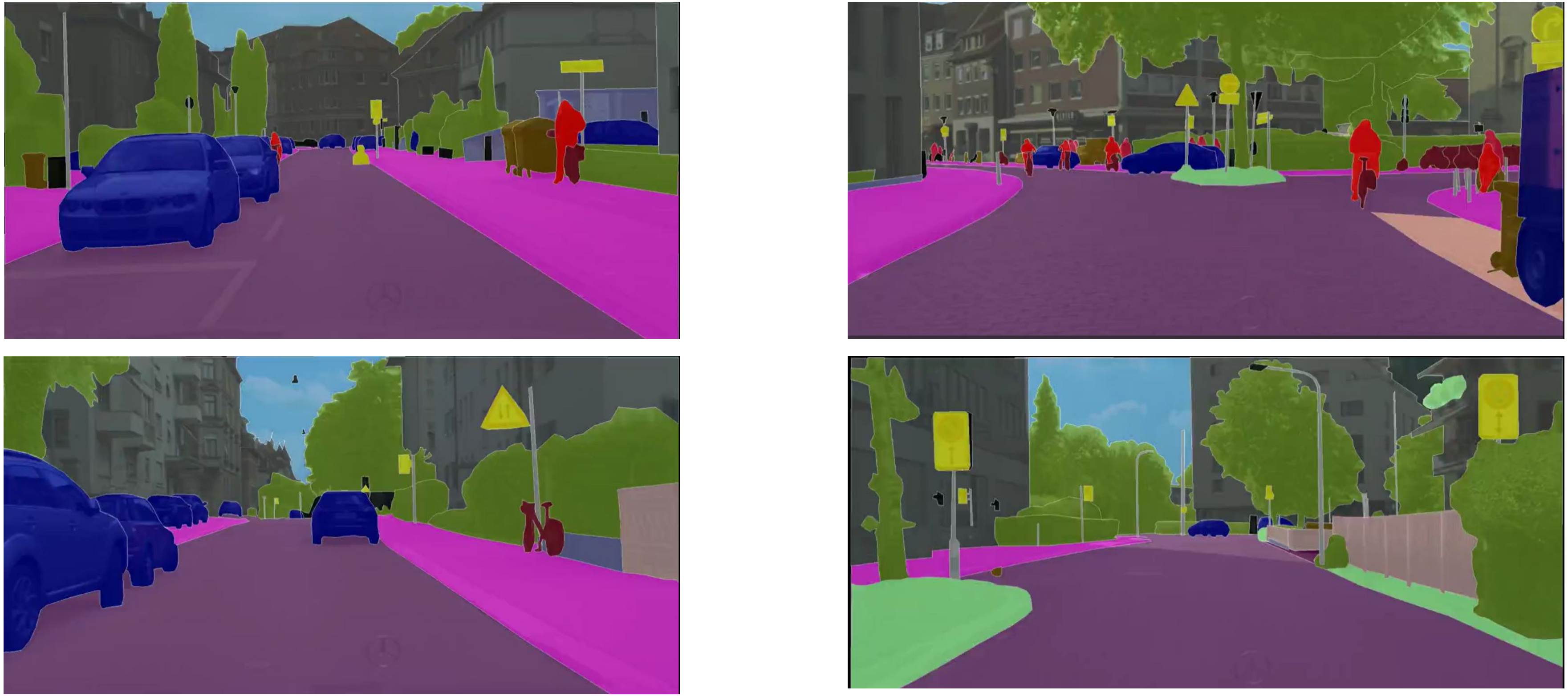
Are Labels Needed for Instance Incremental Learning?

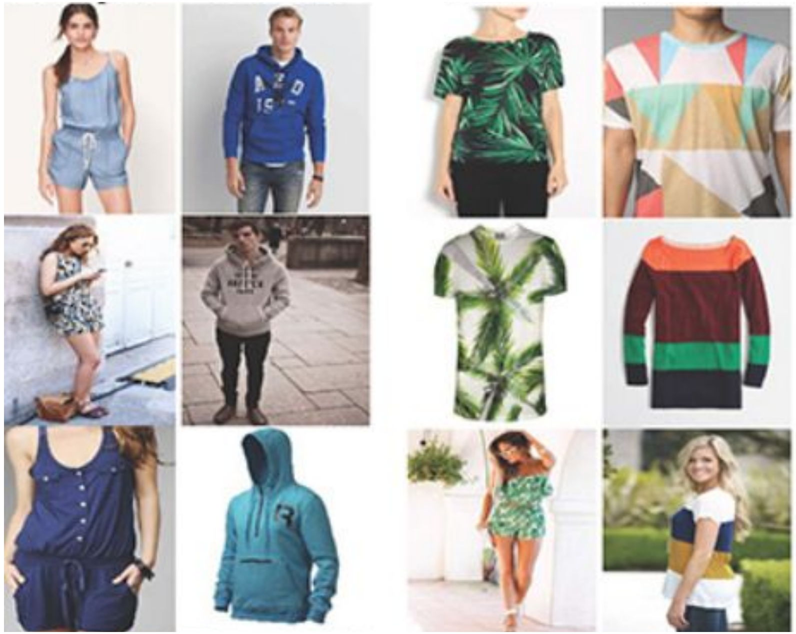
Visual Instances



Self-driving car: Same object, different instances (i.e. bikes)

