U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Exploring Label Efficiency with Semi-Supervision and Self-Supervision Methods

Francisco Gonçalves Cerqueira

Supervisor
Ricardo Pereira de Magalhães Cruz

July 18, 2024
Dissertation
Master in Informatics and Computing Engineering

Table of Contents

Context

- The field of Autonomous Driving has increasingly emerged in recent years.
- Society of Automotive Engineers (SAE) classifies levels of automation from Level 0 to Level 5. [1]
- Artificial Intelligence and Deep Learning become crucial algorithms.
- Training demands large quantities of data.

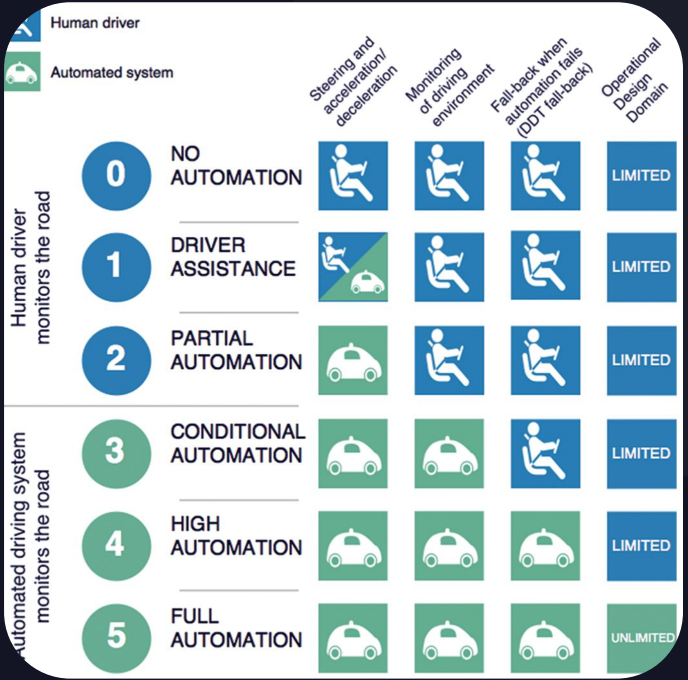


Fig. 1: SAE J3016 levels of driving automation. [2]

[1] Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles. On-Road Automated Driving (ORAD) Committee, April 2021
[2] Alexandru Constantin Serban, Erik Poll, and Joost Visser. A standard driven software architecture for fully autonomous vehicles. In 2018 IEEE International Conference on Software Architecture Companion (ICSA-C).

Motivation

Low Annotation Speed

High Annotation Cost

Susceptible to Errors

Labor-intensive

Theia
Automated Perception Driving

Goals & Contributions

Annotation Dependency

Reduce the dependency of annotated data during the training, by using Semi-supervision or Self-supervision.

Comparative Study

Access and compare the used methodologies in the context of Autonomous Driving.

Software Framework

Develop a software framework, simplifying the adaptation of these techniques to other domains.

Learning Paradigms

Supervised Learning

Model is trained on a labeled dataset, learning the relationship between input data and corresponding labels.

Unsupervised Learning

Model is trained on an unlabeled dataset, seeking to discover inherent patterns, relationships, or structures within the data without explicit labels.

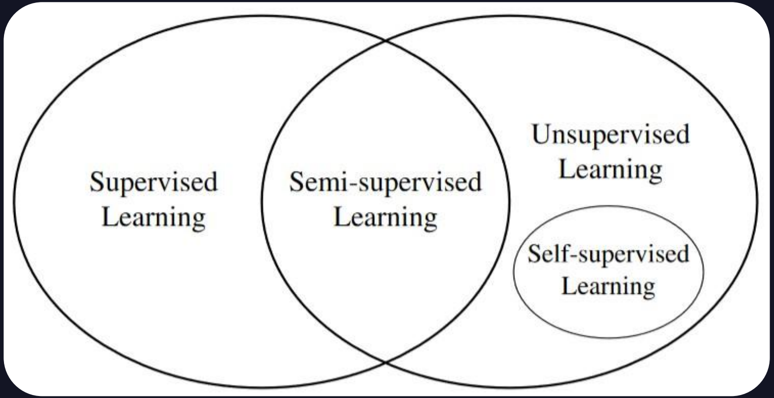


Fig. 2: Venn diagram representing the main learning paradigms.

Semi-supervised Learning

Overview

- Combination of SL and UL.
- Leverages labeled and unlabeled data at a single training instance.
- Comprises loss terms that leverage unlabeled data.
- Flexibility in regularizing the strength of those terms.
- Constrained to the task predetermined by the method.

