [24], que contiene más de mil escenas de conducción grabadas en diferentes áreas urbanas. La tarea de detección se centra en identificar tres tipos de objetos (vehículos, peatones y ciclistas) a partir de las imágenes RGB obtenidas por las cinco cámaras a bordo del vehículo.

Para este estudio se selecciona un subconjunto representativo de alrededor de 100.000 imágenes con casi un millón de objetos anotados manualmente: 74.420 imágenes para entrenamiento y 24.470 para validación. Existe un gran desbalanceo en la distribución de clases: vehículos representan el 87,9 % de las instancias, peatones el 11,5 % y ciclistas únicamente el 0,6 %. Este desbalanceo es un factor determinante en el análisis de resultados, especialmente para las clases minoritarias.

### 3.2. Modelo base: SoftTeacher con Faster R-CNN

El modelo SoftTeacher [27] propone un enfoque de detección de objetos semi-supervisado basado en la arquitectura *teacher-student*. Durante el entrenamiento, el modelo *teacher* genera pseudo-etiquetas sobre las imágenes no etiquetadas mediante un promedio exponencial móvil (EMA) de los pesos del modelo *student*. Dichas pseudo-etiquetas se utilizan conjuntamente con las etiquetas reales para optimizar el modelo *student* de acuerdo con la función de pérdida:

$$\mathcal{L} = \mathcal{L}_s + \alpha \mathcal{L}_u \tag{1}$$

donde $\mathcal{L}_s$ es la pérdida supervisada sobre los datos etiquetados, $\mathcal{L}_u$ es la pérdida no supervisada sobre las pseudo-etiquetas y $\alpha$ pondera la contribución del aprendizaje no supervisado. En todos los experimentos se emplea Faster R-CNN con FPN como meta-arquitectura.

### 3.3. Preentrenamiento self-supervised del backbone

En el trabajo previo [5] el backbone ResNet-50 de SoftTeacher se inicializaba con los pesos supervisados de ImageNet proporcionados por torchvision, que constituyen la línea base estándar en detección de objetos. En este estudio investigamos si es posible mejorar esa inicialización empleando pesos obtenidos mediante aprendizaje self-supervised. El procedimiento es el siguiente: partiendo del mismo ResNet-50 de torchvision, se aplica un algoritmo de preentrenamiento self-supervised con las imágenes de Waymo partiendo de los pesos supervisados de ImageNet; los pesos resultantes se usan después como punto de partida del backbone en SoftTeacher, en lugar de los pesos supervisados originales. De este modo, la única diferencia entre la línea base y cada variante es la inicialización del backbone.

Se evalúan cuatro métodos de preentrenamiento self-supervised, que representan las principales familias del estado del arte:

- **SwAV** [3]: aprende representaciones asignando múltiples vistas de la misma imagen a prototipos compartidos, evitando las comparaciones explícitas entre pares.
- **MoCo v2** [8]: aprendizaje contrastivo que mantiene un banco de claves negativas mediante un encoder de momentum.
- **MoCo v3** [9]: extiende el aprendizaje contrastivo eliminando el banco de claves y adoptando un esquema de dos ramas simétricas.
- **BYOL** [14]: aprende sin ejemplos negativos mediante dos redes asimétricas, donde una se actualiza como media exponencial móvil de la otra, evitando el colapso sin necesidad de contrastividad explícita.

### 3.4. Configuración experimental

Siguiendo el protocolo de validación del trabajo previo [5], evaluamos el rendimiento del sistema con dos niveles de disponibilidad de datos etiquetados: 1 % y 10 % del conjunto de entrenamiento. Para cada configuración se analizan dos números de iteraciones de entrenamiento: 30k y 180k. Estos experimentos permiten comparar el comportamiento de las distintas inicializaciones del backbone tanto en condiciones de datos muy escasos como en un régimen más amplio.

El desarrollo experimental se apoya en el ecosistema OpenMMLab: MMPreTrain se utiliza para los modelos con preentrenamiento self-supervised [21], mientras que MMDetection se emplea para la implementación semi-supervisada de SoftTeacher [6].