Visual Instances

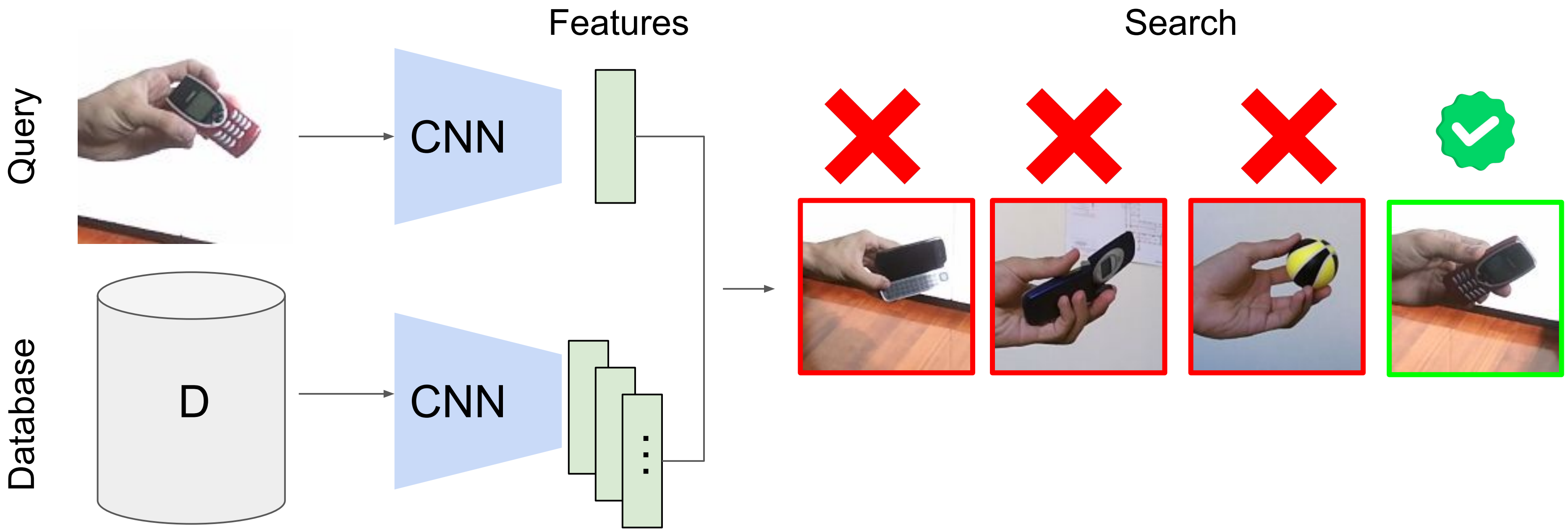
Fashion



Cars

Retail: Same object, different instances (i.e. Clothing, Car brands, etc.)

Visual Instance Learning

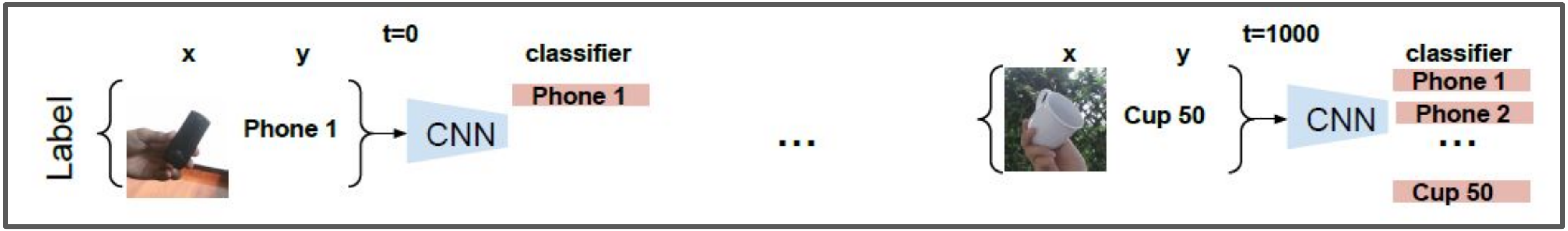


Visual instance learning aims to search for a given instance query in a database.

Visual Instance Learning

Instance learning is performed *offline*, however is unrealistic → Privacy → Incremental Learning.