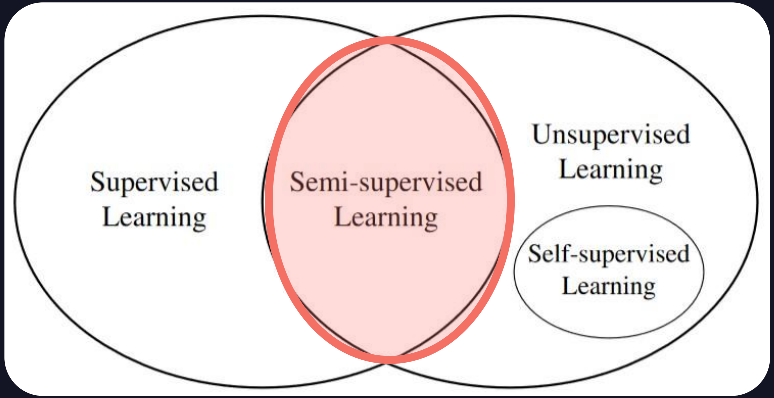


Fig. 3: Venn diagram representing the main learning paradigms with Semi-supervised Learning highlighted.

Categories

Consistency Regularization

Ensures model robustness through consistent predictions when presented with augmented versions of the same input. [3]

Entropy Minimization

Encourages low entropy/ high-confidence predictions on unlabeled data. [4]

Pseudo-labeling

Empowers the model to generate surrogate labels for unlabeled data by assigning "hard" labels. [5]

[3] Philip Bachman, Ouais Alsharif, and Doina Precup. Learning with Pseudo-Ensembles. Advances in neural information processing systems, 27, 2014
[4] Yves Grandvalet and Yoshua Bengio. Semi-supervised Learning by Entropy Minimization. Advances in neural information processing systems, 17, 2004.
[5] Geoffrey J McLachlan. Iterative Reclassification Procedure for Constructing an Asymptotically Optimal Rule of Allocation in Discriminant Analysis. Journal of the American Statistical Association, 70(350):365–369, 1975

Pi-Model [6]

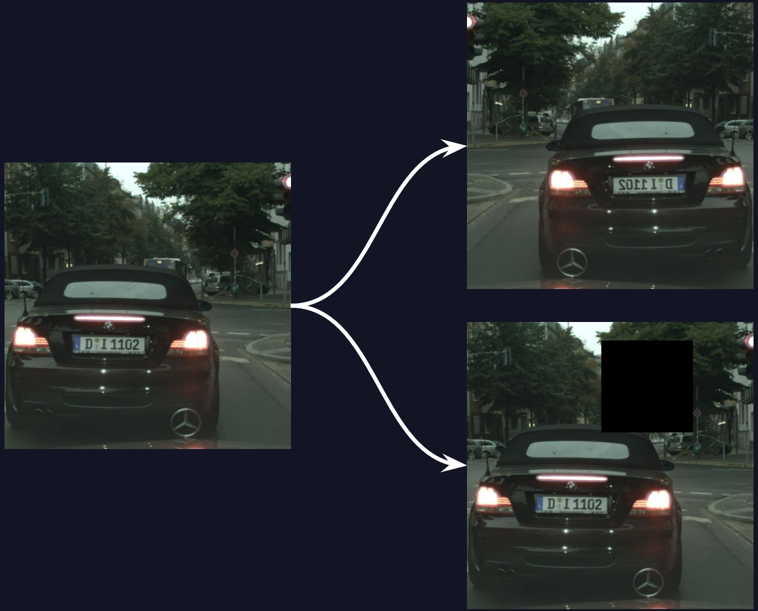


Fig. 4: Exemplification of Pi-Model's data augmentation pipeline.

[6] Samuli Laine and Timo Aila. Temporal Ensembling for Semi-Supervised Learning. In International Conference on Learning Representations, 2017.

Self-supervised Learning

Overview

- Subset of UL.
- Leverages only unlabeled data using a pretext task.
- Fine-tuning is performed later using other paradigms (e.g. SL).
- Easily adapted to any downstream task.

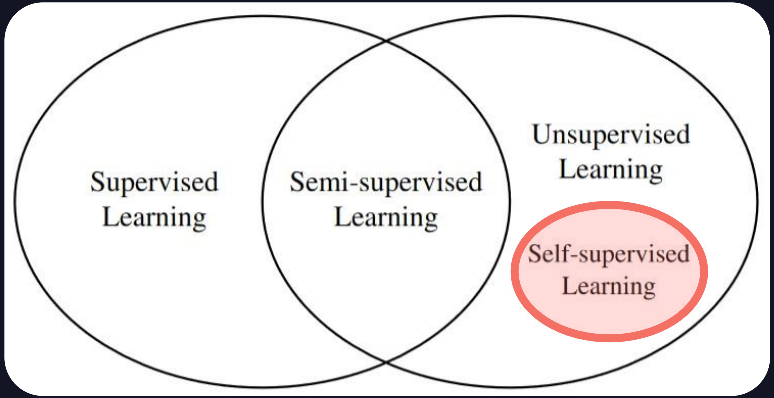


Fig. 5: Venn diagram representing the main learning paradigms with Self-supervised Learning highlighted.