En todos los experimentos se emplea Faster R-CNN con FPN y se mantienen los parámetros por defecto de SoftTeacher: ratio de muestreo de imágenes etiquetadas frente a no etiquetadas de 1:4 y umbral de confianza de pseudo-etiquetado de 0,9. Se aplican todas las transformaciones de color y geométricas definidas en [27]. El entrenamiento se lleva a cabo usando 3 GPUs Nvidia RTX 3090 24GB con un tamaño de batch de 8 imágenes. Se utiliza SGD como optimizador, con una tasa de aprendizaje inicial de 0,01, reduciéndola en un factor de 10 en las iteraciones correspondientes al 67 % y al 95 % del total.

Para el preentrenamiento self-supervised se utilizan las configuraciones por defecto de los modelos disponibles en MMPreTrain, fijando una duración de entrenamiento de 200 epochs para todos los métodos evaluados.

Como métrica de evaluación se emplea el *Average Precision* (AP), evaluado para cada una de las tres clases de objetos: vehículos (AP@IoU=0,7), peatones (AP@IoU=0,5) y ciclistas (AP@IoU=0,5).

## 4. Resultados Experimentales

### 4.1. Comparativa global entre métodos de preentrenamiento

La Tabla 1 presenta los resultados obtenidos con 30k iteraciones de entrenamiento para todos los métodos de preentrenamiento evaluados, con un 1 % y un 10 % de datos etiquetados. La Tabla 2 recoge los resultados correspondientes a 180k iteraciones.

**Cuadro 1.** Comparativa del rendimiento (Average Precision) con 30k iteraciones de entrenamiento según el método de preentrenamiento del backbone y el porcentaje de datos etiquetados.

| Aprendizaje | Init. Backbone | % Etiq. | Vehículos | Peatones | Ciclistas | Media |
|---|---|---|---|---|---|---|
| Supervisado | ImageNet | 100 % | 54,56 | 66,50 | 46,34 | 55,80 |
| Semi-supervisado | ImageNet [5] | 10 % | 48,23 | 59,50 | 33,67 | 47,13 |
| | SwAV | 10 % | **51,26** | **62,54** | **35,40** | **49,73** |
| | MoCo v3 | 10 % | 50,91 | 62,19 | 32,82 | 48,64 |
| | MoCo v2 | 10 % | 50,90 | 61,64 | 24,15 | 45,56 |
| | BYOL | 10 % | 50,68 | 61,49 | 34,21 | 48,79 |
| Semi-supervisado | ImageNet [5] | 1 % | 43,81 | 52,84 | 9,72 | 35,46 |
| | SwAV | 1 % | **46,67** | **57,56** | **23,67** | **42,63** |
| | MoCo v3 | 1 % | 46,63 | 56,90 | 15,21 | 39,58 |
| | MoCo v2 | 1 % | 45,85 | 55,25 | 7,05 | 36,05 |
| | BYOL | 1 % | 46,35 | 57,04 | 19,56 | 40,98 |

En el caso de 30k iteraciones con un 10 % de datos etiquetados, todos los métodos de preentrenamiento self-supervised mejoran el rendimiento medio respecto a la inicialización supervisada estándar (47,13), siendo SwAV el que mayor ganancia aporta con una media de 49,73 (+2,60 puntos). MoCo v3 y BYOL ofrecen mejoras similares en torno a +1,5 puntos. Por su parte, MoCo v2 presenta resultados menos uniformes: mejora las clases de vehículos y peatones pero experimenta una caída significativa en ciclistas (24,15 frente a 33,67 de la inicialización con los pesos de ImageNet), lo que penaliza su media hasta 45,56, por debajo incluso de la inicialización supervisada.

Con un 1 % de datos etiquetados y 30k iteraciones, la mejora introducida por el preentrenamiento self-supervised es aún más pronunciada. La inicialización supervisada obtiene una media de 35,46, mientras que SwAV alcanza 42,63 (+7,17 puntos) y BYOL 40,98 (+5,52 puntos). MoCo v3 también mejora notablemente con 39,58 puntos. Destaca especialmente la mejora en la clase de ciclistas con SwAV (23,67 frente a 9,72 de la inicialización con los pesos de ImageNet), lo que indica que el preentrenamiento self-supervised puede proporcionar representaciones más útiles para detectar instancias escasas cuando los datos etiquetados son muy limitados.