Visual Incremental Instance Learning



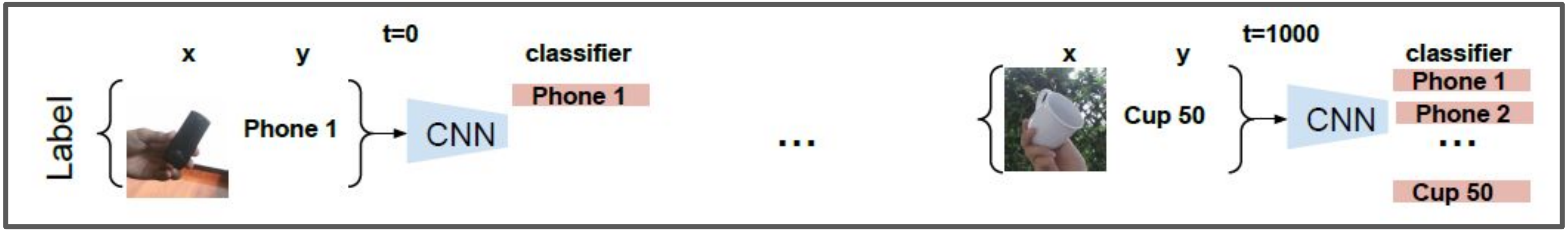
❌ Un-scalable

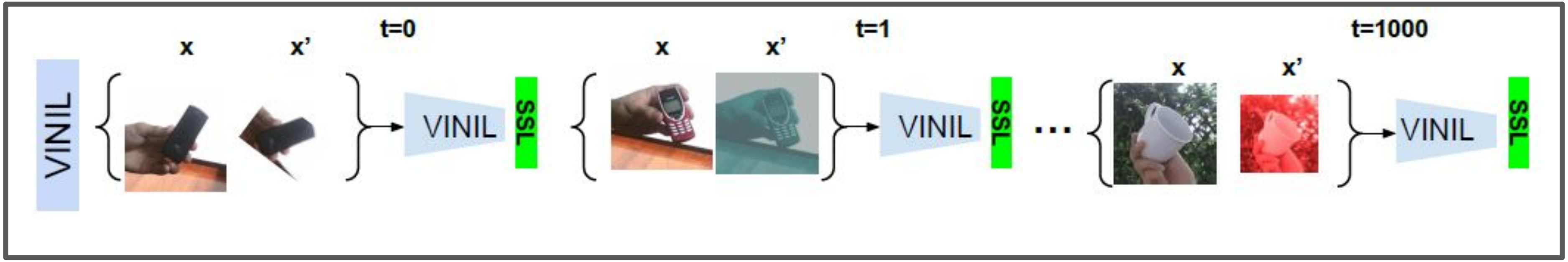
❌ Label-inefficient

❌ Forgetful

How can we learn instances in a scalable, label-free and less forgetful manner?

Visual Self-**In**cremental Instance **L**earning (VINIL)

