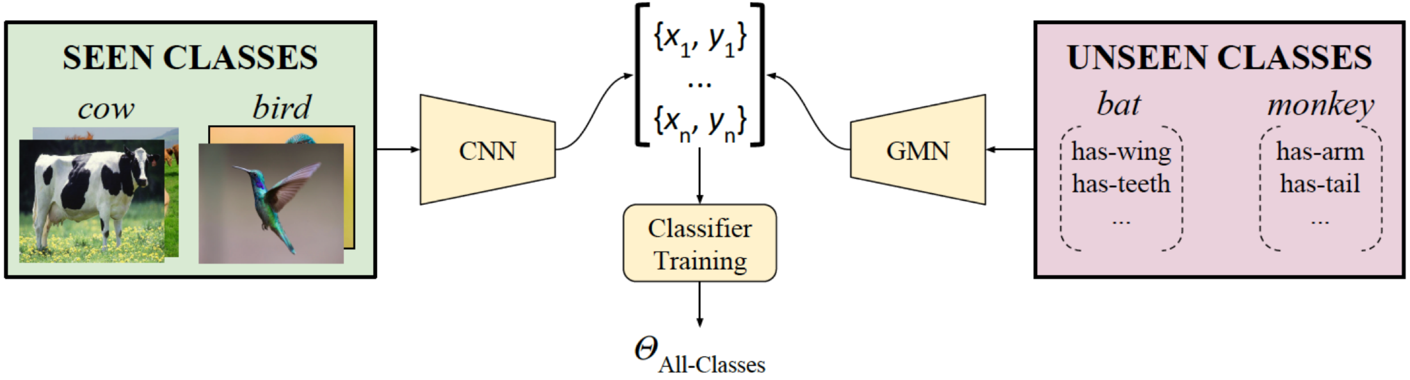
Bilim Akademisi - Bilkent Üniversitesi
Yapay Öğrenme Yaz Okulu 2020

Gökberk Cinbiş

2. Kısım
Eksik Gözetimli Öğrenme

ORTA DOĞU TEKNİK ÜNİVERSİTESİ
MIDDLE EAST TECHNICAL UNIVERSITY
ImageLab

Part 1: Gradient Matching Networks



IEEE / CVF Conf. on Computer Vision and Pattern Recognition (CVPR), June 2019

Part 2: Zero-shot Detection and Captioning of Unseen Objects

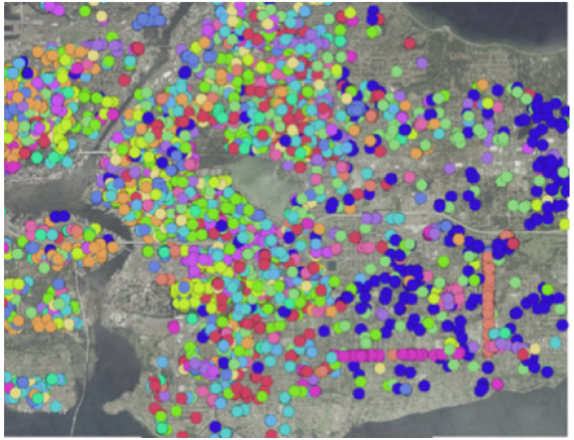
♦: A couple of **elephants** standing next to each other.
★: A couple of **zebra** standing next to each other.

♦: A piece of **cake** on a white plate.
★: A piece of **pizza** on a white plate.

British Machine Vision Conference (BMVC), September 2018

British Machine Vision Conference (BMVC), September 2019

Part 3: Do we really need ZSL in practice?



IEEE Trans. On Geoscience and Remote Sensing, January 2018

IEEE Trans. On Geoscience and Remote Sensing, July 2019

British Machine Vision Conference (BMVC), September 2019

Part 4: Partially-supervised domain transfer for face recognition in the *wildest*

5

Part I

Gradient Matching Networks

Zero-shot object recognition

Seen Classes

cow *bird*

Unseen Classes

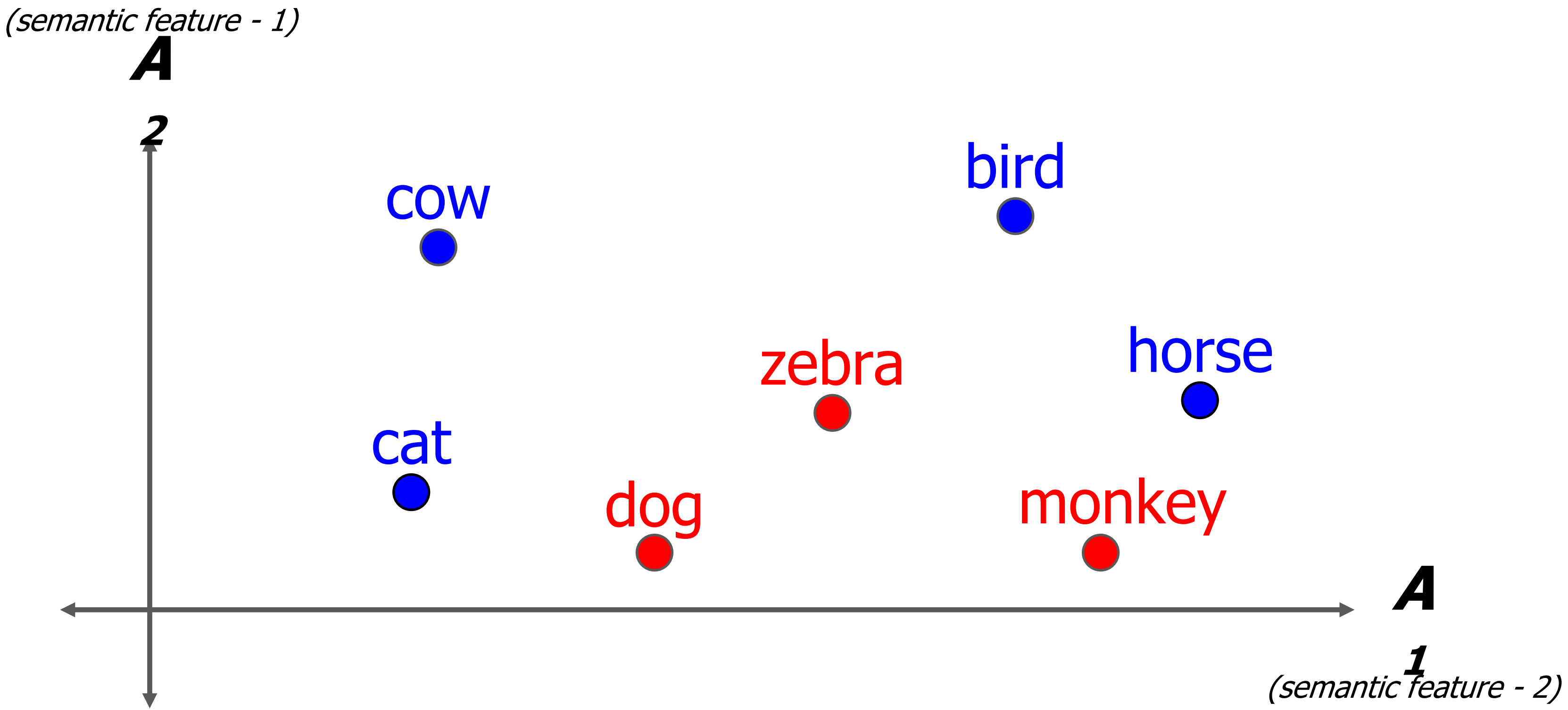
bat *monkey*

~~Training samples~~

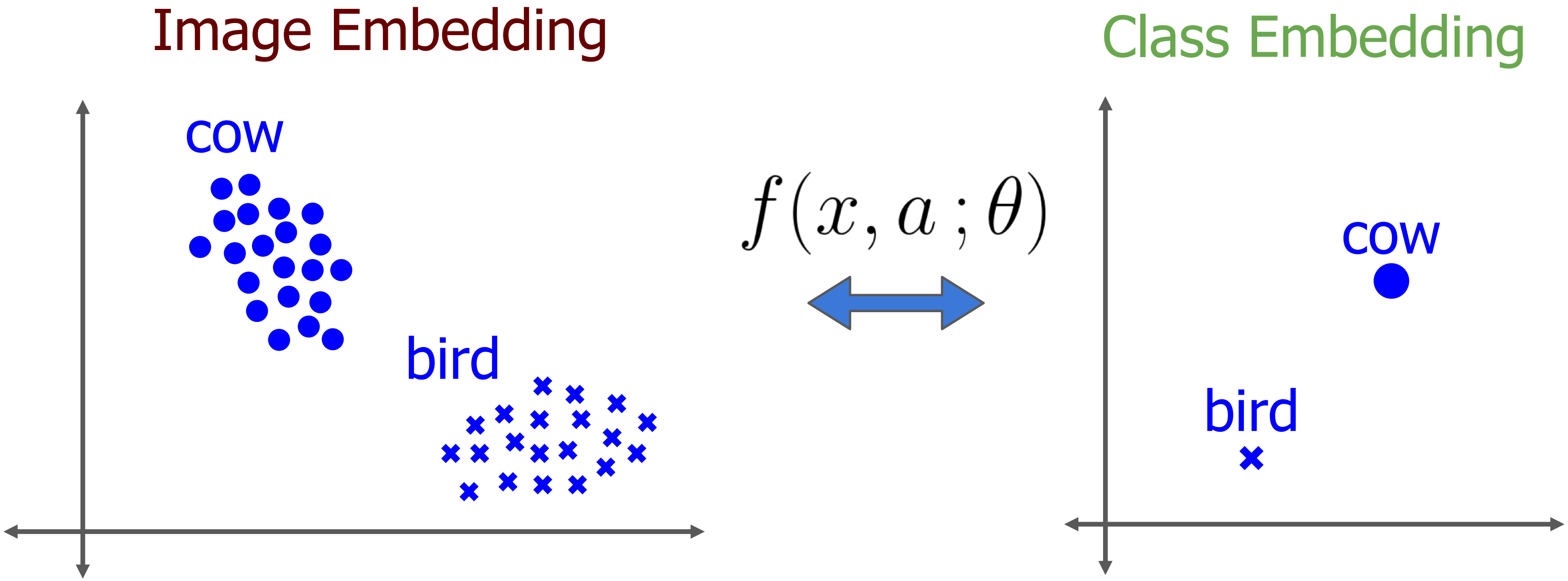
i - Learn a classification model on **seen** classes

ii - Use the model for **both** sets

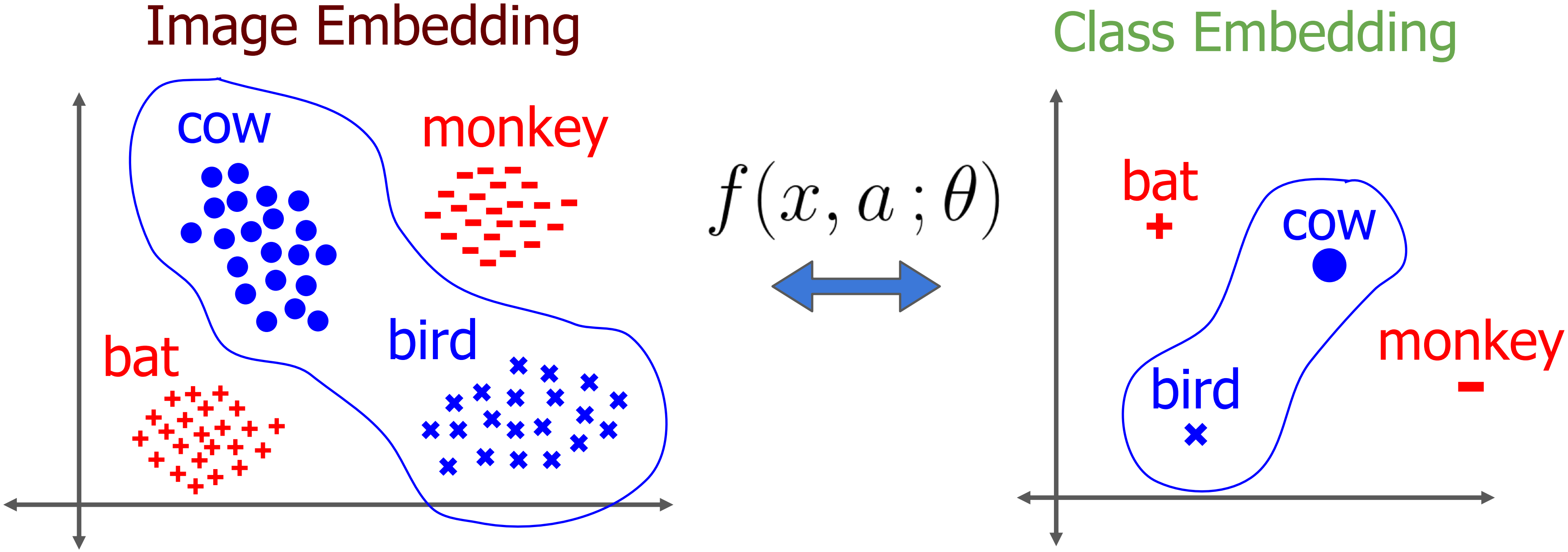
Semantic Class Embedding Space



Mainstream approach

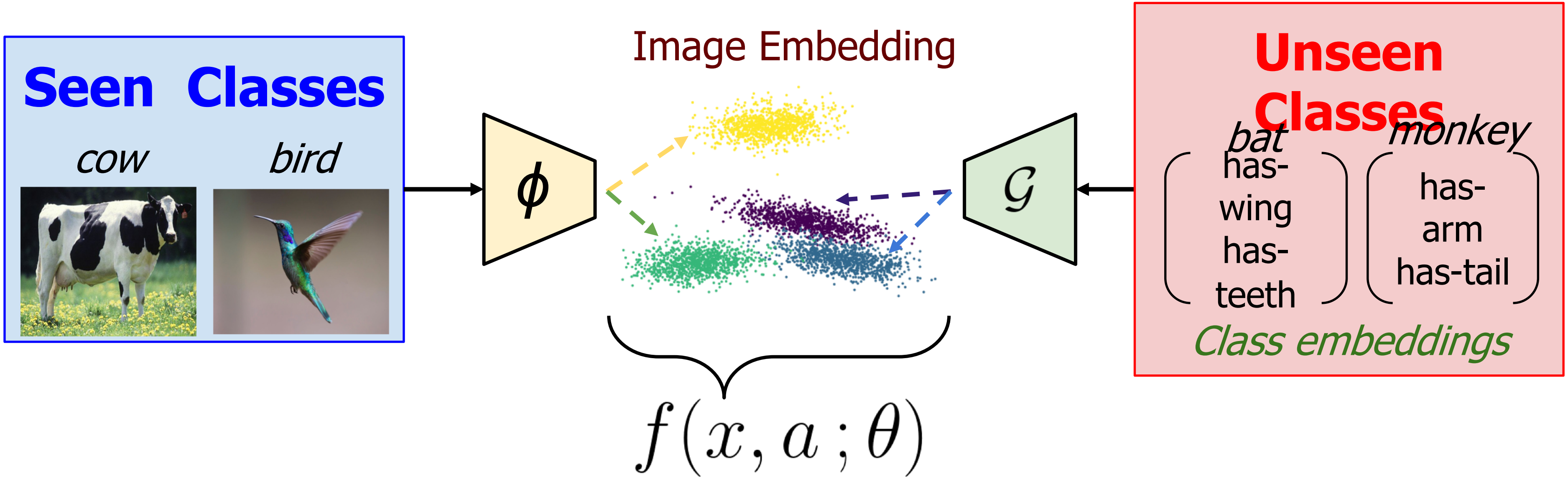


A weakness in purely discriminative approaches



Akata et al. "Label-embedding for attribute-based classification." CVPR 2013.