Rotation Prediction [7]

Fig. 6: Illustration of the rotation prediction self-supervised task.

[7] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised Representation Learning by Predicting Image Rotations.

Workflow

Techniques Selection

Select and implement Semi-supervision and Self-supervision methods.

Context Adaptation

Adapt the methods to Autonomous Driving context.

Comparative Study

Evaluate the performance of these methods and compare them.

Selected Methods

Supervised:

- Random Weights Initialization
- Pretrained Weights Initialization

Semi-Supervised:

- Pi-Model
- Temporal Ensembling
- MixMatch
- ReMixMatch
- FixMatch

Self-Supervised:

- Rotation Prediction
- SimCLR
- BYOL
- MoCo

Datasets, Models & Metrics

	Classification	Semantic Segmentation
Datasets	CIFAR-10 SVHN	Cityscapes KITTI
Models	Wide ResNet28-2 ResNet50	DeepLabV3 ResNet101 MobileNetV3-Large
Metrics	Top-1 Accuracy	mIoU

Fig. 7: Matrix illustrating the combination of datasets,models and metrics used.

313 Experiments

Methods Adaptation

Self-Supervised Learning

No modifications required!

Semi-Supervised Learning

Ensuring consistency in the transformations applied was crucial, as they should not alter the masks.
Therefore, the geometrical transformations should be the same all the transformed versions of the unlabeled data.

Methods Adaptation

Self-Supervised Learning

No modifications required!

Semi-Supervised Learning

Ensuring consistency in the transformations applied was crucial, as they should not alter the masks.
Therefore, the geometrical transformations should be the same all the transformed versions of the unlabeled data.

One of the main contributions!

Results

Methods Legend

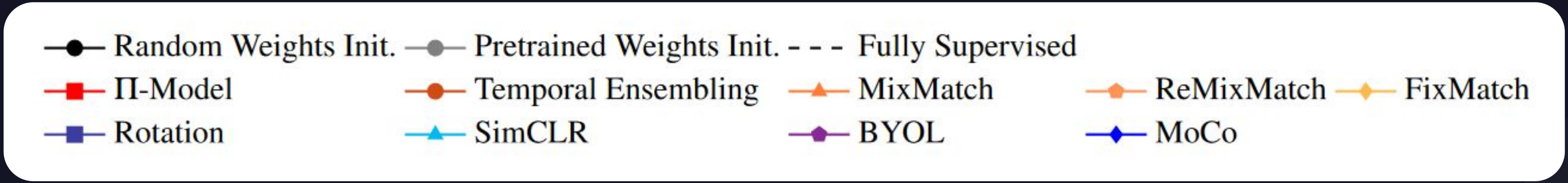
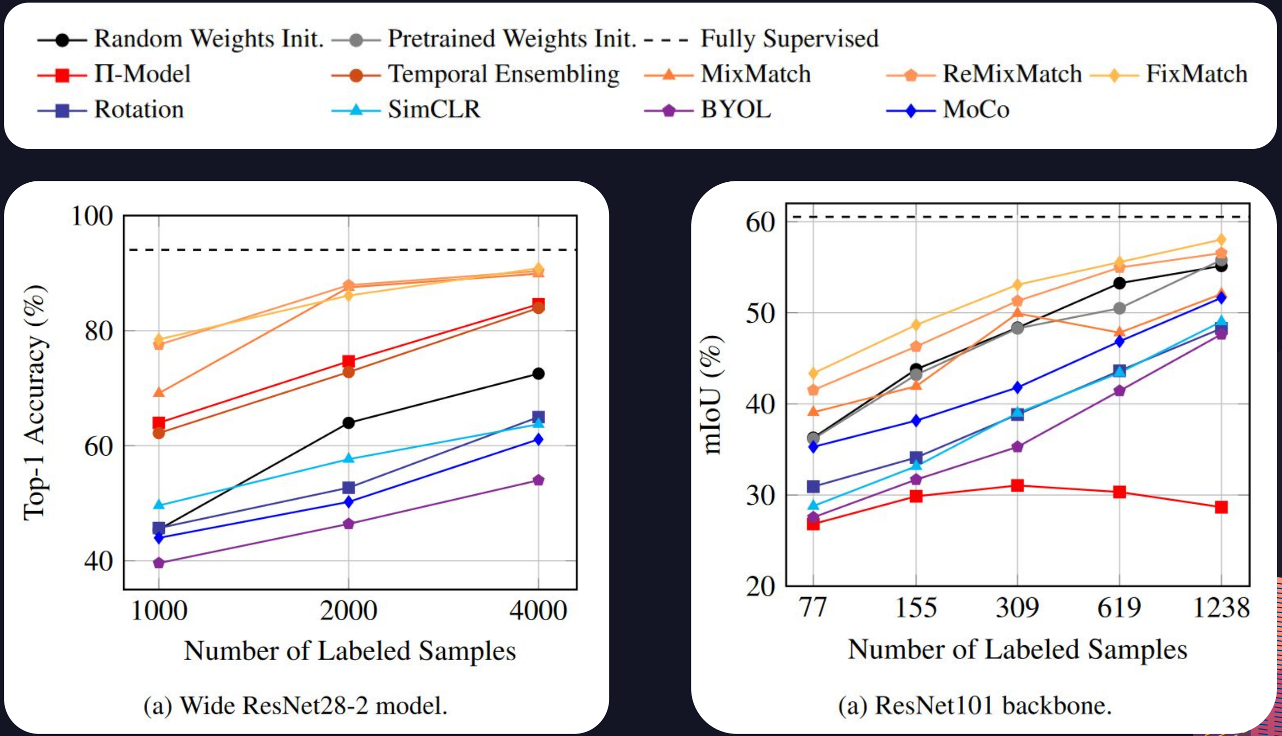