En conjunto, la brecha respecto al techo supervisado con el 100 % de los datos (55,80 puntos de media) es todavía considerable en el régimen de 30k iteraciones: incluso el mejor modelo semi-supervisado con preentrenamiento SSL, SwAV al 10 %, se sitúa 6,07 puntos por debajo.

### 4.2. Efecto del número de iteraciones

Al aumentar el número de iteraciones a 180k con un 10 % de datos etiquetados, todos los métodos convergen a resultados muy similares entre sí, en torno a 54 puntos de media AP. SwAV (54,33) y MoCo v3 (54,52) superan a la línea base

**Cuadro 2.** Comparativa del rendimiento (Average Precision) con 180k iteraciones de entrenamiento según el método de preentrenamiento del backbone y el porcentaje de datos etiquetados.

| Aprendizaje | Init. Backbone | % Etiq. | Vehículos | Peatones | Ciclistas | Media |
|---|---|---|---|---|---|---|
| Supervisado | ImageNet | 100 % | 54,56 | 66,50 | 46,34 | 55,80 |
| Semi-supervisado | ImageNet [5] | 10 % | 53,14 | 63,05 | 42,04 | 52,74 |
| | SwAV | 10 % | **54,96** | 65,25 | 42,77 | 54,33 |
| | MoCo v3 | 10 % | 54,76 | **65,61** | **43,18** | **54,52** |
| | BYOL | 10 % | 54,88 | 65,30 | 41,32 | 53,83 |
| Semi-supervisado | ImageNet [5] | 1 % | 39,64 | 48,53 | 6,81 | 31,66 |
| | SwAV | 1 % | 45,32 | 53,40 | 12,54 | 37,09 |
| | MoCo v3 | 1 % | 45,40 | **54,53** | **22,91** | **40,95** |
| | BYOL | 1 % | **46,17** | 54,16 | 17,89 | 39,41 |

supervisada (52,74) en aproximadamente 1,5–1,8 puntos. BYOL (53,83) también mejora, aunque de forma más moderada. Es relevante que MoCo v3 supera ligeramente a SwAV en el régimen de 180k, mientras que SwAV era superior con 30k, lo que sugiere que MoCo v3 podría beneficiarse en mayor medida de un entrenamiento más prolongado. Respecto al techo supervisado completo (55,80), la brecha se reduce aquí a poco más de 1 punto para los mejores modelos SSL, lo que supone un resultado muy competitivo partiendo únicamente del 10 % de los datos etiquetados.

Con un 1 % de datos etiquetados y 180k iteraciones, el patrón es diferente. La inicialización supervisada desciende hasta 31,66 puntos de media, confirmando el fenómeno de sobreajuste ya observado en el trabajo previo [5]: cuando los datos etiquetados son muy escasos y las pseudo-etiquetas de baja calidad se acumulan, un mayor número de iteraciones perjudica al modelo. Sin embargo, el preentrenamiento self-supervised mitiga parcialmente este efecto. MoCo v3 logra 40,95 puntos en este escenario, una mejora de más de 9 puntos respecto a la línea base; SwAV alcanza 37,09 y BYOL 39,41. Los tres métodos self-supervised superan claramente la línea base en este régimen, lo que confirma que las representaciones iniciales más ricas actúan como regularización implícita frente al sobreajuste por pseudo-etiquetas de baja calidad.

### 4.3. Análisis por clases

El análisis del rendimiento por clase revela patrones importantes relacionados con el desbalanceo del conjunto de datos. Para la clase mayoritaria, vehículos, todos los métodos self-supervised obtienen mejoras modestas pero consistentes respecto a la línea base. Para peatones, los beneficios son también claros y estables. La clase de ciclistas, con apenas un 0,6 % de representación, es la más sensible a la elección del preentrenamiento y la que muestra mayor variabilidad entre métodos.