Generative-model-based approaches



Examples:

- Xian et al. "Feature generating networks for zero-shot learning." CVPR 2018.
- Verma et al. "Generalized zero-shot learning via synthesized examples." CVPR 2018.

First attempt: conditional GAN

A naive idea: *just train a conditional GAN model* (or another implicit generative model), which takes concat(noise,class-embedding) as the input.

First attempt: conditional GAN

A naive idea: *just train a conditional GAN model* (or another implicit generative model), which takes concat(noise,class-embedding) as the input.

.. but there are three important inter-connected challenges:

- **Semantics:** How do we enforce producing samples that truly belong to the target class?
- **Variance:** How do we enforce producing a variety of samples for a given embedding?
- **Data quality:** How do we make sure that the resulting training examples is actually useful? (ie. will the classifier trained over them be accurate?)

A second attempt

Train a conditional GAN using **GAN loss + loss function of a classifier** over the training classes.

At test time: simply synthesize training examples by feeding class-embeddings of test (unseen) classes to the GAN model.

Good: can leverage unsupervised data through the GAN loss.
Good: can enforce generating examples that are classified to the right class.

A second attempt

Train a conditional GAN using **GAN loss + loss function of a pre-trained classifier** over the training classes.

At test time: simply synthesize training examples by feeding class-embeddings of test (unseen) classes to the GAN model.

Good: can leverage unsupervised data through the GAN loss.
Good: can enforce generating examples that are classified to the right class.

However,

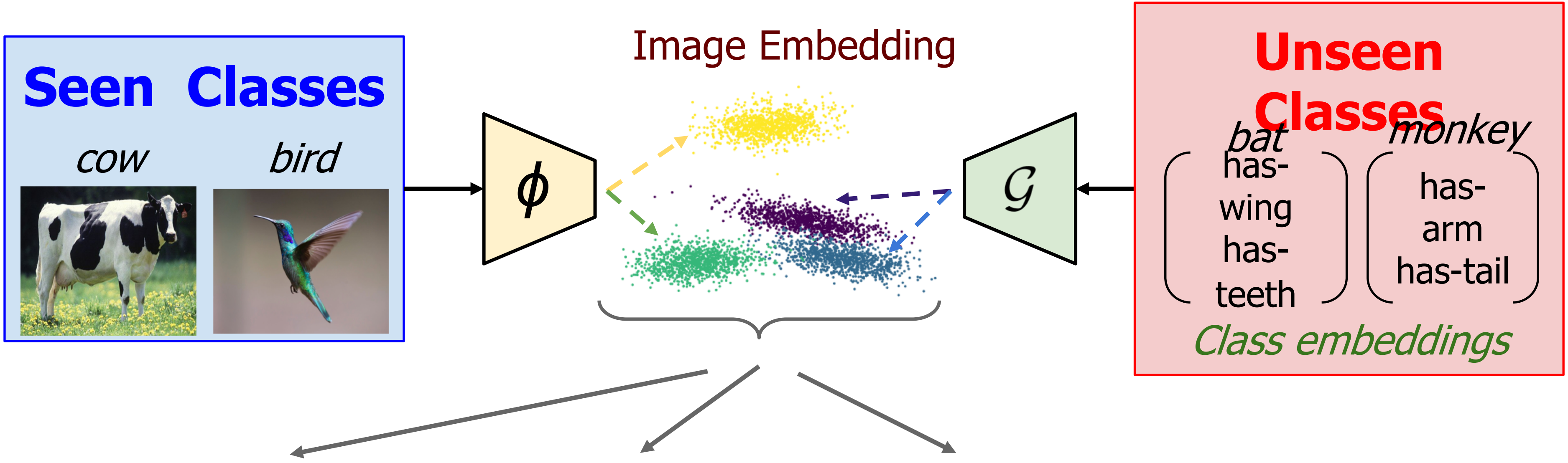
- The generated samples are not necessarily informative (**like support vectors**) ones (Likely, the generative model will learn to synthesize the "easy" samples.)
- The generated samples may contain **artifacts** detrimental for training purposes.

3rd attempt: a meta-learning approach

Assume that the synthetic (+real) examples will be used to train a classifier using a first-order gradient optimization technique.

3rd attempt: a meta-learning approach

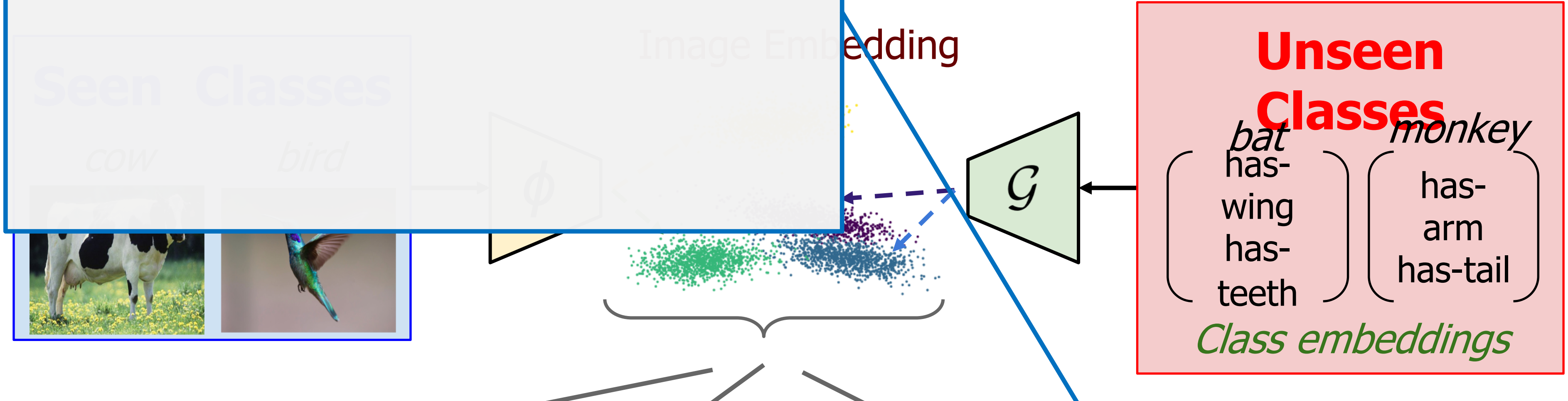
Assume that the synthetic (+real) examples will be used to train a classifier using a first-order gradient optimization technique.



$$\theta^{(0)} \xrightarrow{-\nabla_\theta \mathcal{L}} \theta^{(1)} \xrightarrow{-\nabla_\theta \mathcal{L}} \cdots \xrightarrow{-\nabla_\theta \mathcal{L}} f(x, a; \theta^{(N)})$$

Can we **optimize *G* such that we minimize the loss of final *f*** ?

Assume that the synthetic (+real) examples will be used to train a classifier using a first-order gradient optimization technique.



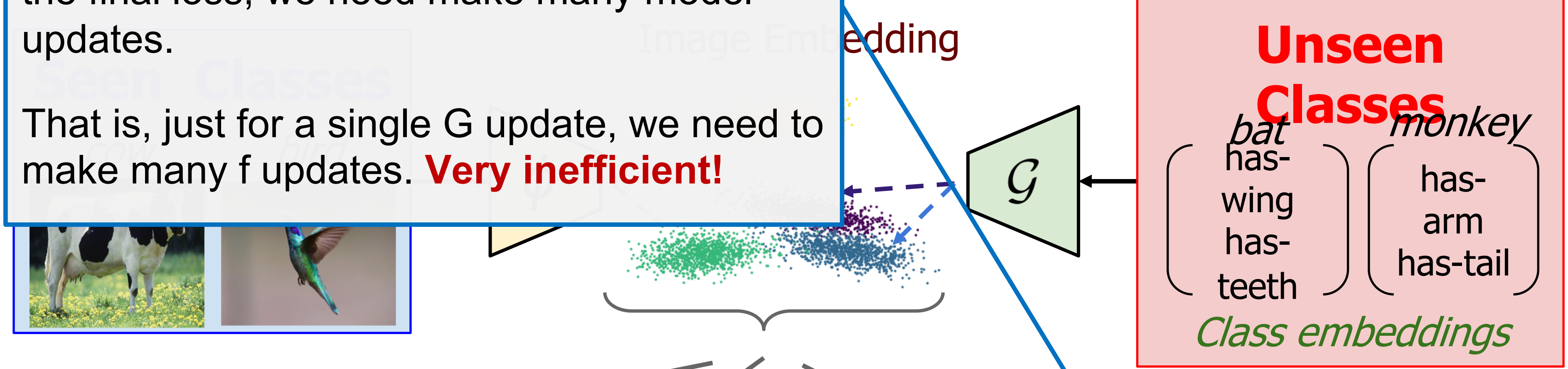
$$\theta^{(0)} \xrightarrow{-\nabla_\theta \mathcal{L}} \theta^{(1)} \xrightarrow{-\nabla_\theta \mathcal{L}} \cdots \xrightarrow{-\nabla_\theta \mathcal{L}} f(x, a; \theta^{(N)})$$

Can we **optimize *G* such that we minimize the loss of final *f*** ?

Not naively. To measure the impact of G on the final loss, we need make many model updates.

That is, just for a single G update, we need to make many f updates. **Very inefficient!**

will be used to train a classifier using



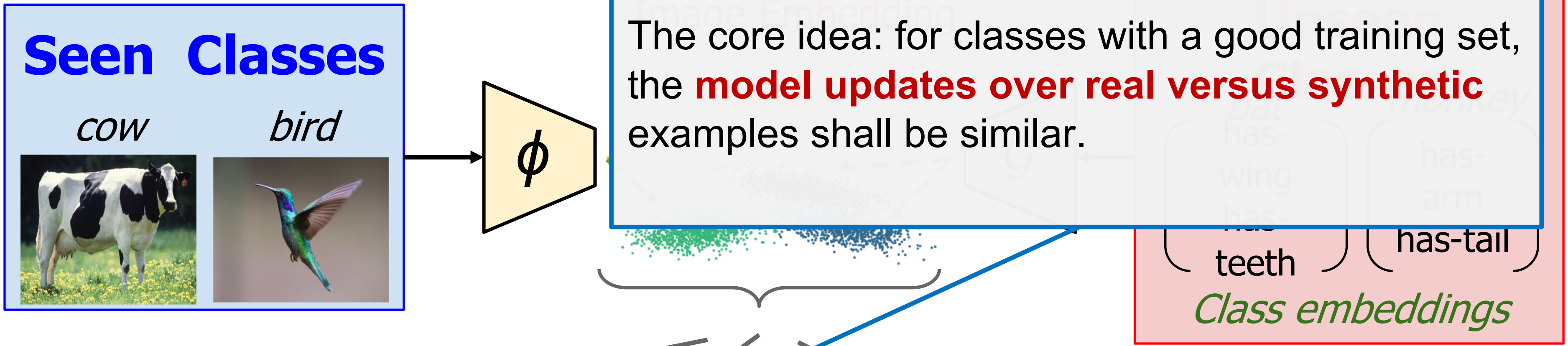
$$\theta^{(0)} \xrightarrow{-\nabla_\theta \mathcal{L}} \theta^{(1)} \xrightarrow{-\nabla_\theta \mathcal{L}} \cdots \xrightarrow{-\nabla_\theta \mathcal{L}} f(x, a; \theta^{(N)})$$

Our idea

Assume that the synthetic (+real) examples will be used to train a classifier using a first-order gradient optimization technique.