Fig. 8: Legend for the benchmark results. Semi-supervised methods are depicted using shades of red, while self-supervised methods are shown with shades of blue.

Intra-Paradigm

- **Semi-Supervision:** two methods stand out - ReMixMatch and FixMatch.
- **Self-Supervision:** results more closely clustered - MoCo generally better.



(a) Wide ResNet28-2 model.

(a) ResNet101 backbone.

Fig. 9: Results for CIFAR-10 and Cityscapes, respectively.

Cross-Paradigm

	CIFAR-10			SVHN		
Method	1000 labels	2000 labels	4000 labels	250 labels	500 labels	1000 labels
Random Weights Init.	39.96	47.82	55.62	20.71	27.56	52.98
Pretrained Weights Init.	**64.47**	**69.71**	**75.30**	40.70	56.01	73.21
Π-Model	51.26	59.32	65.94	**42.69**	**83.41**	82.25
Temporal Ensembling	49.78	58.64	60.92	23.44	37.46	79.95
MixMatch	**61.62**	**73.29**	**80.45**	–	–	–
ReMixMatch	43.33	42.16	59.27	40.28	**79.92**	**85.81**
FixMatch	**61.28**	**69.72**	**81.91**	25.08	**84.06**	**91.47**
Rotation	50.07	61.35	69.38	29.45	60.94	77.88
SimCLR	39.11	44.47	51.74	**50.47**	74.14	**82.97**
BYOL	52.97	58.85	66.53	31.02	52.20	73.30
MoCo	60.74	66.77	73.24	**43.99**	69.98	75.90

Fig. 10: Results for CIFAR-10 and SVHN (ResNet50). For each column, top-3 results are in **bold**.

- **Semi-Supervised Learning** methods generally offer better performance and label-efficiency.

Method	$77_{(1/32)}$	$155_{(1/16)}$	$309_{(1/8)}$	$619_{(1/4)}$	$1238_{(1/2)}$	2475_{All}
Random Weights Init.	36.28	**43.79**	48.35	**53.24**	55.14	60.52
Pretrained Weights Init.	36.16	43.19	48.28	50.49	**55.79**	–
Π-Model	26.82	29.86	31.05	30.33	28.67	–
MixMatch	**39.08**	41.93	**49.93**	47.81	52.06	–
ReMixMatch	**41.51**	**46.29**	**51.29**	**54.96**	**56.54**	–
FixMatch	**43.35**	**48.66**	**53.07**	**55.55**	**58.03**	–
Rotation	30.92	34.11	38.83	43.63	48.28	–
SimCLR	28.79	33.15	38.99	43.42	49.03	–
BYOL	27.55	31.69	35.28	41.43	47.64	–
MoCo	35.28	38.16	41.80	46.86	51.64	–

Fig. 11: Results for Cityscapes (ResNet101). For each column, top-3 results are in **bold**.

Backbone Comparison

- **Semi-supervised** methods perform well with both small and large backbones.
- **Self-supervised** methods require larger backbones to achieve significant performance gains.