Con un 10 % de datos etiquetados y 180k iteraciones, MoCo v3 logra el mejor resultado en ciclistas (43,18), seguido de SwAV (42,77) y BYOL (41,32), mientras que la línea base supervisada obtiene 42,04. Las diferencias en este régimen son pequeñas, lo que indica que con suficientes datos y tiempo de entrenamiento la ventaja del preentrenamiento self-supervised se reduce. Con un 1 % de datos etiquetados y 30k iteraciones, las diferencias son mucho mayores: SwAV llega a 23,67 puntos en ciclistas frente a 9,72 de la línea base, lo que representa una mejora de más del doble. MoCo v3 obtiene 15,21 y BYOL 19,56 en este mismo escenario. En el régimen de 1 % y 180k, la clase de ciclistas es donde más acusa el sobreajuste: la línea base cae a 6,81, mientras que MoCo v3 retiene 22,91 puntos, BYOL 17,89 y SwAV 12,54.

### 4.4. Discusión

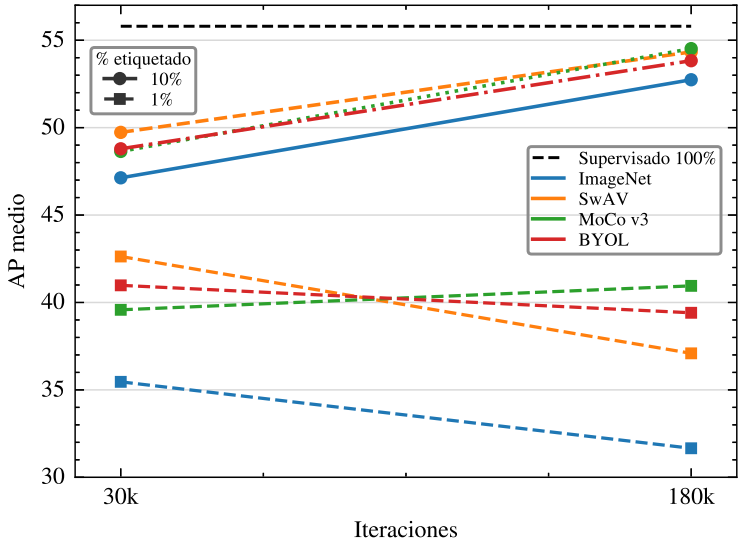



**Figura 2.** Resumen comparativo del rendimiento medio (AP) entre las tres clases de objetos para las distintas configuraciones de preentrenamiento, porcentaje de datos etiquetados e iteraciones de entrenamiento.

La Figura 2 resume el comportamiento global de los métodos evaluados a partir del AP medio entre vehículos, peatones y ciclistas. Los resultados obtenidos demuestran que el preentrenamiento self-supervised del backbone constituye una mejora efectiva y complementaria al aprendizaje semi-supervisado para la detección de objetos en vehículos autónomos. La ganancia es especialmente notable cuando los datos etiquetados son muy escasos (1 %), donde el preentrenamiento proporciona una mejor inicialización de las representaciones y reduce el sobreajuste al aprendizaje de pseudo-etiquetas de baja calidad.

SwAV y MoCo v3 emergen como los métodos más robustos a lo largo de las distintas configuraciones evaluadas. BYOL ofrece igualmente mejoras consistentes, posicionándose por encima de la línea base en todos los escenarios y mostrando una resistencia al sobreajuste comparable a la de MoCo v3 en el régimen de 1 % y 180k iteraciones. MoCo v2 muestra el comportamiento más irregular, con una caída destacada en la clase de ciclistas.

En perspectiva global, la combinación de preentrenamiento SSL con aprendizaje semi-supervisado permite acercarse notablemente al rendimiento del modelo completamente supervisado con el 100 % de los datos (55,80 puntos). Con tan solo el 10 % de etiquetas y 180k iteraciones, MoCo v3 alcanza 54,52 puntos, una diferencia de apenas 1,28 puntos respecto al techo supervisado. Este resultado es especialmente relevante en el contexto de la conducción autónoma, donde la obtención de grandes volúmenes de anotaciones es costosa y lenta.