Focus on learning to generate examples that maximizes the correctness of **individual model updates.**

The core idea: for classes with a good training set, the **model updates over real versus synthetic** examples shall be similar.



$$\theta^{(0)} \xrightarrow{-\nabla_\theta \mathcal{L}} \theta^{(1)} \xrightarrow{-\nabla_\theta \mathcal{L}} \cdots \xrightarrow{-\nabla_\theta \mathcal{L}} f(x, a; \theta^{(N)})$$

Gradient matching loss

$$\mathcal{L}_{\text{GM}} = \mathbb{E}_{\theta}\left[1 - \frac{g_r(\theta)^T g_f(\theta)}{||g_r(\theta)||_2 \; ||g_f(\theta)||_2}\right]$$

Gradient by real

$$g_r(\theta) = \mathbb{E}_{(x,a)\sim p_{\text{data}}}\left[\nabla_\theta \mathcal{L}(x, a, f_\theta)\right]$$

Gradient by generated

$$g_f(\theta) = \mathbb{E}_{\tilde{x}\sim\mathcal{G}(z,a),\, a\sim p_{\text{data}}}\left[\nabla_\theta \mathcal{L}(\tilde{x}, a, f_\theta)\right]$$

To approximate the expectation over θ

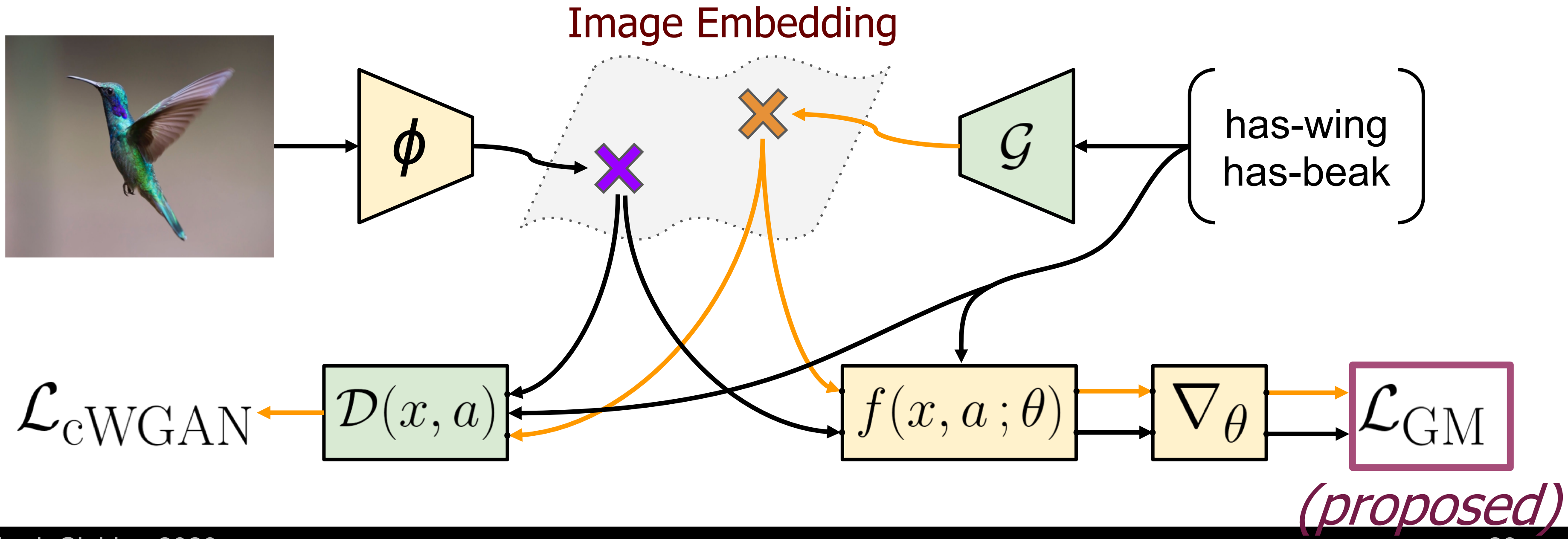
$$\mathcal{L}_{\mathrm{GM}} = \underline{\mathbb{E}_\theta} \left[1 - \frac{g_r(\theta)^T g_f(\theta)}{||g_r(\theta)||_2 \; ||g_f(\theta)||_2} \right]$$

Repeatedly:

- train the classification model **N** epochs,
- re-initialize all parameters and reset the optimizer state.

Gradient matching network (GMN)

Gradient matching loss
+ adversarial loss
(can be used for unsupervised learning)



Experiments - Datasets

- Caltech-UCSD Birds-200-2011 (**CUB**) - 200 bird species - 12k

- SUN Attribute (**SUN**) - 717 scene categories - 14k

- Animals with Attributes (**AWA**) - 50 animal categories - 30k

Wah et al. “The Caltech-UCSD Birds-200-2011 Dataset”, 2011.
Patterson et al. “Sun attribute database: Discovering, annotating, and recognizing scene attributes” CVPR, 2012.
Lampert et al. “Attribute-based classification for zero-shot visual object categorization” TPAMI, 2013.

Evaluation Metrics

Normalized score (NS) : average of the top-1 per-class scores

- **T-1** : NS of unseen classes in ZSL setting
- **u**: NS of unseen classes in GZSL setting
- **s**: NS of seen classes in GZSL setting
- **h:** harmonic mean of **u** and **s** $\frac{2 \times \mathbf{u} \times \mathbf{s}}{\mathbf{u} + \mathbf{s}}$

Zero-shot prediction (unseen classes)

				CUB	SUN	AWA
				T-1	T-1	T-1
1	*Zhang et al.* '18			52.6	61.7	67.4
2	*Bucher et al.* '17			57.8	60.4	66.3
3	*Xian et al.* - `DEVISE` '18			60.3	60.9	66.9
4	*Xian et al.* - `ALE` '18			61.5	62.1	68.2
5	*Xian et al.* - `Softmax` '18			57.3	60.8	68.2
6	*Verma et al.* '18			59.6	63.4	69.5
7	*Felix et al.* - `cycle-WGAN` '18			57.8	59.7	65.6
8	*Felix et al.* - `cycle-CLSWGAN` '18			58.4	60.0	66.3
9	Bilinear	LN	$\mathcal{L}^{\mathrm{S}}_{\mathrm{cWGAN}}$	61.7	62.7	67.3
10	Bilinear	LN	$\mathcal{L}^{\mathrm{S}}_{\mathrm{cWGAN}} + \mathcal{L}_{\mathrm{CLS}}$	61.9	62.7	66.4
11	Bilinear	LN	$\mathcal{L}^{\mathrm{S}}_{\mathrm{cWGAN}} + \mathcal{L}_{\mathrm{CYCLE}}$	62.2	62.7	68.2
12	Bilinear	LN	$\mathcal{L}^{\mathrm{S}}_{\mathrm{cWGAN}} + \mathcal{L}_{\mathrm{GM}}$ (*Ours*)	**67.0**	**63.6**	**72.0**
13	Linear	LN	$\mathcal{L}^{\mathrm{S}}_{\mathrm{cWGAN}} + \mathcal{L}_{\mathrm{GM}}$ (*Ours*)	63.1	58.9	70.1
14	Bilinear	AC	$\mathcal{L}^{\mathrm{S}}_{\mathrm{cWGAN}} + \mathcal{L}_{\mathrm{GM}}$ (*Ours*)	65.7	62.6	69.7
15	Linear	AC	$\mathcal{L}^{\mathrm{S}}_{\mathrm{cWGAN}} + \mathcal{L}_{\mathrm{GM}}$ (*Ours*)	63.8	61.1	66.8

Generalized zero-shot prediction (seen + unseen classes)

				CUB			SUN			AWA		
				u	s	h	u	s	h	u	s	h
1	*Zhang et al.* '18			31.5	40.2	35.3	41.2	26.7	32.4	38.7	74.6	51.0
2	*Bucher et al.* '17			28.8	55.7	38.0	40.5	37.2	38.8	2.3	**90.2**	4.5
3	*Xian et al.* - `DEVISE` '18			52.2	42.4	46.7	38.4	25.4	30.6	35.0	62.8	45.0
4	*Xian et al.* - `ALE` '18			40.2	59.3	47.9	41.3	31.1	35.5	47.6	57.2	52.0
5	*Xian et al.* - `Softmax` '18			43.7	57.7	49.7	42.6	36.6	39.4	57.9	61.4	59.6
6	*Verma et al.* '18			41.5	53.3	46.7	40.9	30.5	34.9	56.3	67.8	61.5
7	*Felix et al.* - `cycle-WGAN` '18			46.0	60.3	52.2	48.3	33.1	39.2	56.4	63.5	59.7
8	*Felix et al.* - `cycle-CLSWGAN` '18			45.7	61.0	52.3	49.4	33.6	40.0	56.9	64.0	60.2
9	Bilinear	LN	$\mathcal{L}^{S}_{\text{cWGAN}}$	45.6	59.2	51.5	50.6	30.3	37.3	53.5	72.0	61.4
10	Bilinear	LN	$\mathcal{L}^{S}_{\text{cWGAN}} + \mathcal{L}_{\text{CLS}}$	45.5	58.9	51.4	50.6	30.3	37.3	52.7	71.0	60.5
11	Bilinear	LN	$\mathcal{L}^{S}_{\text{cWGAN}} + \mathcal{L}_{\text{CYCLE}}$	51.1	54.9	52.9	50.6	30.3	37.3	55.4	70.1	61.8
12	Bilinear	LN	$\mathcal{L}^{S}_{\text{cWGAN}} + \mathcal{L}_{\text{GM}}$ *(Ours)*	**54.7**	58.4	**56.5**	42.5	35.5	38.7	**61.1**	71.3	65.8
13	Linear	LN	$\mathcal{L}^{S}_{\text{cWGAN}} + \mathcal{L}_{\text{GM}}$ *(Ours)*	48.5	**62.8**	54.7	42.0	**39.3**	40.7	57.1	81.3	**67.1**
14	Bilinear	AC	$\mathcal{L}^{S}_{\text{cWGAN}} + \mathcal{L}_{\text{GM}}$ *(Ours)*	53.8	58.2	55.9	43.2	36.2	39.4	54.8	74.1	63.0
15	Linear	AC	$\mathcal{L}^{S}_{\text{cWGAN}} + \mathcal{L}_{\text{GM}}$ *(Ours)*	45.8	65.5	53.9	**53.2**	33.0	**42.8**	46.8	84.8	60.3

In summary

- a **novel** proxy loss for **zero-shot learning**
  - better estimation of class distributions
- **state of the art** on CUB, AWA and SUN

Source code: https://mbsariyildiz.github.io/

Part II

Zero-shot Detection and Captioning of Unseen Objects

Why study zero-shot *detection*?

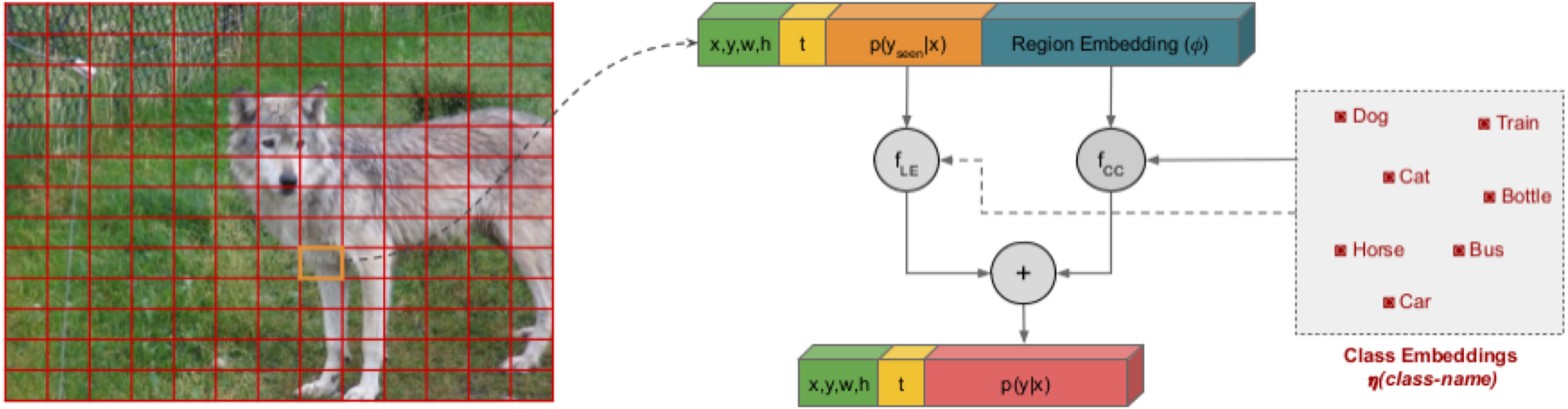
Detection in the Wild
using text-based queries

Robotic

Our approach

➔ Our method consists of two components:
  - (i) utilize a convex combination of class embeddings,
  - (ii) directly learn to map regions to the space of class embeddings.

➔ Zero-shot object detection within the YOLO detection framework.



Convex Combination of Class Embeddings

- Represent a given image region (i.e. a bounding box) as the convex combination of training class embeddings.

$$f_{\mathrm{CC}}(x,b,y) = \frac{\phi_{\mathrm{CC}}(x,b)^{\mathrm{T}} \underbrace{\eta(y)}_{\text{Class embedding}}}{\|\phi_{\mathrm{CC}}(x,b)\| \|\eta(y)\|}$$

$$\phi_{\mathrm{CC}}(x,b) = \frac{1}{\sum_{y \in \mathcal{Y}_s} p(y|x,b)} \sum_{y \in \mathcal{Y}_s} p(y|x,b)\eta(y)$$

Convex Combination of Class Embeddings

- Represent a given image region (i.e. a bounding box) as the convex combination of training class embeddings.

Sum of class embeddings, weighted by posterior probability

$$f_{\mathrm{CC}}(x,b,y) = \frac{\phi_{\mathrm{CC}}(x,b)^{\mathrm{T}}\eta(y)}{\|\phi_{\mathrm{CC}}(x,b)\|\|\eta(y)\|}$$

$$\boxed{\phi_{\mathrm{CC}}(x,b) = \frac{1}{\sum_{y\in\mathcal{Y}_s} p(y|x,b)} \sum_{y\in\mathcal{Y}_s} p(y|x,b)\eta(y)}$$

Region Scoring by Label Embedding

- The goal is to directly model the compatibility between the visual features of image regions and class embeddings.
- The equation can be interpreted as a dot product between L2-normalized image region descriptors and class embeddings.