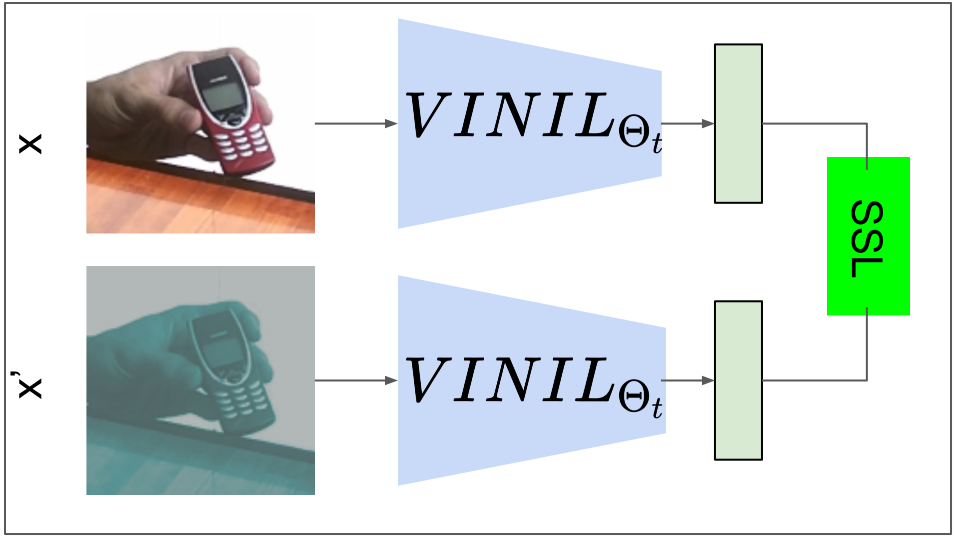




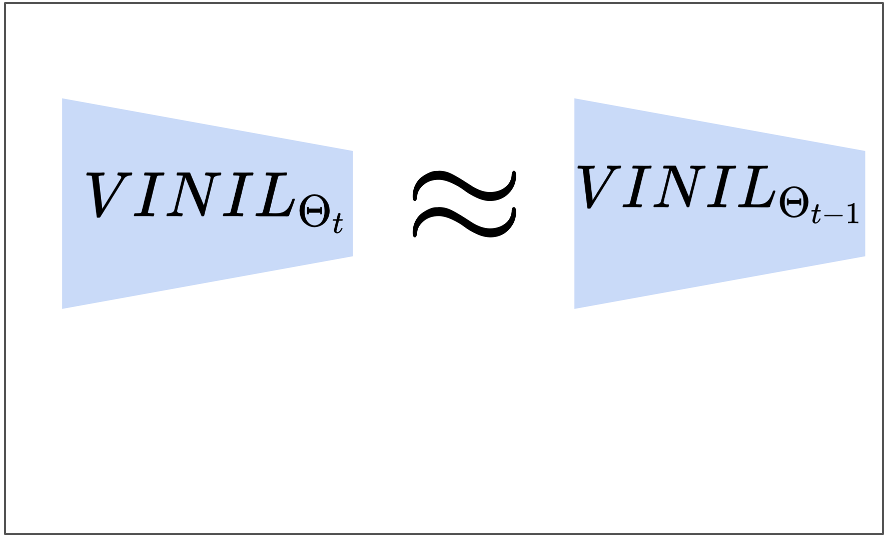
VINIL leverages **S**elf-**S**upervised **L**earning to be: Scalable and Label-free while forgetting less.

VINIL: Objective

Instance discrimination: Self-Supervised Learning



Incremental Learning: Regularization or Memory



$$\mathcal{L} = w * \mathcal{L}_{inst} + (1 - w) * \mathcal{L}_{incr}$$

VINIL: Implementation

Method	Supervision	Input	Memory	Loss
SGD	Label-supervised	(x,y)	n/a	$CE(y,y')$
SGD	Self-supervised	(x)	n/a	$BT(x,x')$
Replay	Label-supervised	(x,y)	(x^m,y^m)	$CE(y,y')+CE(y^m,y^{m'})$
Replay	Self-supervised	(x)	(x^m)	$BT(x,x')+BT(x^m,x^{m'})$
EwC	Label-supervised	(x,y)	n/a	$CE(y,y')+Reg(\Theta,y')$
EwC	Self-supervised	(x)	n/a	$BT(x,x')+Reg(\Theta)$

Instance Learning: Cross-Entropy **(CE)** with label-sup. | BarlowTwins **(BT)** with self-sup.

Incremental Learning: SGD (Fine-tuning) | Memory Replay | Elastic Weight Consolidation **(EwC)**

Experimental Setup

~Datasets~

Core-50: Hand-held Objects | 10 Categories | 50 Instances Per-category | 120k training & 45k test images

iLab-20M: Turntable Dataset | 10 Categories | 90 Instances Per-category | 125k training & 31k test images

~Metrics~

Accuracy: Top-k retrieval, **Forgetfulness**: Drop in Accuracy across Learning Sessions.

~Protocol~

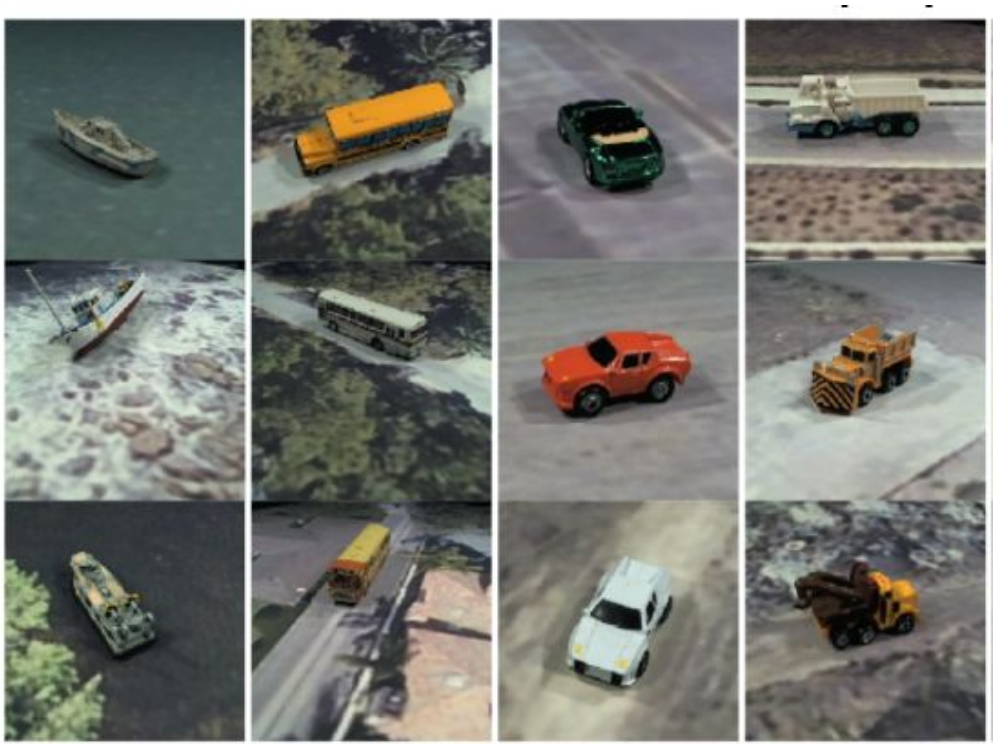
Tasks: 5 main tasks (2 categories each, bus, car, etc.) | N-instances per task (i.e. 100 for Core-50)