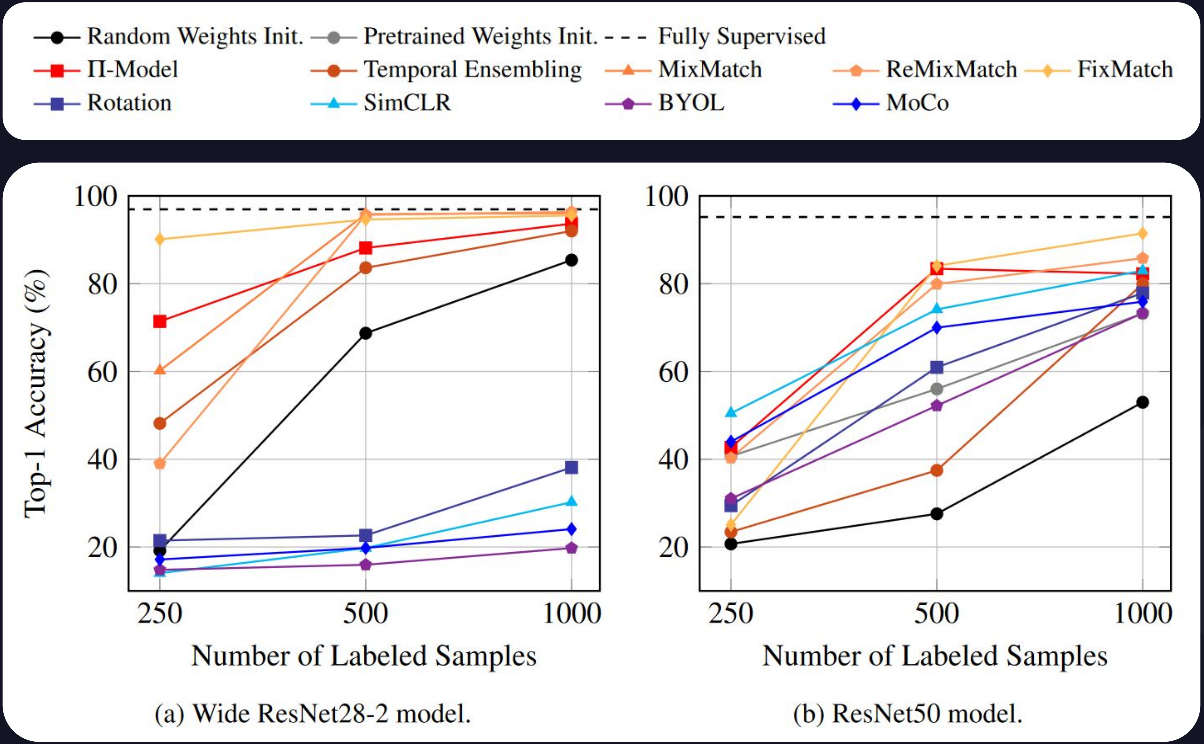


Fig. 12: Results for SVHN.

Pretrained Baseline Comparison

- The top-2 **semi-supervised** methods surpass the performance of pre-trained models.
- **Self-supervised** methods are more inconsistent in outperforming pre-trained models.
- Pretrained models achieved better performance in the early training stages than any learning paradigm.