Region descriptor

$$f_{\mathrm{LE}}(x,b,y) = \frac{\phi_{\mathrm{LE}}(x,b)^{\mathrm{T}}\eta(y)}{\|\phi_{\mathrm{LE}}(x,b)\|\,\|\eta(y)\|}$$

Hybrid region embedding

- The two scores are accumulated within the loss function:

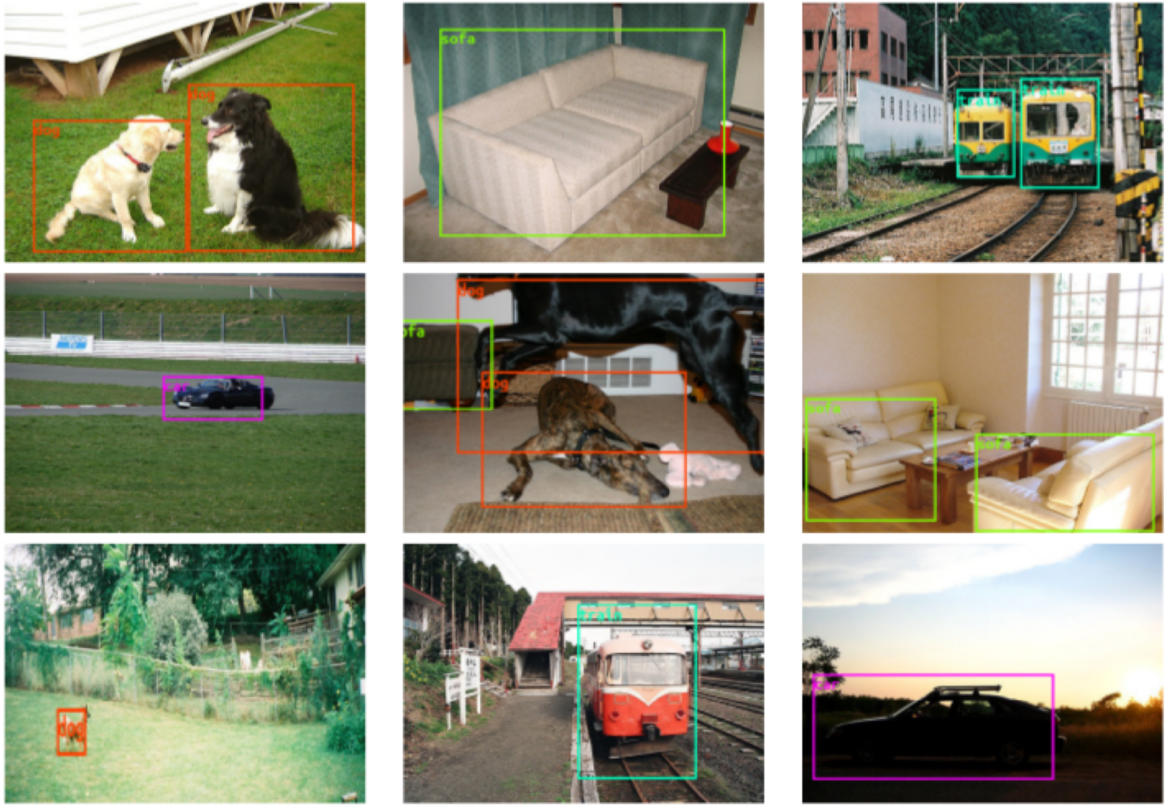
$$L_{\mathrm{LE}}(x,b,y) = \frac{1}{|\mathcal{Y}_s|-1} \sum_{y' \in \mathcal{Y}_s \setminus \{y\}} \max\left(0, 1 - f_{\mathrm{LE}}(x,b,y) + f_{\mathrm{LE}}(x,b,y')\right)$$

Experimental Results on PASCAL VOC

- Select 16 of the 20 classes as the training set.
- Remaining 4 classes as the test set. These test classes are car, dog, sofa and train respectively.
- Class-attribute relations of aPaY dataset are used for semantic descriptions.
- 65.6% mAP on seen classes, 54.6% mAP on unseen ones.

Method	Test split	aeroplane	bicycle	bird	boat	bottle	bus	cat	chair	cow	dining table	horse	motorbike	person	potted plant	sheep	tvmonitor	car	dog	sofa	train	mAP (%)
LE	v	.46	.50	.44	.28	.12	.59	.44	.20	.11	.38	.35	.47	.65	.16	.18	.53	-	-	-	-	36.8
	t	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	.54	.79	.45	.12	47.9
	v+t	.34	.48	.40	.23	.12	.34	.28	.12	.09	.32	.28	.36	.60	.15	.13	.50	.27	.26	.20	.05	27.4
CC	v	.69	.74	.72	.63	.43	.83	.73	.43	.43	.66	.78	.80	.75	.41	.62	.75	-	-	-	-	65.0
	t	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	.60	.85	.44	.27	53.8
	v+t	.67	.73	.70	.59	.41	.61	.58	.32	.32	.65	.74	.68	.72	.39	.57	.72	.49	.24	.10	.15	52.0
H	v	.70	.73	.76	.54	.42	.86	.64	.40	.54	.75	.80	.80	.75	.34	.69	.79	-	-	-	-	**65.6**
	t	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	.55	.82	.55	.26	**54.2**
	v+t	.68	.72	.74	.48	.41	.61	.48	.25	.48	.73	.75	.71	.73	.33	.59	.57	.44	.25	.18	.15	**52.3**

Example detections



Captioning with Unseen Objects

- **Motivation:** Overcome the data collection bottleneck in image captioning.
- **Task:** Define a new paradigm for generating captions of unseen classes.
- **Key Idea:** Use zero-shot object detector with template based sentence generator.

Zero-shot Image Captioning

	Image Captioning
Visual Input	
Textual Input	"a **person** riding a **horse**"

Zero-shot Image Captioning

{**person**, **horse**} ∈ **unseen classes**

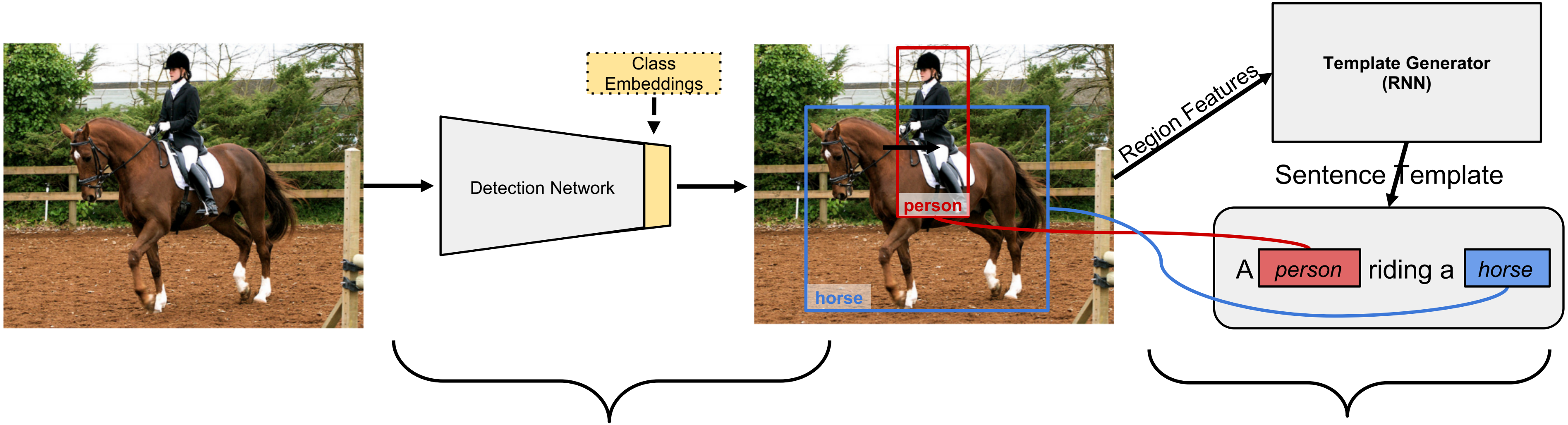
	Image Captioning	(Partial) Zero-Shot Image Captioning
Visual Input		
Textual Input	“a **person** riding a **horse**”	“a ~~**person** riding a **horse**~~”

Zero-shot Image Captioning

{person, horse} ∈ unseen classes

	Image Captioning	(Partial) Zero-Shot Image Captioning	True Zero-Shot Image Captioning
Visual Input			
Textua l Input	"a **person** riding a **horse**"	~~"a **person** riding a **horse**"~~	~~"a **person** riding a **horse**"~~

Framework - Fully Zero-shot Image Captioning



Generalized Zero-Shot Object Detector

Image Caption Generator*

* Lu, Jiasen, et al. "***Neural baby talk***." CVPR 2018.

Improving ZSD: Generalized Zero-shot Detection

- Unlike the prior work on ZSD, test captioning images contain a mixture of seen and unseen classes.

- Typically there is a significant bias towards the seen classes.

- Aim to overcome this problem by learning a scaling coefficient:

$$f(x,c,i) = \begin{cases} \alpha f(x,c,i), & if\ c \in \hat{Y}_s \\ f(x,c,i), & otherwise \end{cases}$$

Improving (G)ZSD - Better embeddings

- Reminder: detection scoring function $f(x, c, i)$ is defined as follows:

$$f(x, c, i) = \frac{\Omega(x, i)^T \psi(c)}{||\Omega(x, i)||||\psi(c)||}$$

- Here, $\psi(c)$ represents the class embedding for class c, which is now obtained in terms of **target-class to training-class similarities** in the word embedding space:

$$\psi(c) = [\varphi(c)^T \varphi(\bar{c}) \ + 1]_{\bar{c}}$$

- We also drop the convex combination approach to be able to deal with GZSD better.

Experimental Setup

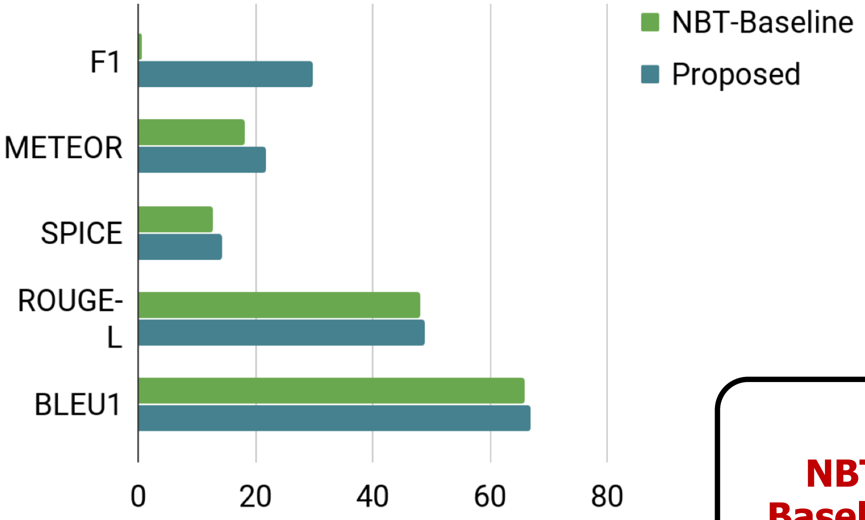
- **Dataset**: **MS-COCO** splits for evaluating zero-shot image captioning.
- **Evaluation**: F1 score, METEOR, SPICE, ROUGE-L, BLEU metrics.
- **Class embeddings:** Use 300-dim word2vec of class embeddings.
- **Evaluation - GZSD:** Use COCO val5k split, which contains both seen and unseen class instances.

Generalized-ZSD results

Classes	GZSD w/o α	GZSD
Bottle	0	0.8
Bus	0	21.4
Couch	2.7	4.9
Microware	0	1.2
Pizza	0	4.8
Racket	0	0.7
Suitcase	0	9.1
Zebra	0	15.8
U-mAP(%)	0.3	7.3
S-mAP(%)	27.4	19.2
Harmonic Mean	0.7	10.6

Typically, an unseen class instance is detected as the instance of some seen class

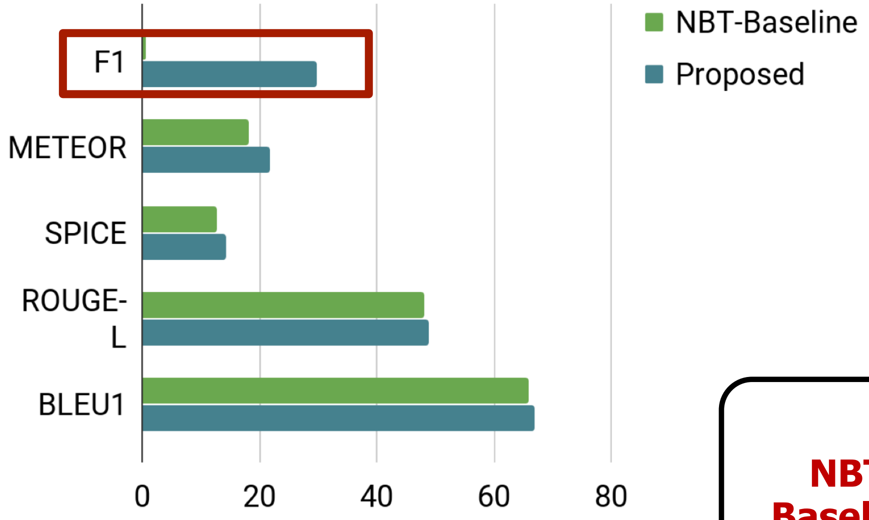
Image Captioning Results



Comparison Results

NBT-Baseline	A piece of **cake** on a white plate.	A yellow and black **train** traveling down the road.
Proposed	A piece of **pizza** on a white plate.	A yellow and black **bus** driving down a road.

Image Captioning Results

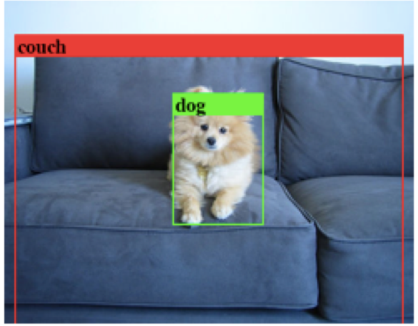


Comparison Results

NBT-Baseline	A piece of **cake** on a white plate.	A yellow and black **train** traveling down the road.
Proposed	A piece of **pizza** on a white plate.	A yellow and black **bus** driving down a road.