k-NN: All methods are evaluated via k-NN (N=100) | Activations of Last ResNet-18 Layer (layer4)

Datasets

Core-50

iLab-20M



Exp 1. How Does VINIL Compare to Label-supervision?

Method	Supervision	Core-50		iLab-20M	
		Accuracy (↑)	Forgetting (↓)	Accuracy (↑)	Forgetting (↓)
SGD	Label	71.450	22.436	89.340	6.500
SGD	VINIL	**74.914**	**4.802**	**90.398**	**0.000**
Replay	Label	**88.180**	**6.741**	84.464	5.696
Replay	VINIL	67.677	10.095	**90.543**	**0.000**
EwC	Label	**75.117**	18.268	87.690	4.535
EwC	VINIL	73.011	**2.167**	**90.655**	**0.000**

VINIL is more accurate (in **4/6** settings) and much less forgetful (in **5/6** settings) *without* using any labels.

Label-supervised variant leverages memory, whereas VINIL is distracted by memory.

Exp 2. Can VINIL Generalize Across Datasets?

	Train on⟹	Core-50	iLab-20M		iLab-20M	Core-50	
	Test on⟹	Core-50	Core-50		iLab-20M	iLab-20M	
Method	Supervision	Accuracy	Accuracy	%Δ(↓)	Accuracy	Accuracy	%Δ(↓)
SGD	Label	71.450	59.850	16	89.340	67.249	24
SGD	VINIL	**74.914**	**66.704**	**10**	**90.398**	**76.302**	**15**
Replay	Label	**88.180**	55.692	36	84.464	69.412	17
Replay	VINIL	67.677	**61.857**	**8**	**90.543**	**76.125**	**15**
EwC	Label	**75.117**	59.030	21	87.690	70.087	20
EwC	VINIL	73.011	**70.648**	**3**	**90.655**	**75.793**	**16**

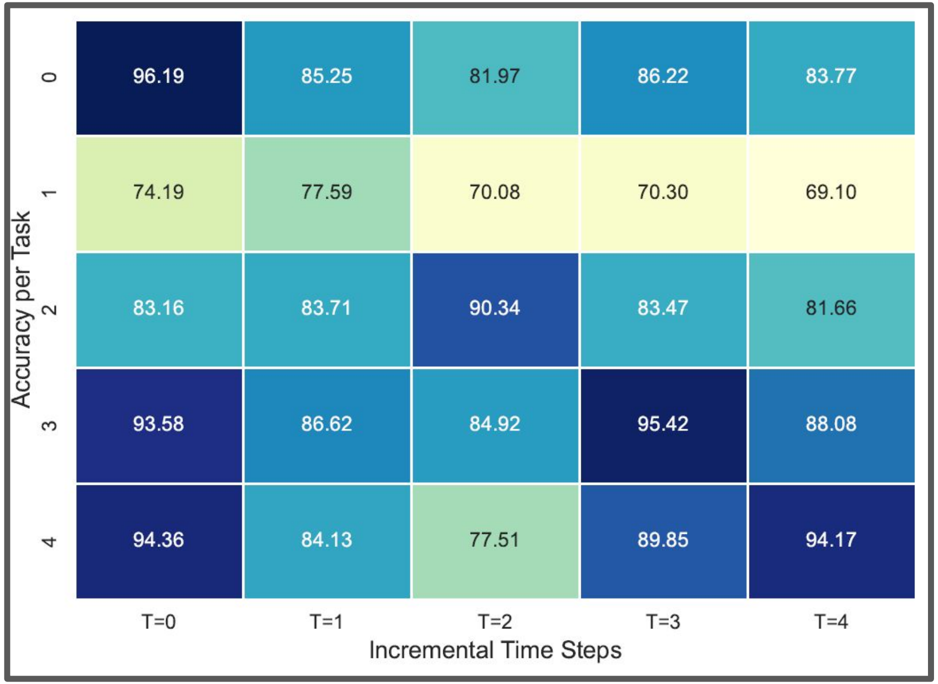
VINIL learns more generalizable feature space, exhibiting much lower drop in accuracy.