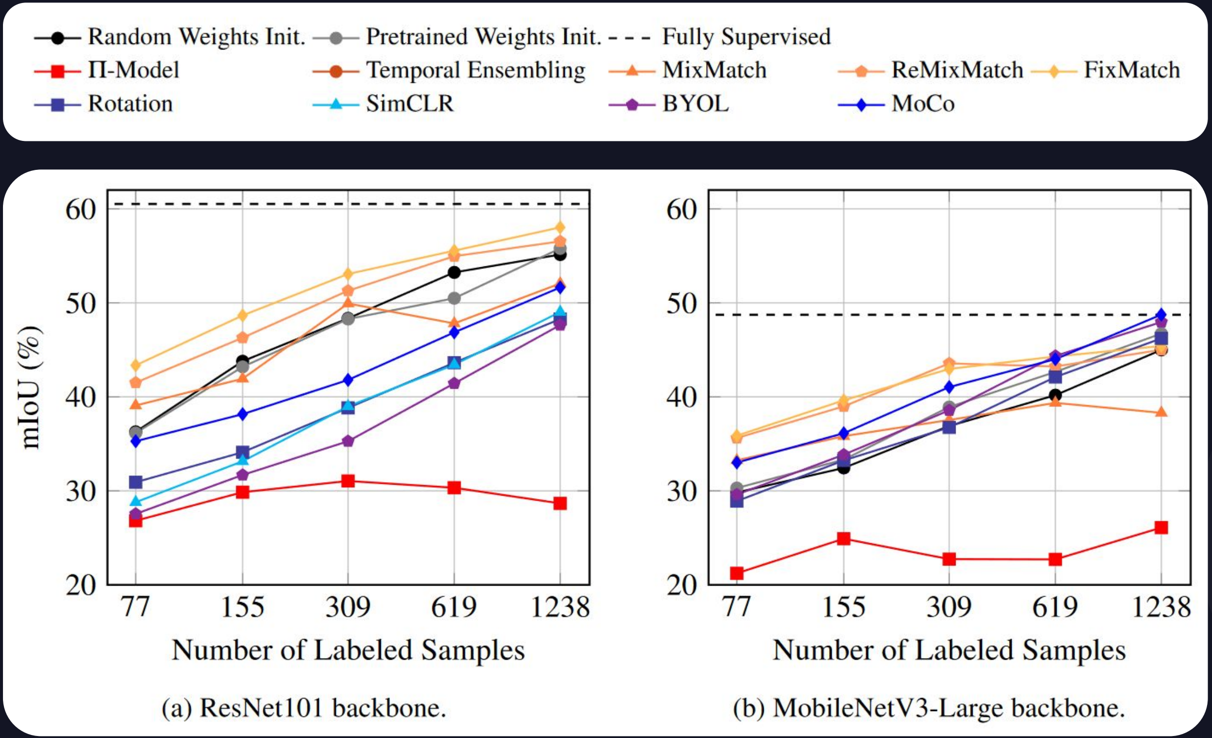


Fig. 13: Results for Cityscapes.

Training Time

- **Supervised Learning** is the quickest
- **Self-Supervised Learning** has the longest training duration, but it is independent from the labeled data.
- Although **semi-supervision** methods tend to be **faster**, some cases might benefit from using **self-supervision**.

Method	Wide ResNet28-2	ResNet101
Rotation	20.757±0.078	2397.229±1.673
SimCLR	395.335±0.152	896.545±49.879
BYOL	497.620±5.741	3358.129±59.964
MoCo	209.849±0.456	1011.251±9.537

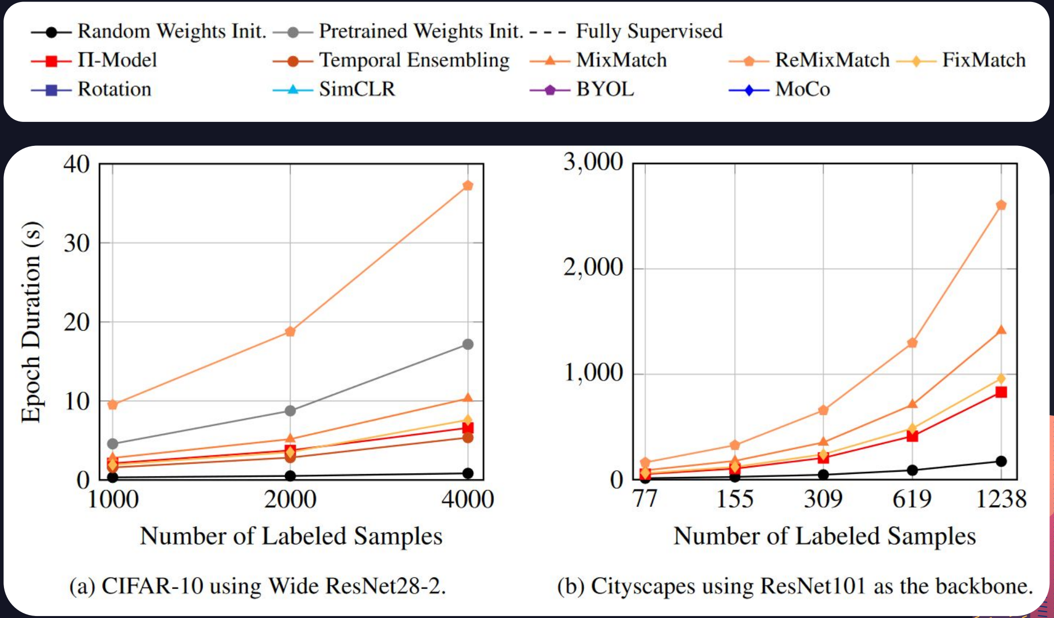


Fig. 14: Mean epoch duration (seconds).