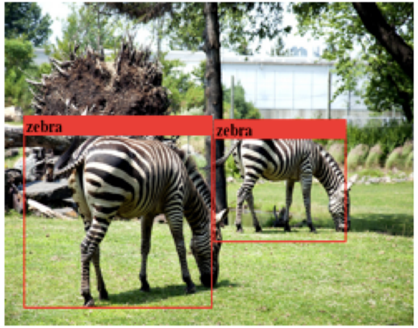
Qualitative Results

Image captioning results of images which consist of **seen** and **unseen** classes:

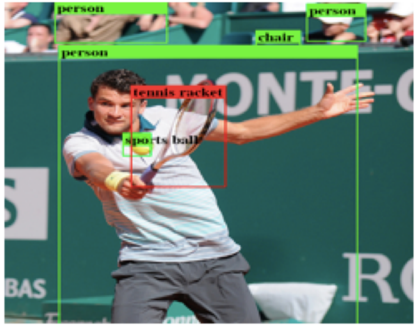


A small white **dog** is sitting on a **couch**.

A red **bus** is driving down the street.



A couple of **zebra** standing in a field.



A **tennis player** is about to hit a **racket**.

A white plate topped with a piece of **pizza**.

A kitchen with a **microwave** and a counter.

In summary,

- a **new** problem: generating captions of images with **unseen classes**.
- a **novel** approach for generalized zero-shot object detection problem.

Part III

Do we really need ZSL in practice?

Do we really need ZSL in practice?

- ZSL sometimes sounds like a pratically irrelevant problem given that there are several large-scale datasets, such as Google Open Images, ImageNet, etc.

However:

- 1) These datasets are arguably still very far from capturing **richness** of human vision
- 2) Large-scale data collection can be **inherently difficult** due to physical constraints, lack of annotation experts, etc. in certain problems.

Do we really need ZSL in practice?

- I will talk about two examples:
  - Fine-grained recognition in remote sensing
  - Sign Language Recognition

Traditional Object Recognition in Remote Sensing

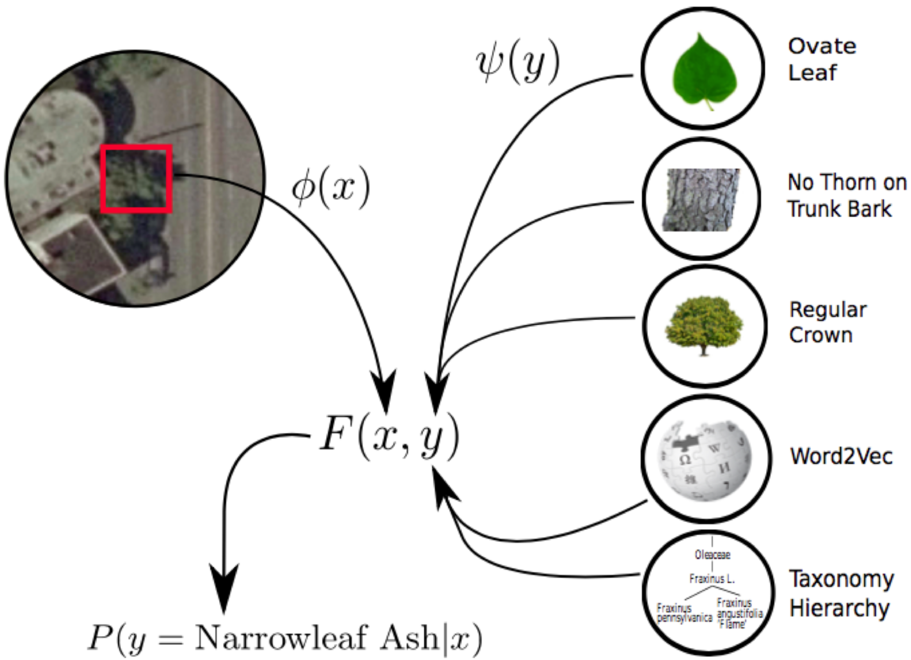
- The mainstream object recognition task in remote sensing:
  - Benchmark datasets: UC Merced, AID, etc.
  - Assign each pixel/patch to one of few categories
  - *eg. Agricultural vs Beach vs Forest vs Freeway vs Harbor*
  - Typically there are a large number of examples per class
- Distinct classes
- Largely an over-simplified categorization of earth surface

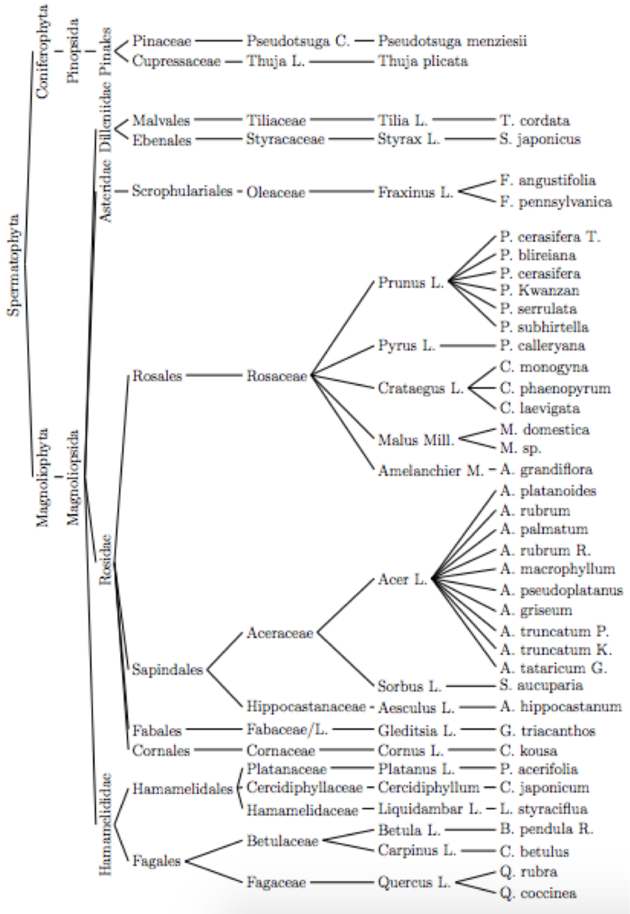
gisgeography.com/image-classification-techniques-remote-sensing/

Fine-grained Recognition in Remote Sensing

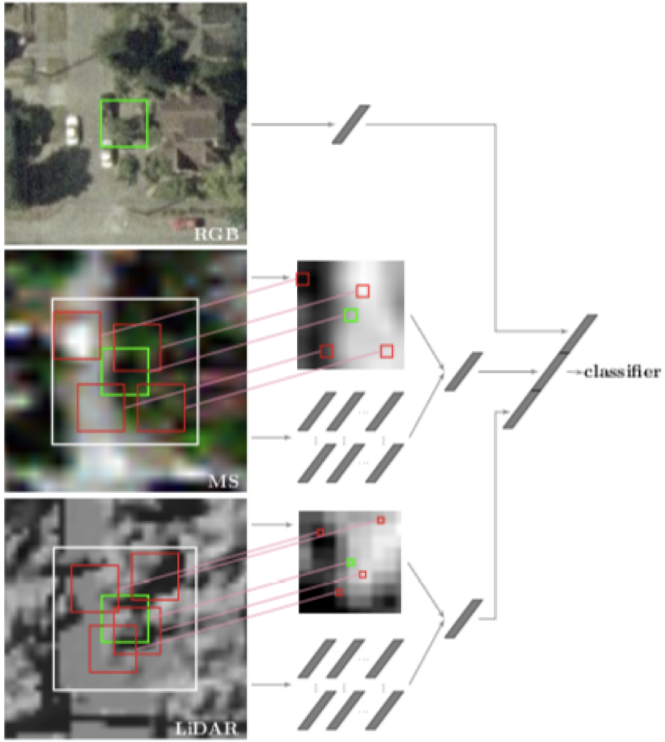
- Under-studied problem: fine-grained, semantically rich recognition
- Our focus: 40 different tree species and their satellite views
- We manually cleaned 48k GPS-tagged samples belonging to 40 top categories

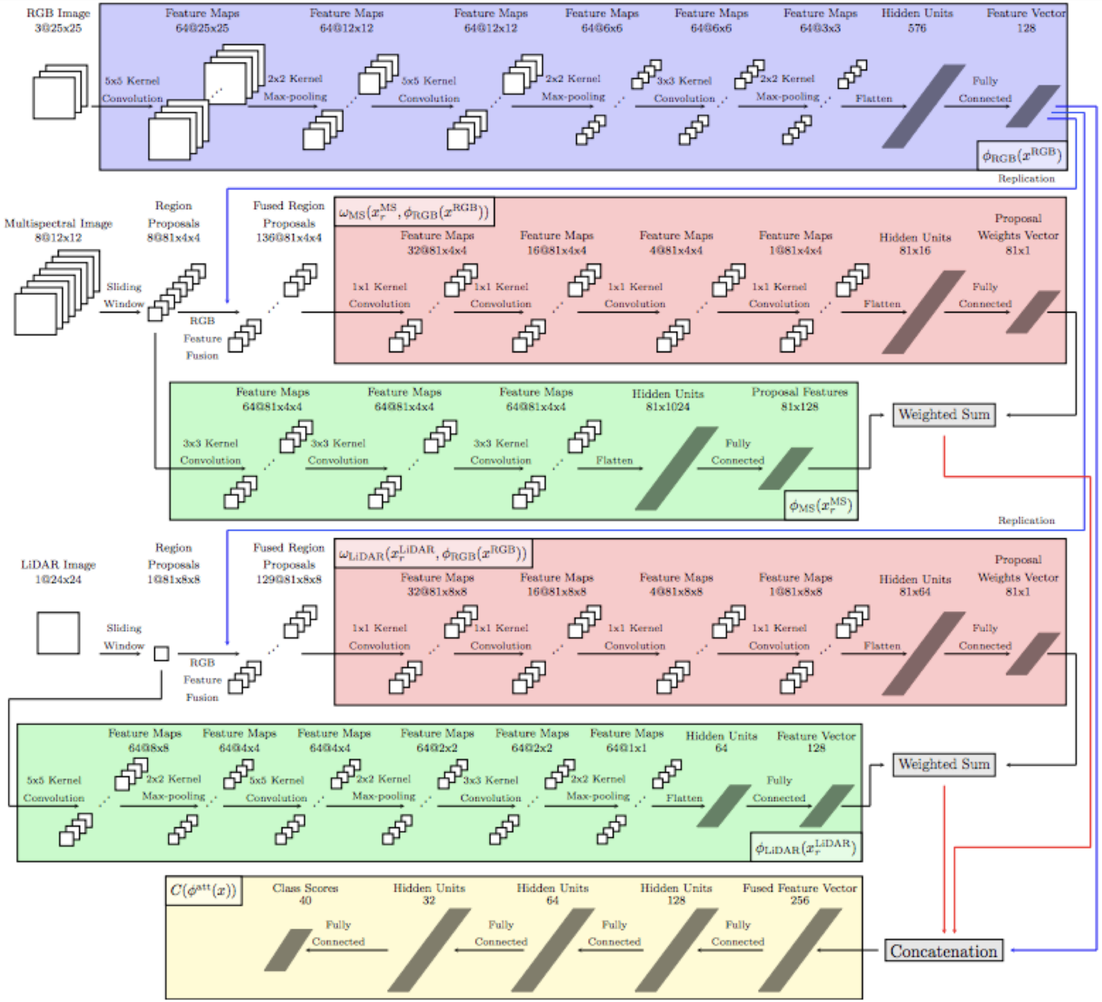
Formulation





Multisource Region Attention Network





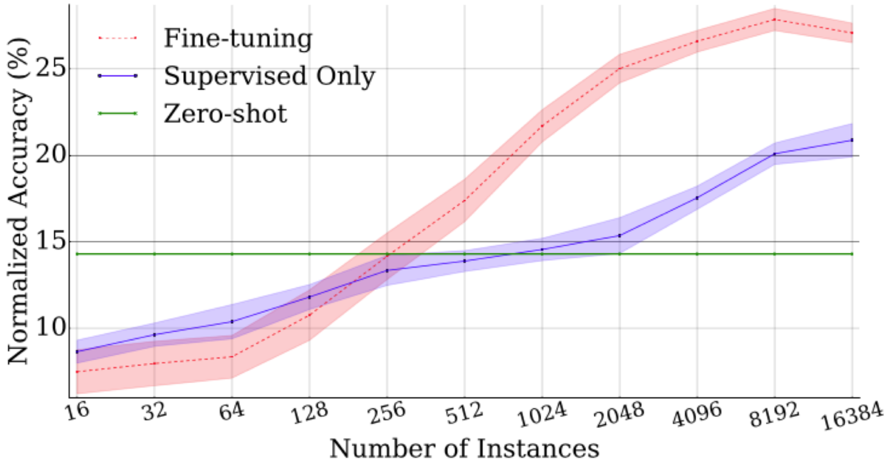
The quantitative annotation advantage of ZSL

SUPERVISED CLASSIFICATION RESULTS (IN %)

	Random guess	Logistic regression	Random forest	CNN	CNN with perturbation
Normalized accuracy	5.6	16.4	15.7	27.9	34.6

ZERO-SHOT LEARNING RESULTS (IN %)

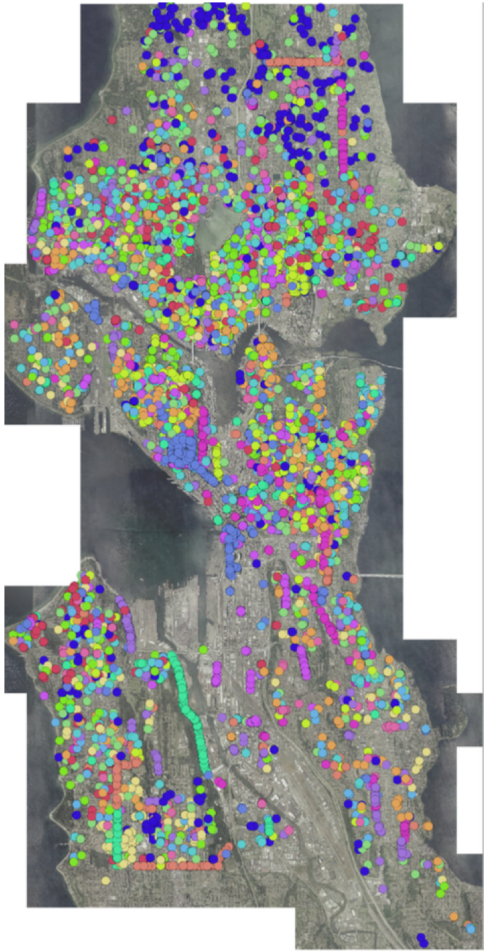
	Random guess	ALE [28]	SJE [14]	ESZSL [9]	Ours
Normalized accuracy	6.3	12.5	12.6	13.2	14.3



- ZSL is advantageous up to 256 supervised samples
- Note that (i) ZSL uses no examples, (ii) most categories are hard to distinguish even by visual inspection

Data collection problem, revisited

- Collection of even 256 samples can be very costly
- Just not possible to annotation by looking into the images. It is necessary to physically visit the instances & GPS-tag them.
- 16-test classes, around Seattle (WA), scattered around **217 km²**
- Arguably not feasible for scaling up for monitoring tree species all over the world.