A pesar de las mejoras obtenidas, el desbalanceo de clases sigue siendo el principal desafío del sistema. Incluso con el mejor preentrenamiento self-supervised, la detección de ciclistas en condiciones de datos muy limitados se mantiene por debajo de niveles operativamente aceptables para un sistema de conducción autónoma.

## 5. Conclusiones

En este trabajo hemos investigado el impacto del preentrenamiento self-supervised del backbone ResNet-50 sobre el rendimiento del modelo de detección semi-supervisado SoftTeacher en el dominio de los vehículos autónomos, utilizando datos reales del conjunto de Waymo. Los resultados muestran que SwAV, MoCo v3 y BYOL mejoran consistentemente el rendimiento respecto a la inicialización supervisada estándar con los pesos de ImageNet, siendo la ganancia más notable cuando la disponibilidad de datos etiquetados es muy reducida (1 %). En este escenario, el preentrenamiento self-supervised actúa como regularizador implícito, mitigando el sobreajuste a pseudo-etiquetas de baja calidad que se observa con la inicialización estándar al aumentar el número de iteraciones.

En el régimen con mayor disponibilidad de etiquetas y entrenamiento prolongado, los métodos self-supervised tienden a converger hacia un rendimiento muy similar entre sí y mejoran la inicialización supervisada estándar. Sin embargo, todavía no alcanzan por completo el comportamiento del modelo entrenado con todas las anotaciones disponibles. Esto indica que el preentrenamiento self-supervised ayuda a reducir la dependencia del etiquetado manual, pero no sustituye por completo la necesidad de datos anotados del propio dominio.

El desbalanceo de clases permanece como la principal limitación del sistema, con la clase de ciclistas mostrando la mayor variabilidad y las menores mejoras absolutas. Trabajos futuros deberán explorar estrategias específicas para clases minoritarias, como técnicas de aumento de datos específicas del dominio, selección activa de instancias representativas, preentrenamiento self-supervised sobre datos del propio dominio de conducción autónoma, y estrategias de ponderación en la función de pérdida para equilibrar el aprendizaje entre clases desbalanceadas.

**Acknowledgments.** Este trabajo de investigación es parte del proyecto PID2023-146037OB-C21 financiado por MICIU/AEI/10.13039/501100011033.

## Referencias

1. Cai, Z., Vasconcelos, N.: Cascade R-CNN: Delving into high quality object detection. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6154–6162 (2018). https://doi.org/10.1109/CVPR.2018.00644
2. Carion, N., Massa, F., Synnaeve, G., et al.: End-to-end object detection with transformers. In: Computer Vision – ECCV 2020. pp. 213–229. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58452-8_13
3. Caron, M., Misra, I., Mairal, J., et al.: Unsupervised learning of visual features by contrasting cluster assignments. In: Advances in Neural Information Processing Systems. vol. 33, pp. 9912–9924 (2020)
4. Caron, M., Touvron, H., Misra, I., et al.: Emerging properties in self-supervised vision transformers. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 9630–9640 (2021). https://doi.org/10.1109/ICCV48922.2021.00951
5. Carranza-García, M., Martínez-Ballesteros, M., Riquelme, J.C.: Detección de objetos con enfoque semi-supervisado en vehículos autónomos. In: XX Conferencia de la Asociación Española para la Inteligencia Artificial (CAEPIA 2023). pp. 549–554 (2023)
6. Chen, K., Wang, J., Pang, J., et al.: MMDetection: Open MMLab detection toolbox and benchmark. arXiv preprint arXiv:1906.07155 (2019)
7. Chen, L., Teng, S., Li, B., et al.: Milestones in autonomous driving and intelligent vehicles—Part II: Perception and planning. IEEE Transactions on Systems, Man, and Cybernetics: Systems **53**(10), 6096–6110 (2023). https://doi.org/10.1109/TSMC.2023.3283021
8. Chen, X., Fan, H., Girshick, R., He, K.: Improved baselines with momentum contrastive learning. arXiv preprint arXiv:2003.04297 (2020)
9. Chen, X., Xie, S., He, K.: An empirical study of training self-supervised vision transformers. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 9640–9649 (2021). https://doi.org/10.1109/ICCV48922.2021.00950
10. Chen, Z., Zhang, W., Wang, X., Chen, K., Wang, Z.: Mixed pseudo labels for semi-supervised object detection. arXiv preprint arXiv:2312.07006 (2023)
11. Chib, P.S., Singh, P.: Recent advancements in end-to-end autonomous driving using deep learning: A survey. IEEE Transactions on Intelligent Vehicles **9**(1), 103–118 (2024). https://doi.org/10.1109/TIV.2023.3318070
12. Dosovitskiy, A., Beyer, L., Kolesnikov, A., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: 9th International Conference on Learning Representations (ICLR 2021) (2021)
13. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The Pascal visual object classes (VOC) challenge. International Journal of Computer Vision **88**(2), 303–338 (2010). https://doi.org/10.1007/s11263-009-0275-4
14. Grill, J.B., Strub, F., Altché, F., et al.: Bootstrap your own latent: A new approach to self-supervised learning. In: Advances in Neural Information Processing Systems. vol. 33, pp. 21271–21284 (2020)

15. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 9729–9738 (2020). https://doi.org/10.1109/CVPR42600.2020.00975
16. Lin, T.Y., Dollár, P., Girshick, R., et al.: Feature pyramid networks for object detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 936–944 (2017). https://doi.org/10.1109/CVPR.2017.106
17. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence **42**(2), 318–327 (2020). https://doi.org/10.1109/TPAMI.2018.2858826
18. Lin, T.Y., Maire, M., Belongie, S., et al.: Microsoft COCO: Common objects in context. In: Computer Vision – ECCV 2014. pp. 740–755. Springer (2014). https://doi.org/10.1007/978-3-319-10602-1_48
19. Liu, M., Yurtsever, E., Fossaert, J., et al.: A survey on autonomous driving datasets: Statistics, annotation quality, and a future outlook. IEEE Transactions on Intelligent Vehicles **9**(7), 7138–7164 (2024). https://doi.org/10.1109/TIV.2024.3371566
20. Liu, Y.C., Ma, C.Y., Kira, Z.: Unbiased teacher v2: Semi-supervised object detection for anchor-free and anchor-based detectors. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 9819–9828 (2022). https://doi.org/10.1109/CVPR52688.2022.00959
21. MMPreTrain Contributors: OpenMMLab's pre-training toolbox and benchmark. https://github.com/open-mmlab/mmpretrain (2023)
22. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**, 1137–1149 (2017). https://doi.org/10.1109/TPAMI.2016.2577031
23. Sohn, K., Zhang, Z., Li, C.L., et al.: A simple semi-supervised learning framework for object detection. arXiv preprint arXiv:2005.04757 (2020)
24. Sun, P., Kretzschmar, H., Dotiwalla, X., et al.: Scalability in perception for autonomous driving: Waymo open dataset. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2443–2451 (2020). https://doi.org/10.1109/CVPR42600.2020.00252
25. TorchVision contributors: Torchvision: Pytorch's computer vision library. https://github.com/pytorch/vision (2016)
26. Trigka, M., Dritsas, E.: A comprehensive survey of machine learning techniques and models for object detection. Sensors **25**(1), 214 (2025). https://doi.org/10.3390/s25010214
27. Xu, M., Zhang, Z., Hu, H., et al.: End-to-end semi-supervised object detection with soft teacher. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 3040–3049 (2021). https://doi.org/10.1109/ICCV48922.2021.00305
28. Zhao, Y., Lv, W., Xu, S., et al.: Detrs beat yolos on real-time object detection. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 16965–16974 (2024). https://doi.org/10.1109/CVPR52733.2024.01605
29. Zhou, H., Ge, Z., Liu, S., et al.: Dense teacher: Dense pseudo-labels for semi-supervised object detection. In: Computer Vision – ECCV 2022. pp. 35–50. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-20077-9_3