Conclusions

- Generally, Semi-supervised methods (mainly FixMatch and ReMixMatch) outperformed self-supervised and supervised methods.
- Semi-supervised methods require task-specific adaptations, while self-supervised methods do not.
- Self-supervised methods' success highly depends on model architecture, as larger backbones had larger performance impact.
- Label Efficiency:
  - Classification: 10% of annotations achieve 95% performance
  - Segmentation: 50% of annotations achieve 90% performance

Conclusions

- Generally, Semi-supervised methods (mainly FixMatch and ReMixMatch) outperformed self-supervised and supervised methods.
- Semi-supervised methods require task-specific adaptations, while self-supervised methods do not.
- Self-supervised methods' success highly depends on model architecture, as larger backbones had larger performance impact.
- Label Efficiency:
    - Classification: 10% of annotations achieve 95% performance
    - Segmentation: 50% of annotations achieve 90% performance

IT IS POSSIBLE to reduce the need for data annotation!

Future Work

Memory Comparison

Compare memory requirements of different learning methods.

Autonomous Driving Tasks

Extend study to other AD tasks, such as object detection and segmentation using sensors like LIDAR.

Explore More Methods

Investigate broader range of semi-supervised and self-supervised methods (e.g., FlexMatch, S4L, SimSiam, SwAV).

U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Exploring Label Efficiency with Semi-Supervision and Self-Supervision Methods

Francisco Gonçalves Cerqueira

Supervisor
Ricardo Pereira de Magalhães Cruz

July 18, 2024
Dissertation
Master in Informatics and Computing Engineering

Appendix

Categories

Generative

Recreate realistic representations of unlabeled data by employing encoder-decoder architectures.

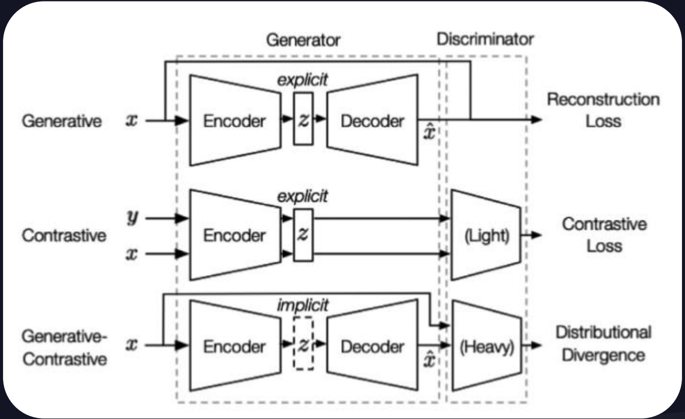
Requires substantial computational resources.

Contrastive

Contrast positive samples against a pool of negative or dissimilar samples.

Adversarial

Combines generative and contrastive elements.



Training Cycle

Supervised

Annotated samples are used.
Validation loss monitoring performance and selecting the best model to reduce overfitting.

Self-Supervised

Unannotated data is used, and validation loss is not employed.
Post-training, the model is fine-tuned on downstream tasks with labeled data.

Semi-Supervised

Labeled and unlabeled data are used together.
Must handle situations where one set is exhausted before the other.

Solution: Early Stop Cycle

End an epoch when the smaller set is exhausted, reducing training duration.Periodic shuffling increases sample diversity despite some data remaining unused.

Epoch	1		2		
Iteration	1	2	1	2	...
Labeled Batch	L1	L2	L1	L2	
Unlabeled Batch	U1	U2	U1	U2	

Training Cycle

Epoch	1			2			
Iteration	1	2	3	1	2	3	…
Labeled Batch	L1	L2	–	L1	L2	–	
Unlabeled Batch	U1	U2	U3	U1	U2	U3	

Epoch	1			2			
Iteration	1	2	3	1	2	3	…
Labeled Batch	L1	L2	L1	L1	L2	L1	
Unlabeled Batch	U1	U2	U3	U1	U2	U3	

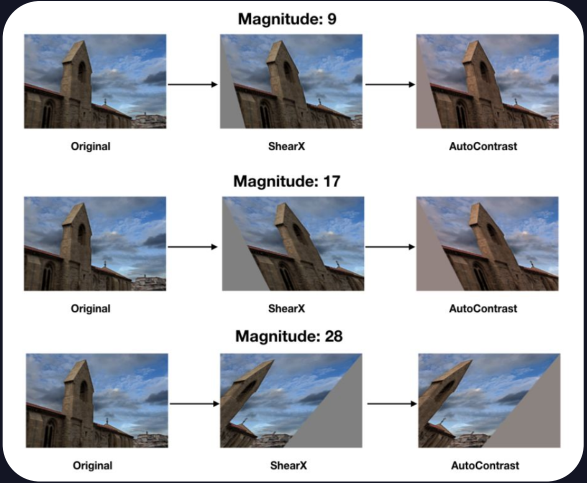
Fair Comparison

Method	Dataset	Epochs	Labeled BS	Unlabeled BS	EMA Decay
Supervised Learning	CIFAR-10	200	128	–	–
	SVHN	200	128	–	–
	Cityscapes	120	16	–	–
	KITTI	120	16	–	–
Semi-Supervised Learning	CIFAR-10	1000	16	112	0.999
	SVHN	1000	16	112	0.999
	Cityscapes	120	1	3	0.99
	KITTI	120	2	6	0.99
Self-Supervised Learning	CIFAR-10	200	–	128	–
	SVHN	100	–	128	–
	Cityscapes	100	–	16	–
	KITTI	100	–	16	–

Invariant RandAugment

RandAugment is a data augmentation technique that only utilizes two hyperparameters.