Label-supervision *overfits* with memory to the training source (**36**% relative drop rate!).

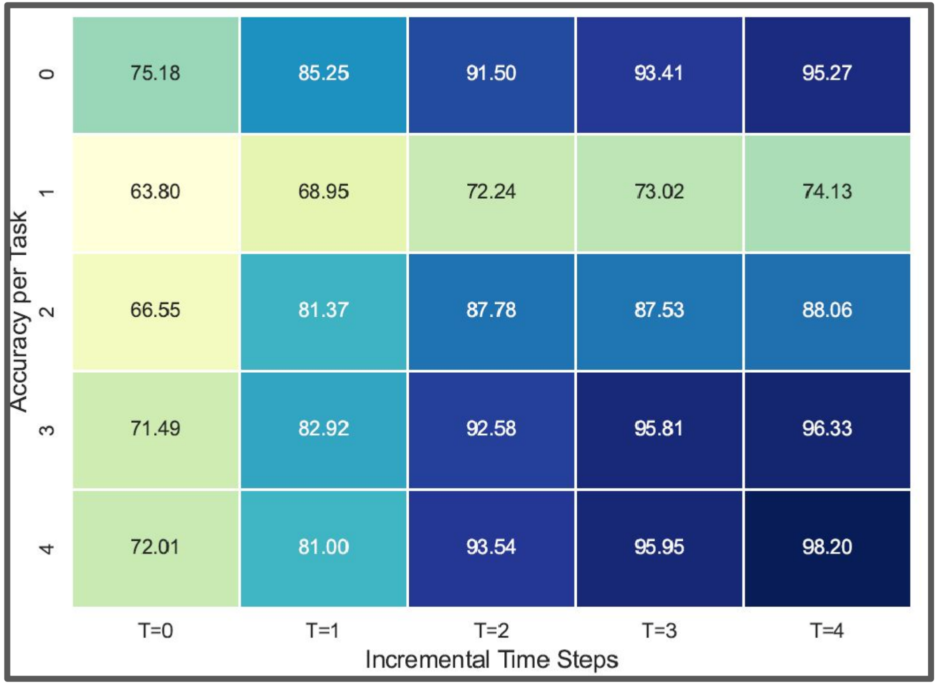
Why Does VINIL Perform Well?

Analysis 1: VINIL Leverages Incoming Stream of Tasks

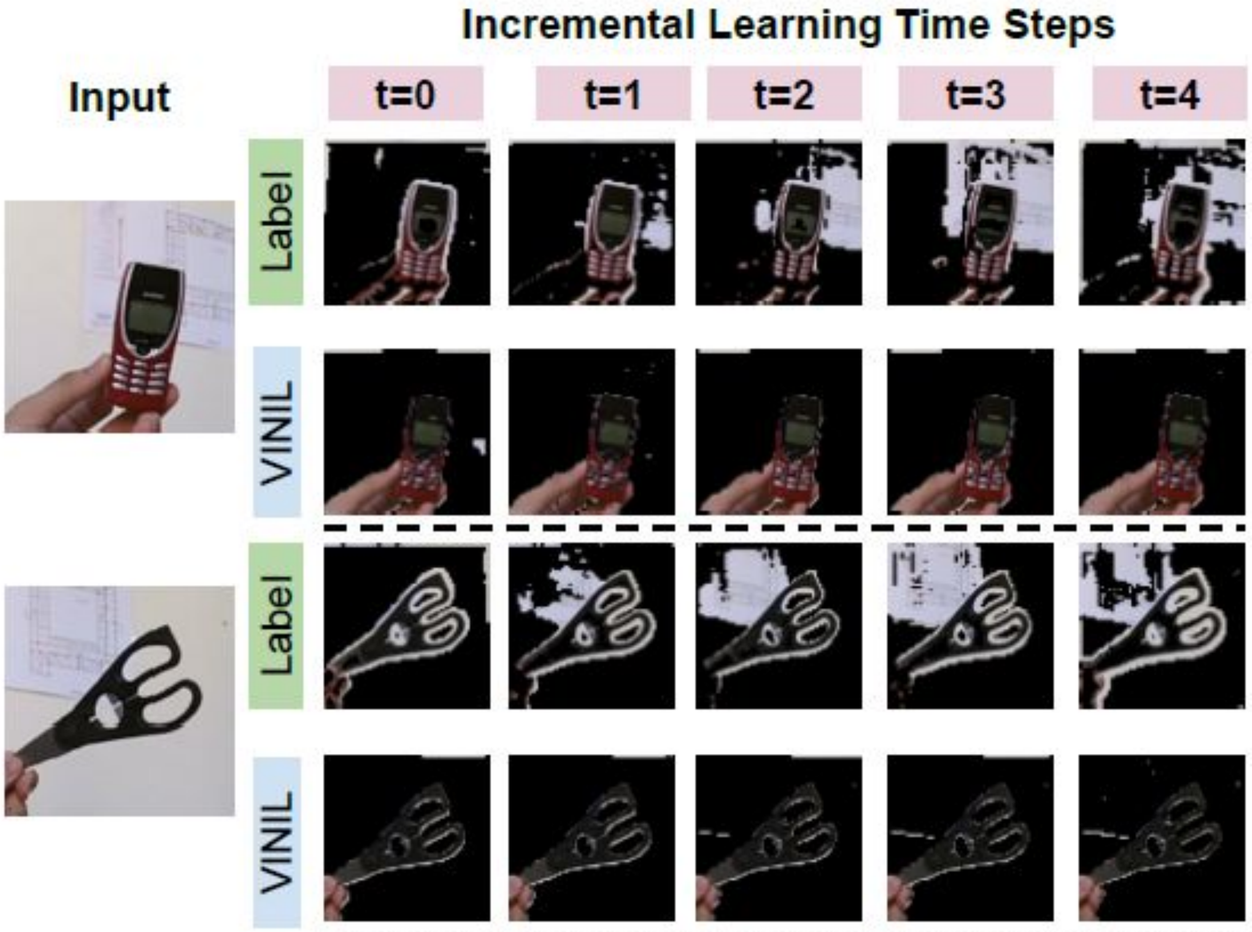
Label (SGD)