Sign Language Recognition

Most of the SLR approaches require a large amount of annotated data to recognize predefined sign classes.

Problem 1: Excessive manual annotation

Problem 2: What if we want to recognize other signs?

60

Learning Signs with Minimal Supervision

- Languages are constantly growing
  - Thousands of new words are added to OED every year.
- Same is true for Sign Languages

OED Oxford English Dictionary
The definitive record of the English language

Home > Updates to the OED > New words list March 2019

New words list March 2019

New word entries

— anti-suffragism, n.: "Opposition to the extension of the right to vote in political elections to women; the political movement dedicated to this."

— Aperol, n.: "A proprietary name for: an orange-coloured Italian aperitif flavoured with gentian, rhubarb, and a variety of herbs and roots. Also: a drink of this."

— archicembalo, n.: "Any of various types of harpsichord having more than twelve keys to the octave and therefore capable of producing intervals smaller than a semitone…"

— Argonautical, adj.: "Of, relating to, or likened to the Argonauts. Cf. Argonautic adj."

— Assiniboine, n. and adj.: "A member of a Siouan people of the Great Plains, now living mainly in southern Saskatchewan and northern Montana."

— Aucklander, n.: "A native or inhabitant of city or region of Auckland, New Zealand."

— baff, n.2: "A slipper; = baffie n. Usually in plural."

— baffie, n.: "A slipper, esp. one that is old and worn out (cf. bauchle n. 1). Usually in plural. Cf. baff n.2"

— baggataway, n.: "The game of lacrosse, esp. as played by certain indigenous peoples of eastern Canada and the midwestern and northeastern United States, using sticks…"

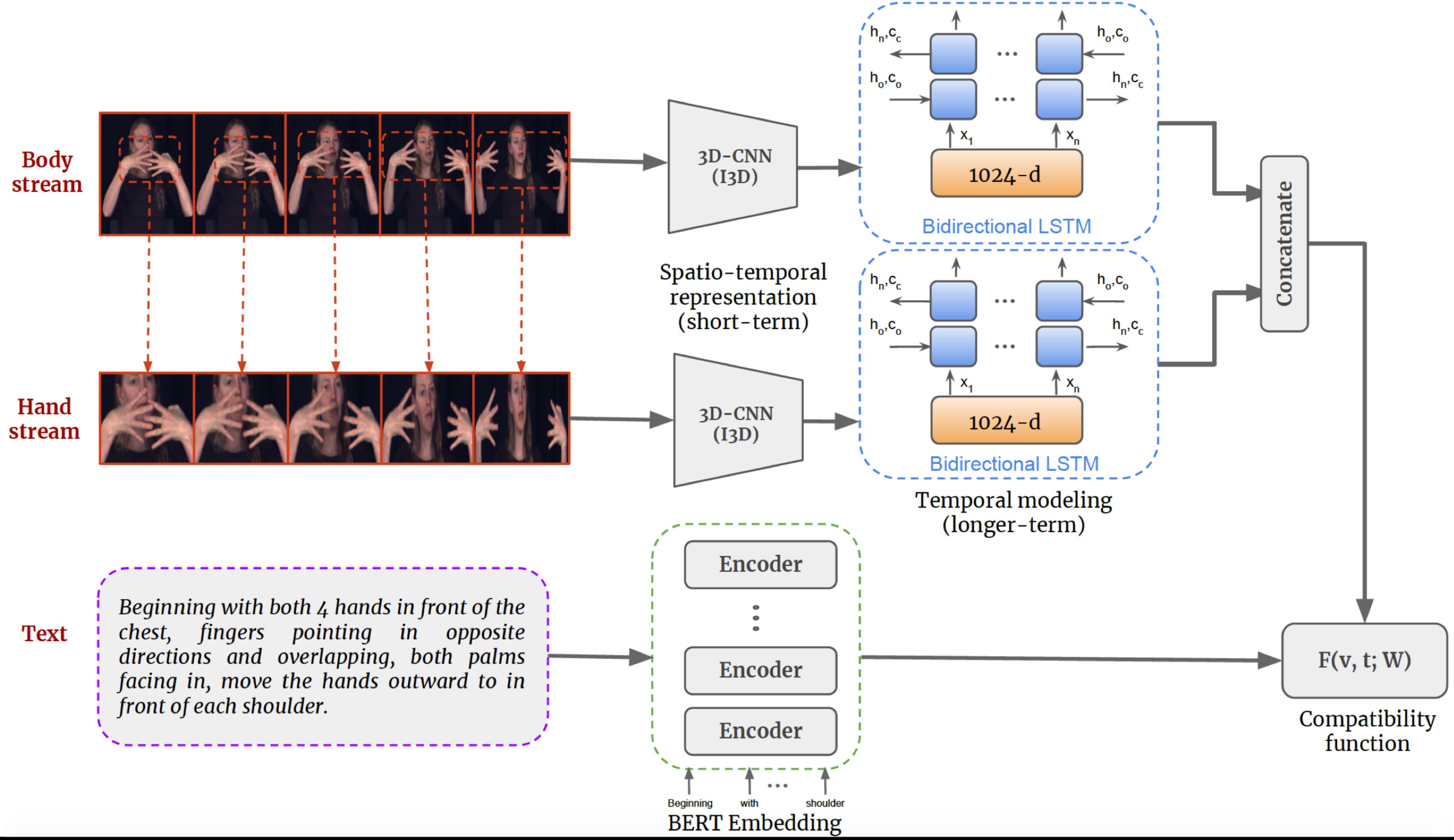
…

Zero-shot Sign Language Recognition (ZSSLR)

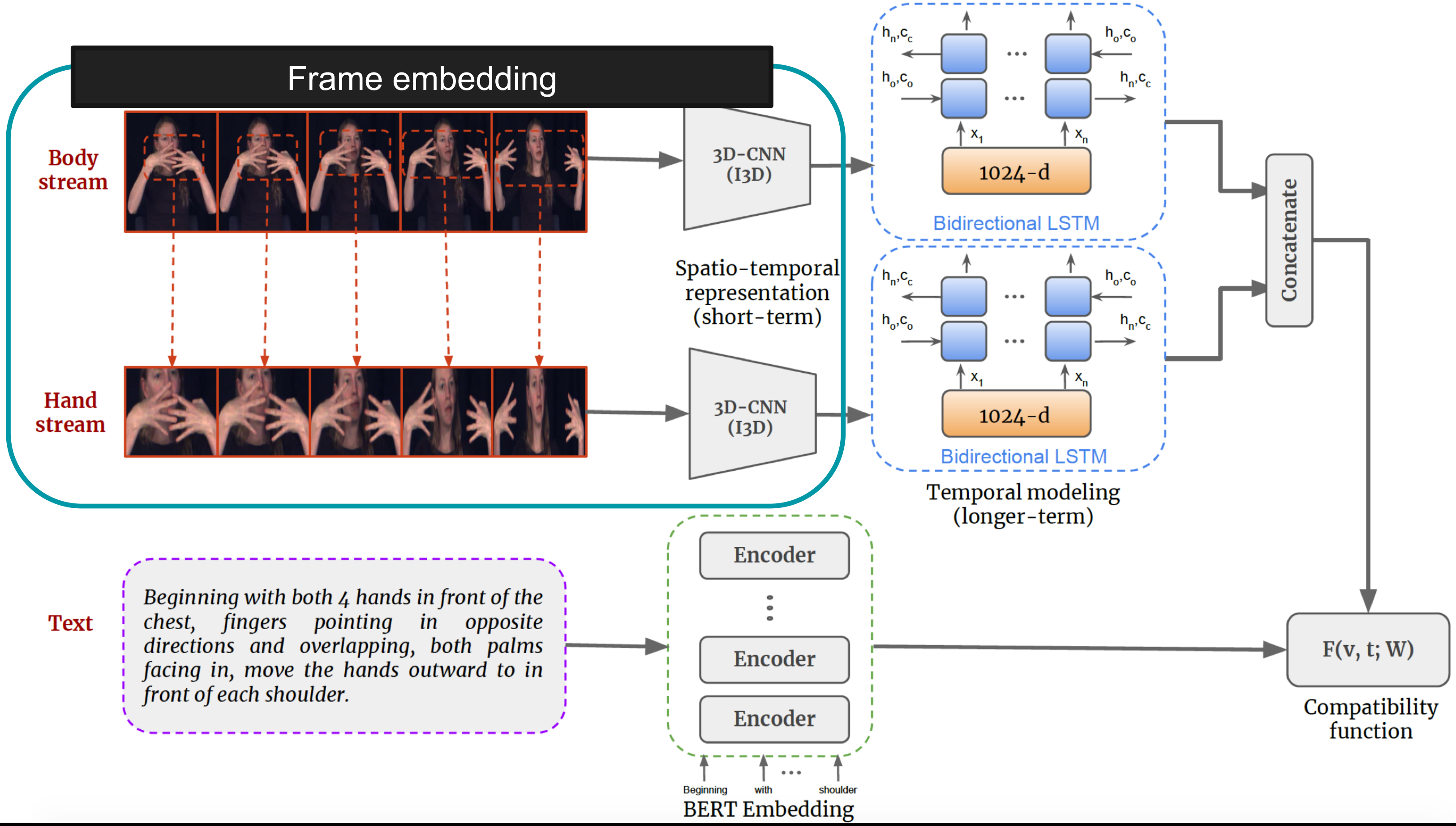
BICYCLE: Move both S hands in alternating forward circles, palms facing down, in front of each side of the body.

HIGH: Move the right H hand, palm facing left and fingers pointing forward, from in front of the right side of the chest upward to near the right side of the head.