As RandAugment was a common data augmentation technique, a modified version of it was created to mitigate the mask consistency problem.



Removed:

- Shearing
- Translation
- Rotation
- Sharpness

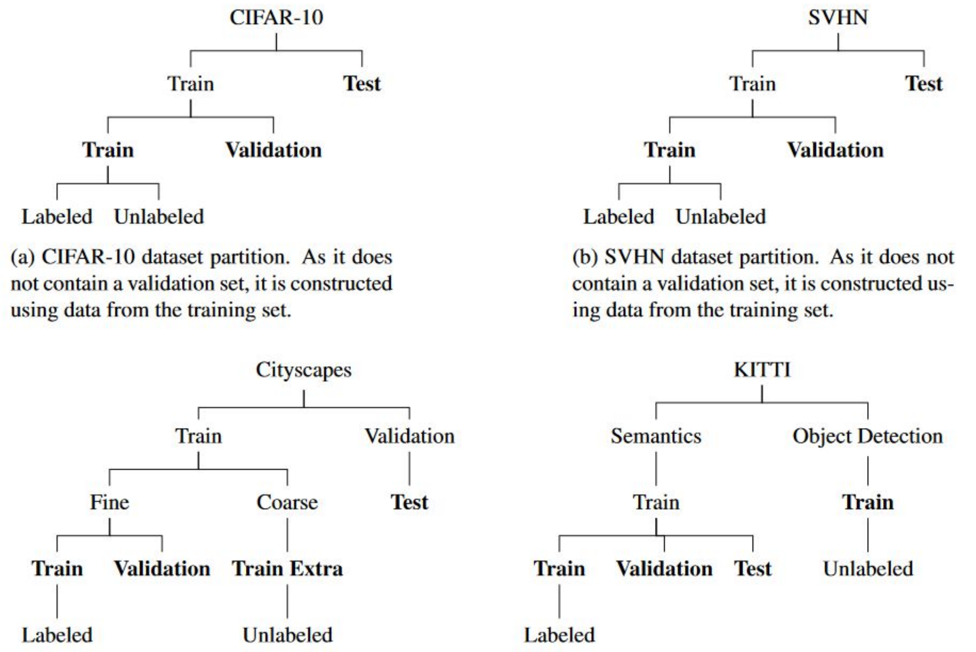
Maintained:

- Brightness
- Contrast
- Color Jittering
- Solarize
- Posterize
- Equalize

Code Structure

```
exploring-label-efficiency/
├── configs/  # holds the configuration files
│   ├── configs/  # configuration files for the experiments
│   ├── datasets.yaml
│   ├── losses.yaml
│   ├── metrics.yaml
│   ├── models.yaml
│   ├── optimizers.yaml
│   ├── schedulers.yaml
│   ├── selfsl_methods.yaml
│   ├── semisl_methods.yaml
│   └── stop_conditions.yaml
├── data/  # default directory for storing input data
├── docs/  # documentation files
├── logs/  # default directory for storing logs
├── src/  # source code
│   ├── core/  # contains the core functionalities
│   ├── datasets/  # contains the datasets
│   ├── methods/  # contains the SemiSL and SelfSL methods
│   ├── models/  # contains the models
│   ├── tools/  # scripts for training, testing, etc.
│   │   ├── selfsl_train.py
│   │   ├── semisl_train.py
│   │   ├── sl_train.py
│   │   └── test.py
│   ├── trainers/  # contains the trainer classes
│   └── utils/  # utility functions
├── weights/  # default directory for storing model weights
└── requirements.txt  # project dependencies
```

Datasets Partition Tree



(a) CIFAR-10 dataset partition. As it does not contain a validation set, it is constructed using data from the training set.

(b) SVHN dataset partition. As it does not contain a validation set, it is constructed using data from the training set.

(c) Cityscapes dataset partition. The train and validation sets derive from the "Fine" annotated set. An additional training set containing coarse annotated images was used as the unlabeled set.

(d) KITTI dataset partition. The train set was split into the train, validation, and test sets. The train set of the object detection benchmark was used as the unlabeled set.