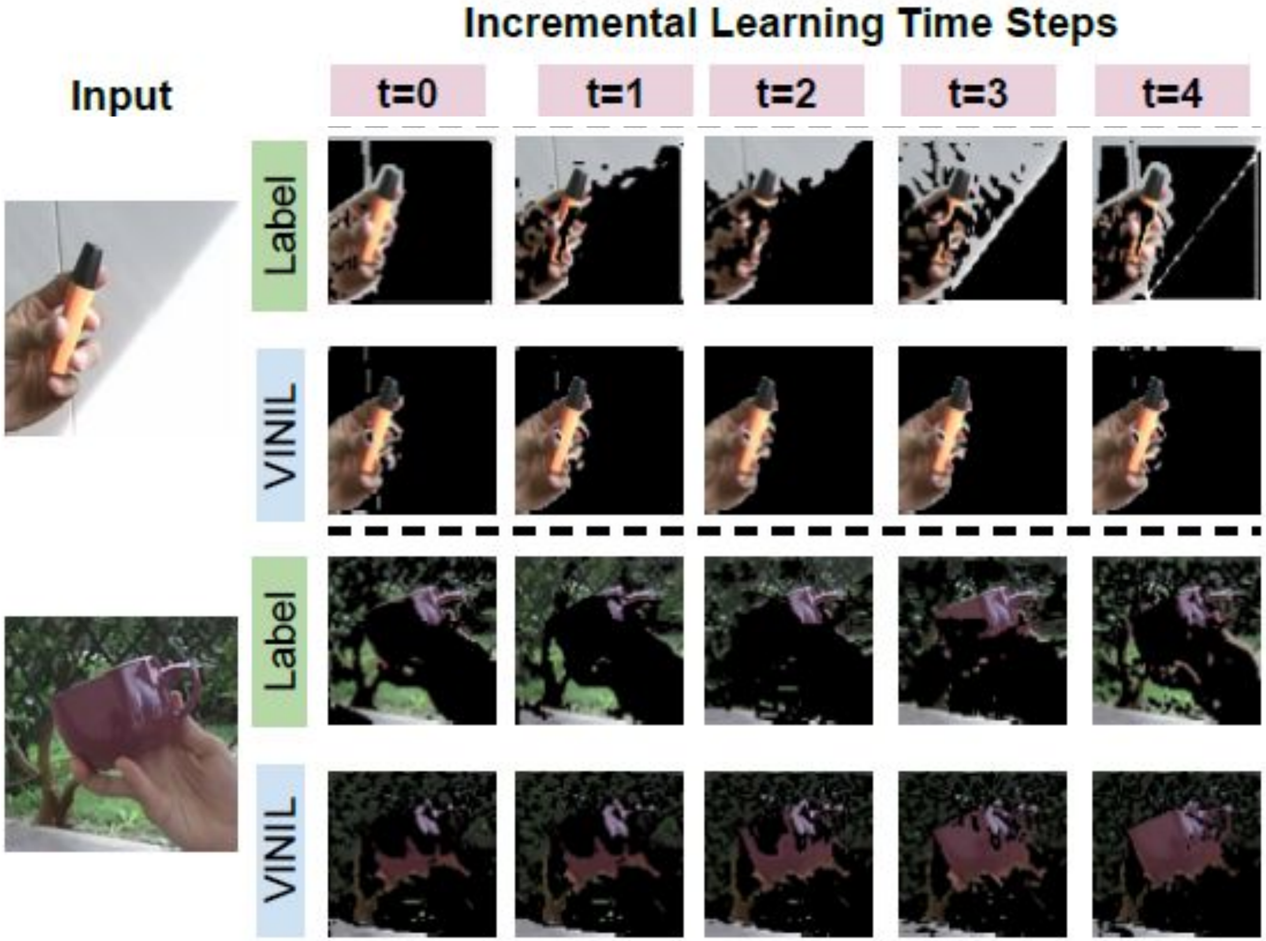
VINIL (SGD)