Our ZSSLR model

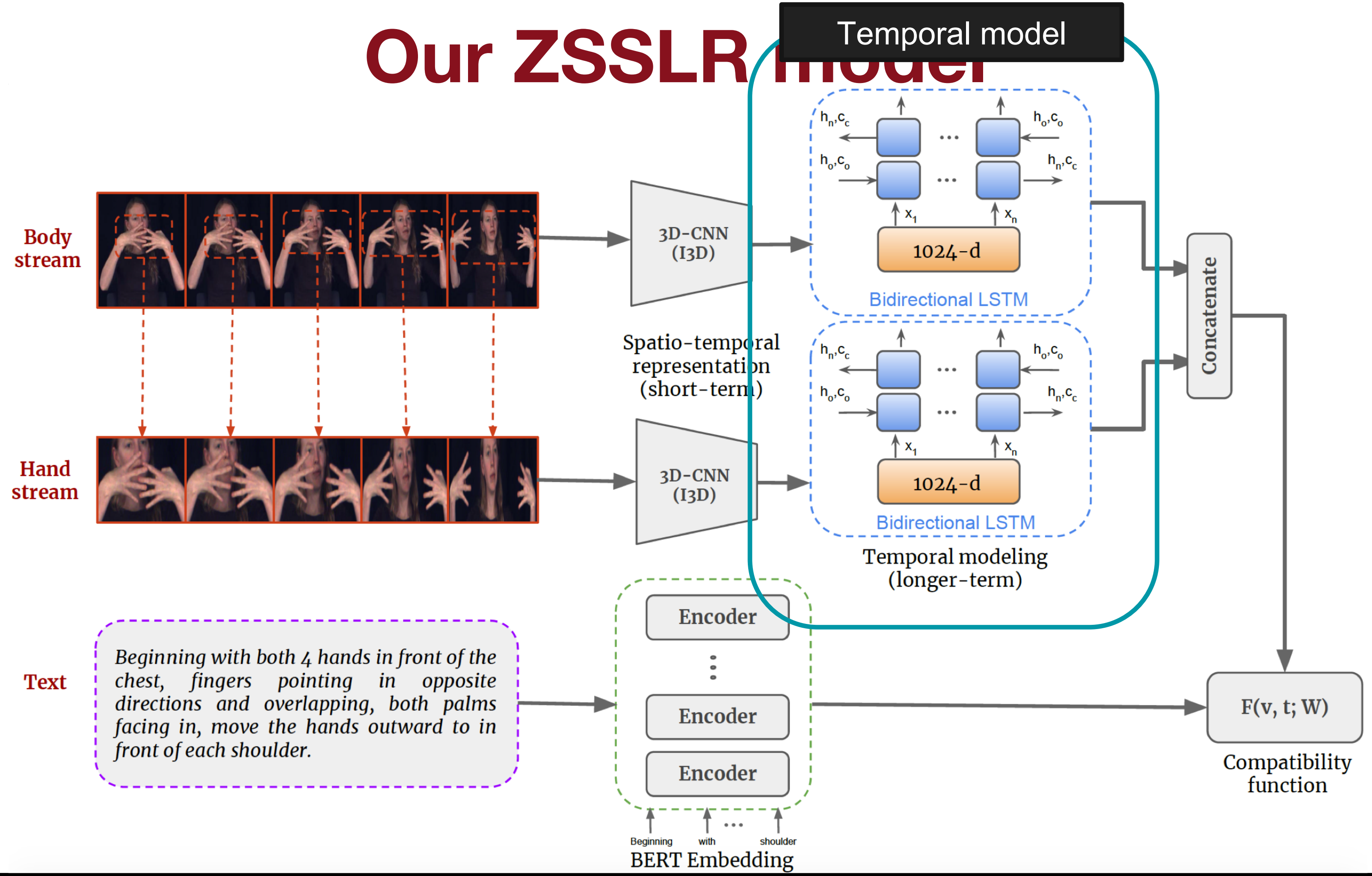


Our ZSSLR model

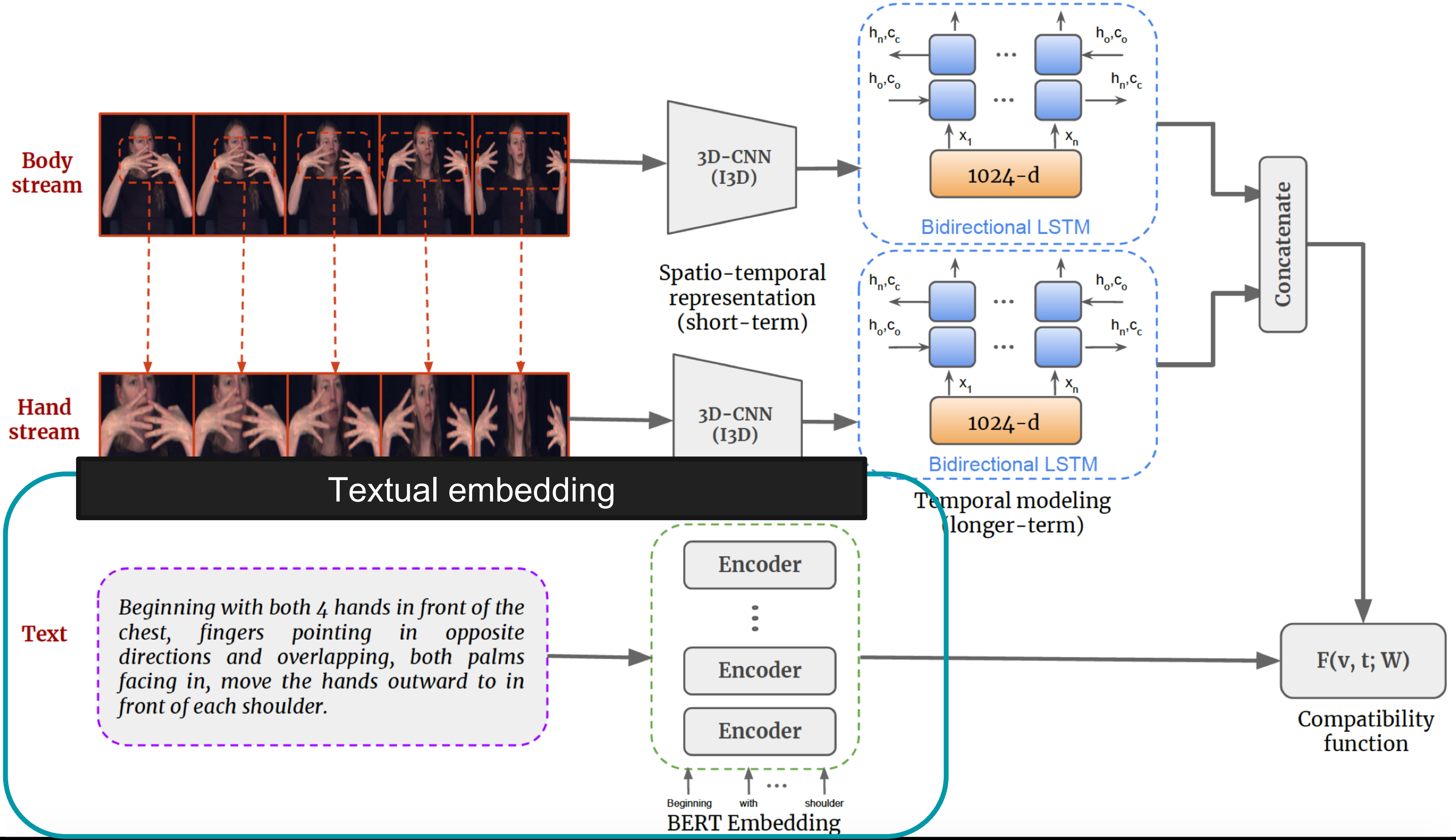


64

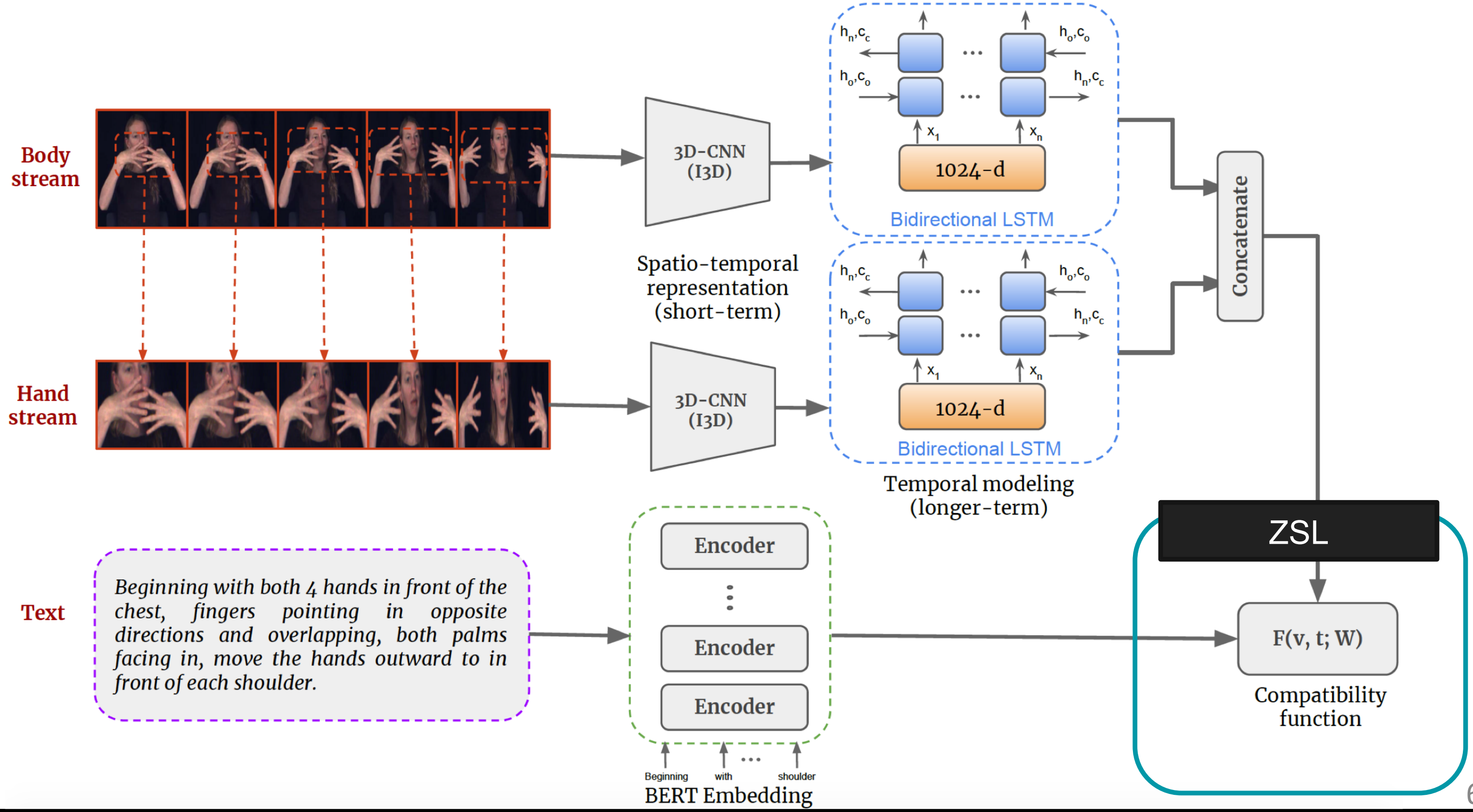
Our ZSSLR model



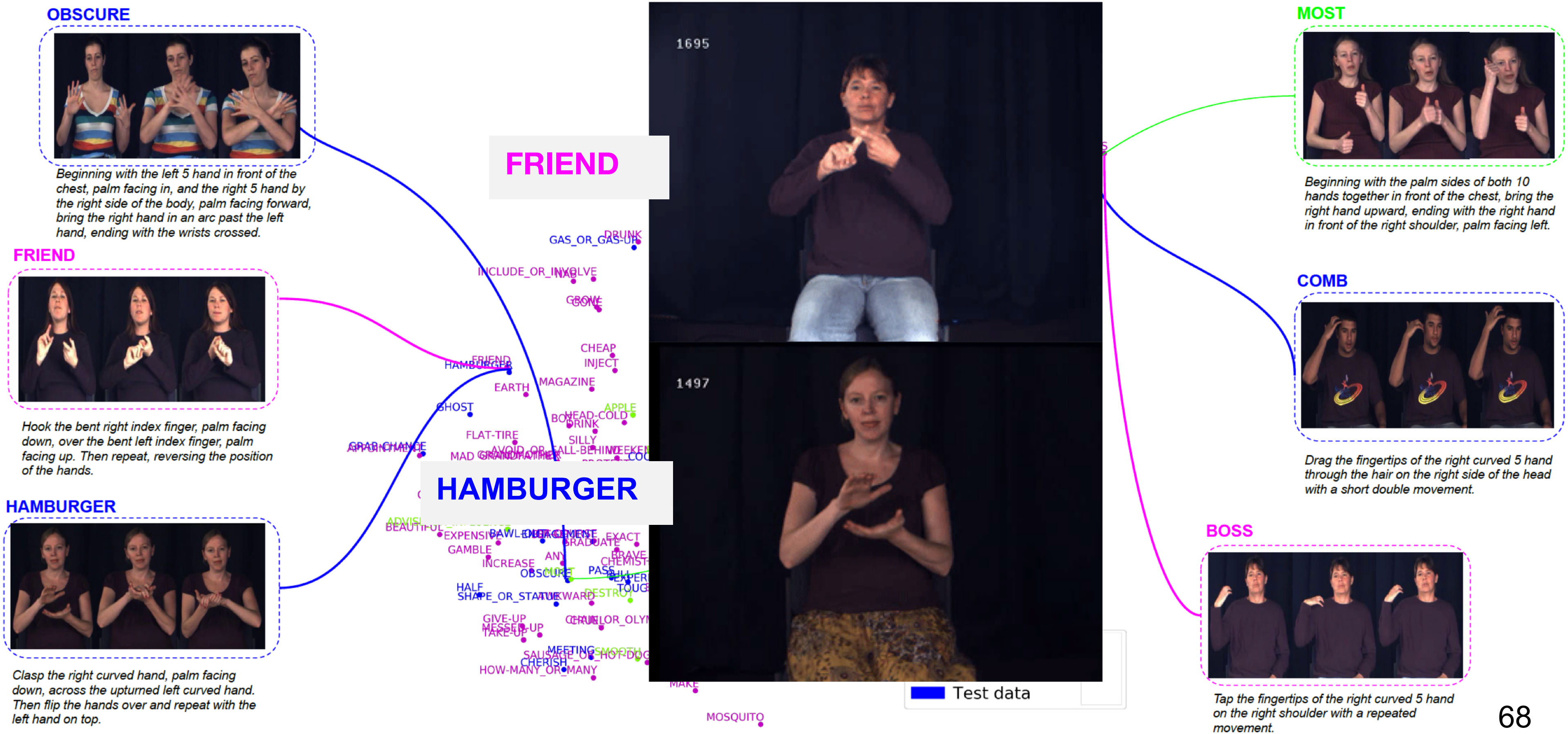
Our ZSSLR model



Our ZSSLR model



Text description embeddings



68

ZSSLR Experimental Results

Temporal Representation	top-1	top-2	top-5
AvePool	18.0	27.4	43.8
LSTM	18.2	28	47.2
GRU	19.7	31.8	50.0
bi-LSTM	**20.9**	**32.5**	**51.4**

Accuracies are still quite low, large room for improvement

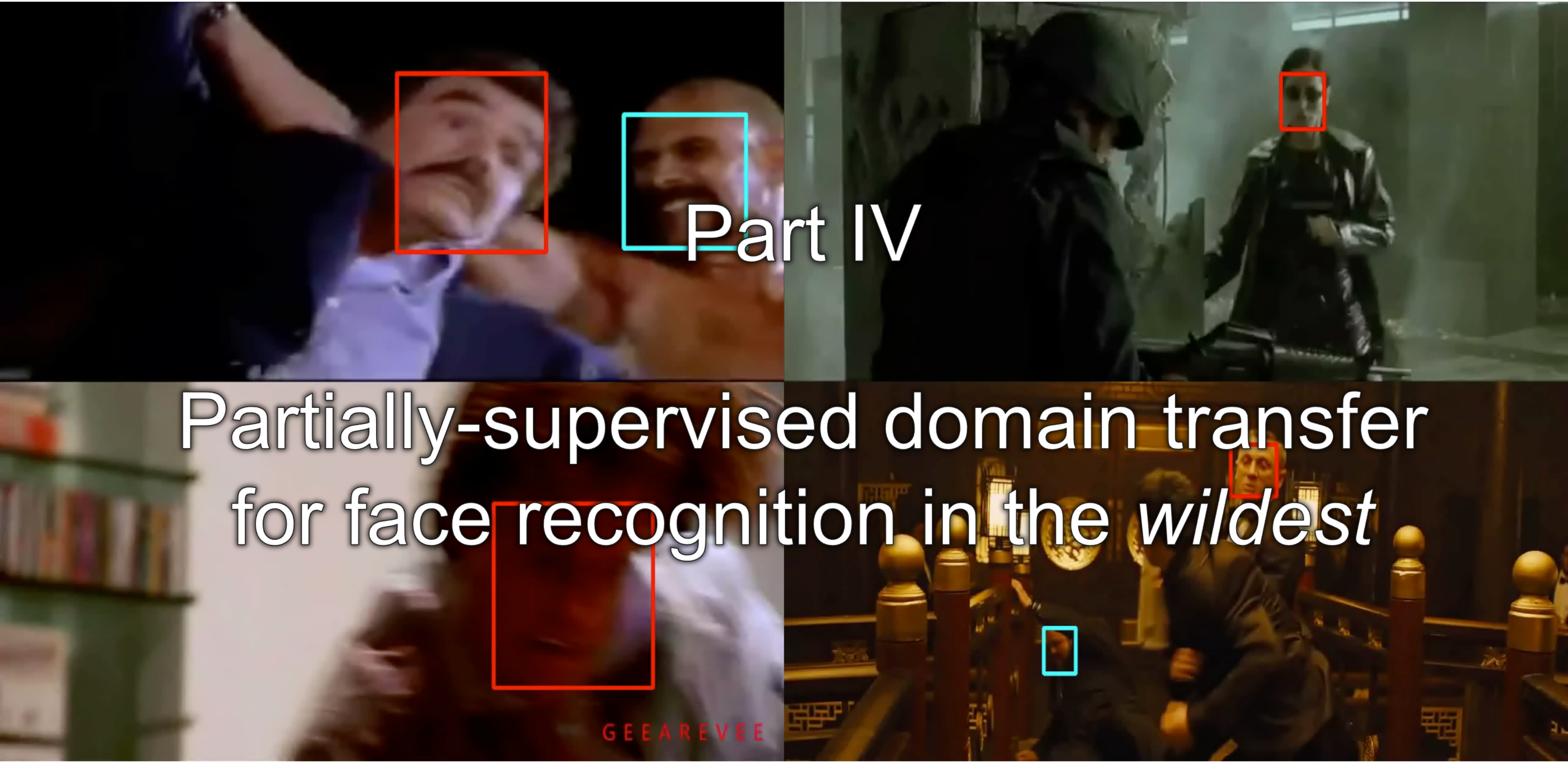
Dataset available for download : https://ycbilge.github.io/zsslr.html

Conclusions

- Presented two problems where the need for zero-shot learning naturally emerges:
  1. Fine-grained recognition in remote sensing – towards globally monitoring all tree species
  2. Sign Language Recognition – towards recon – towards recognizing all words in all sign languages, with quick adaptation to novel words

Part IV

Partially-supervised domain transfer for face recognition in the *wildest*



Face Recognition in the “Wildest”

- Face Recognition is largely solved in controlled cases (> 95% accuracy).
- People in criminal activity expose a diverse set of facial expressions
- These people may not necessarily have prior criminal records
    - Only have passport or Facebook type photos

Al Pacino

Brad Pitt

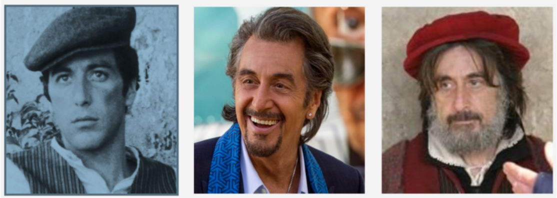
Hillary Swank

73

WildestFaces Dataset

Collected video scenes from YouTube

- Car chase, fist fights, gun fights, heated arguments
- 64 actors
- 2186 shots - 64,242 frames



Clean images from IMDB-WIKI & Internet

- 64 actors
- 8069 images

Some may not have prior violent footage –> Partially supervised

GZSL-like setting

- **Train:** clean images of 64 classes, videos of 40 seen classes
- **Val:** videos of 40 seen classes, 10 unseen classes
- **Test:** videos of 64 classes (seen + unseen)

Partially Supervised Domain Transfer

source domain classifier

$$\min_{W} R(W) + \sum_{i=1}^{n_x} \ell(\boxed{\phi(x_i)}^{\top} W, y_i)$$

Image representation

target domain classifier

$$\min_{\tau} R(\tau) + \sum_{j=1}^{n_v} \ell(\boxed{\tau(\Psi(v_j)}))^{\top} W, y_j)$$

Classifier transfer + temporal adaptation

Classifier transfer layer

$$\min_{\tau} R(\tau) + \sum_{j=1}^{n_v} \ell(\boxed{\tau(\Psi(v_j))}^\top W, y_j)$$

$$\boxed{\tau_{\text{fc}}(\Psi(v)) = Q\Psi(v)}$$ **Fully-connected classifier transfer**

$$\boxed{\tau_{\text{affine}}(\Psi(v)) = \alpha \odot \Psi(v) + \beta}$$ **Affine classifier transfer**

$$\boxed{\tau_{\text{rsa}}(\Psi(v)) = \alpha_2 \odot \max(\alpha_1 \odot \Psi(v) + \beta_1, 0) + \beta_2 + \Psi(v)}$$ **Residual stacked affine**

Temporal adaptation

$$\min_{\tau} R(\tau) + \sum_{j=1}^{n_v} \ell(\tau(\boxed{\Psi(v_j)})^\intercal W, y_j)$$

Temporal average pooling

$$\Psi_{\text{AvgPool}}(v) = \frac{1}{|v|} \sum_{t=1}^{|v|} \phi(v[t])$$

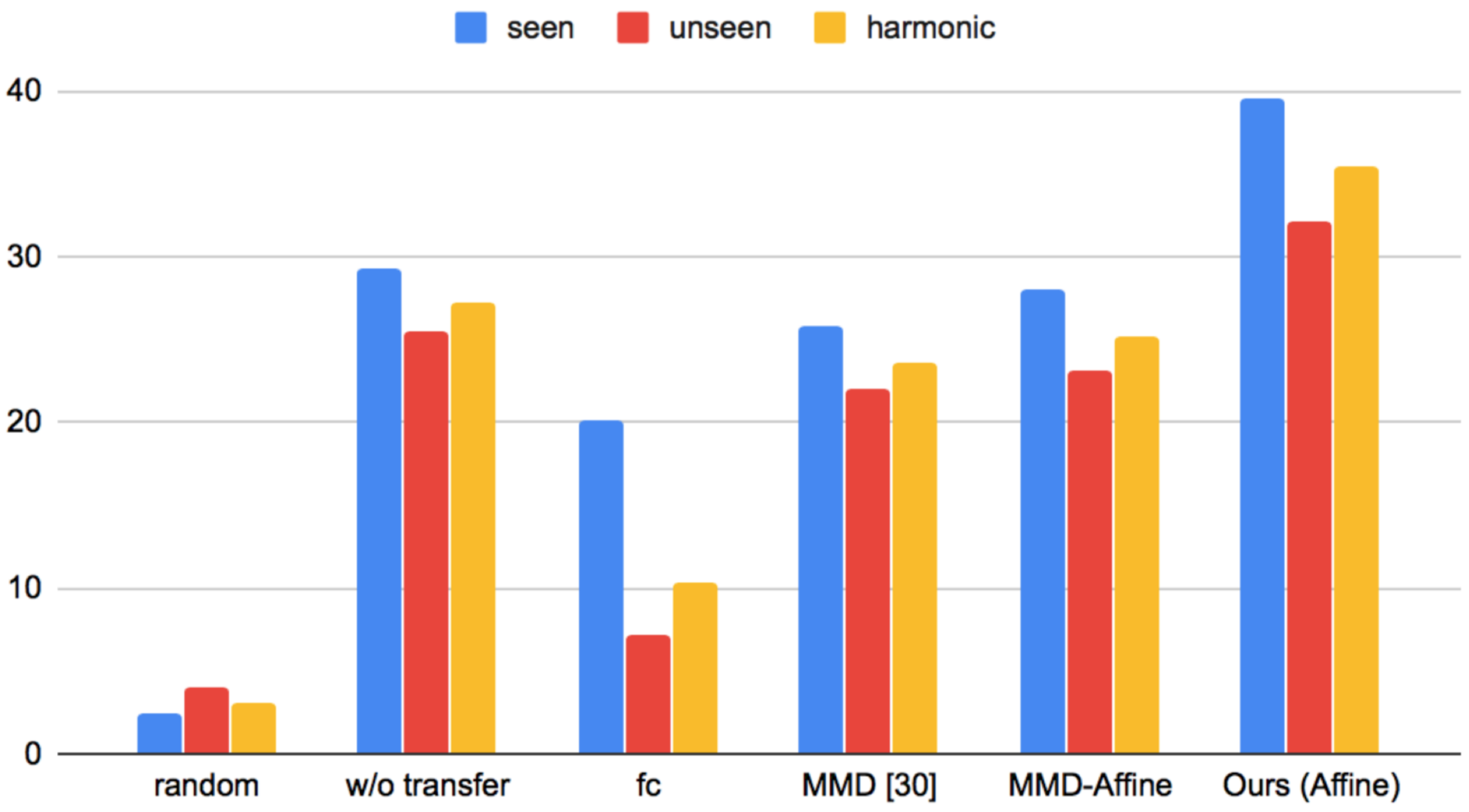
Attention matrix

Attentive temporal pooling

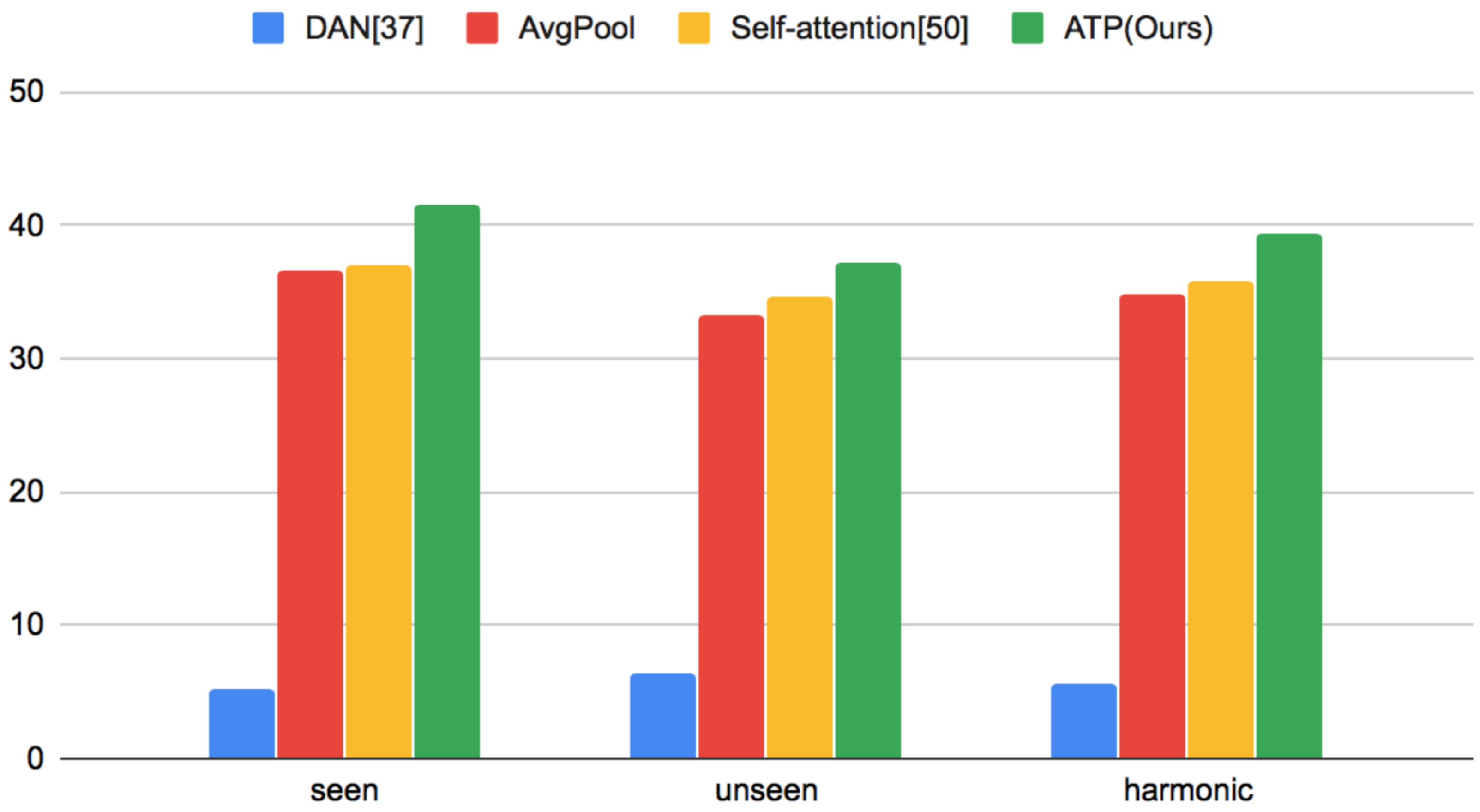
$$\Psi_{\text{ATP}}(v) = \frac{1}{K} \Phi(v) \boxed{\Gamma(v)} \mathbf{1}_K$$

$$[\Gamma(v)]_{t,k} = \frac{\exp\left[\phi(v[t])^\intercal a_k\right]}{\sum_{t'=1}^{|v|} \exp\left[\phi(v[t'])^\intercal a_k\right]}$$

Classifier Transfer



Comparison to literature



Summary

- Towards semantically rich recognition systems, build models that are
  - more flexible
  - more tightly integrated with language
  - requires less supervision
- Presented:
  - Gradient Matching Networks (currently for ZSL)
  - A zero-shot object detection approach, with application to image captioning
  - Two real-world applications of ZSL
  - A partially supervised model domain transfer problem

Thank you!