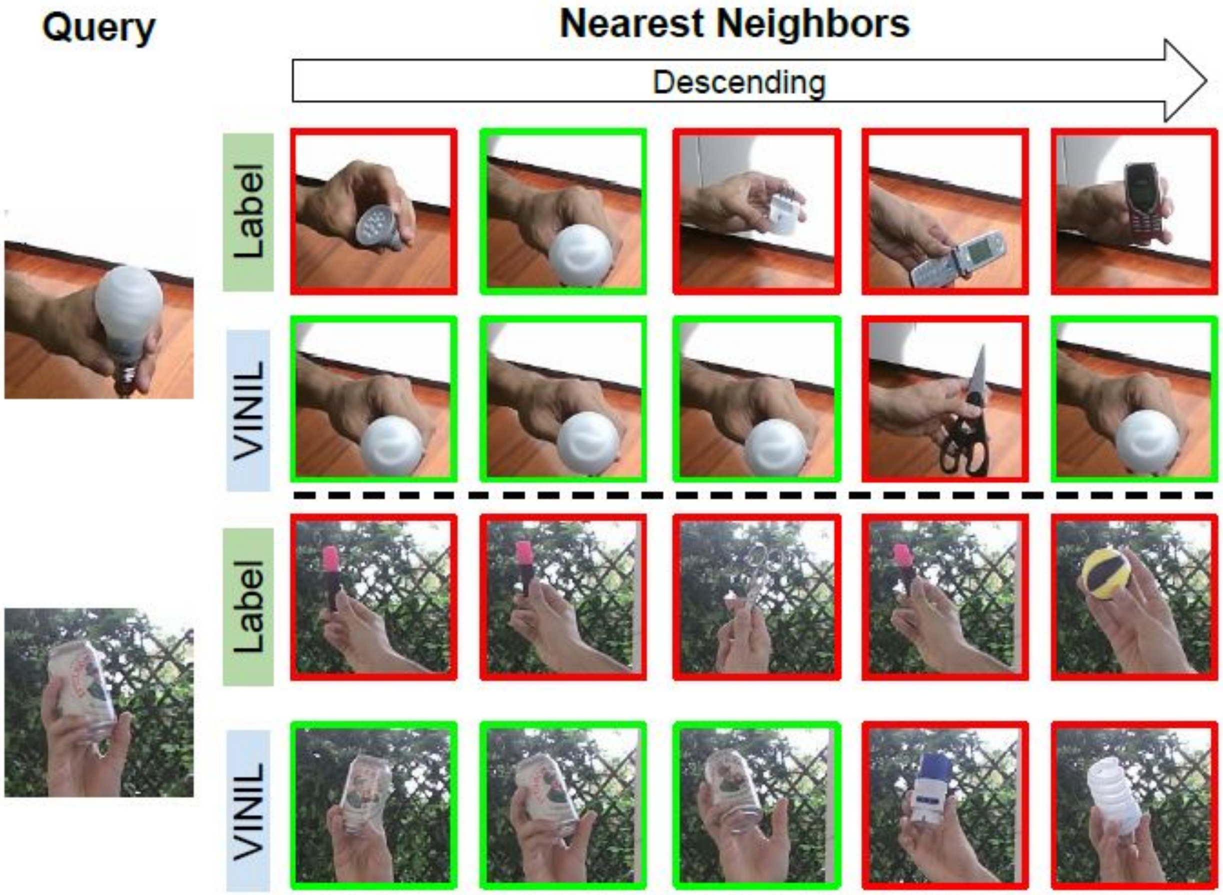
Analysis 2: VINIL Focuses on the Object



Analysis 2: VINIL Focuses on the Object



Analysis 3: VINIL Maintains Instance-level Variation



Summary

We proposed VINIL: A self-incremental visual instance learner.

VINIL is more scalable, generalizable, label-free and less forgetful in comparison to label-supervision.

VINIL does so by accumulating representations and focusing on instance-level variation.