Neural Algebra of Classifiers

A deep learning framework for composition of classifiers

Rodrigo Santa Cruz[1], Basura Fernando[1], Anoop Cherian[1,2], and Stephen Gould[1]
[1]Australian Centre for Robotic Vision, Australian National University, Canberra, Australia
[2]Mitsubishi Electric Research Labs, 201 Broadway, Cambridge, MA

Motivation

Motivation

Which one is an **albatros**?

Motivation

Which one is an **albatros**?

Albatrosses are birds with hooked beak and large wingspan.

Motivation

Which one is an **albatros**?

Albatrosses are birds with hooked beak and large wingspan.

Albatross

Motivation

Which one is an **albatros**?

Albatrosses are birds with hooked beak and large wingspan.

Which one is a **frigatebird**?

Albatross

Motivation

Which one is an **albatros**?

Albatrosses are birds with hooked beak and large wingspan.

Which one is a **frigatebird**?

Frigatebirds seem black albatrosses with white or red pouch.

Albatross

Motivation

Which one is an **albatros**?

Albatrosses are birds with hooked beak and large wingspan.

Which one is a **frigatebird**?

Frigatebirds seem black albatrosses with white or red pouch.

Albatross

Frigatebird

Motivation

Which one is an **albatros**?

Albatrosses are birds with hooked beak and large wingspan.

Which one is a **frigatebird**?

Frigatebirds seem black albatrosses with white or red pouch.

Albatross

Frigatebird

The **human recognition system** is fundamentally **compositional**, so **unseen visual complex concepts** are recognized from the composition of **simple visual primitives** according to **well-defined rules**.

Motivation

Which one is an **albatros**? (hooked beak AND large wingspan)

Albatrosses are birds with hooked beak and large wingspan.

Which one is a **frigatebird**?

Frigatebirds seem black albatrosses with white or red pouch.

Albatross

Frigatebird

The **human recognition system** is fundamentally **compositional**, so **unseen visual complex concepts** are recognized from the composition of **simple visual primitives** according to **well-defined rules**.

Motivation

Which one is an **albatros**? (hooked beak AND large wingspan)

Albatrosses are birds with hooked beak and large wingspan.

Which one is a **frigatebird**? (albatross AND (white pouch OR red pouch)

Frigatebirds seem black albatrosses with white or red pouch.

Albatross

Frigatebird

The **human recognition system** is fundamentally **compositional**, so **unseen visual complex concepts** are recognized from the composition of **simple visual primitives** according to **well-defined rules**.

Motivation

Which one is an **albatros**? (hooked beak AND large wingspan)

Albatrosses are birds with hooked beak and large wingspan.

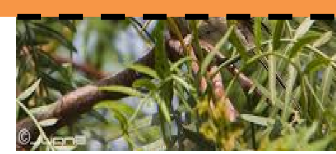
Which one is a **frigatebird**? (albatross AND (white pouch OR red pouch)

Frigatebirds seem black albatrosses with white or red pouch.

Can we learn how to compose classifiers for unseen complex concepts from simple visual primitives?

Can we develop an algebra for composition of primitives?

Albatross

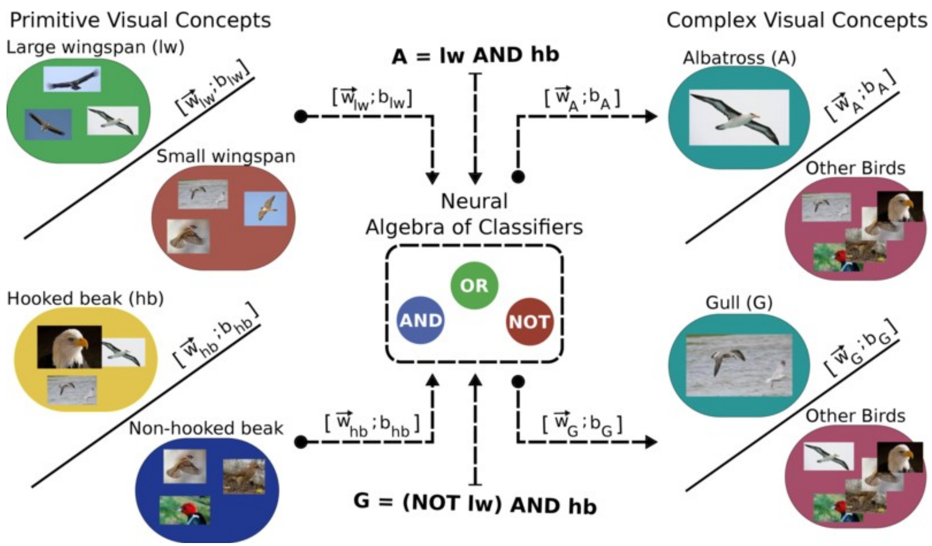


Frigatebird

The **human recognition system** is fundamentally **compositional**, so **unseen visual complex concepts** are recognized from the composition of **simple visual primitives** according to **well-defined rules**.

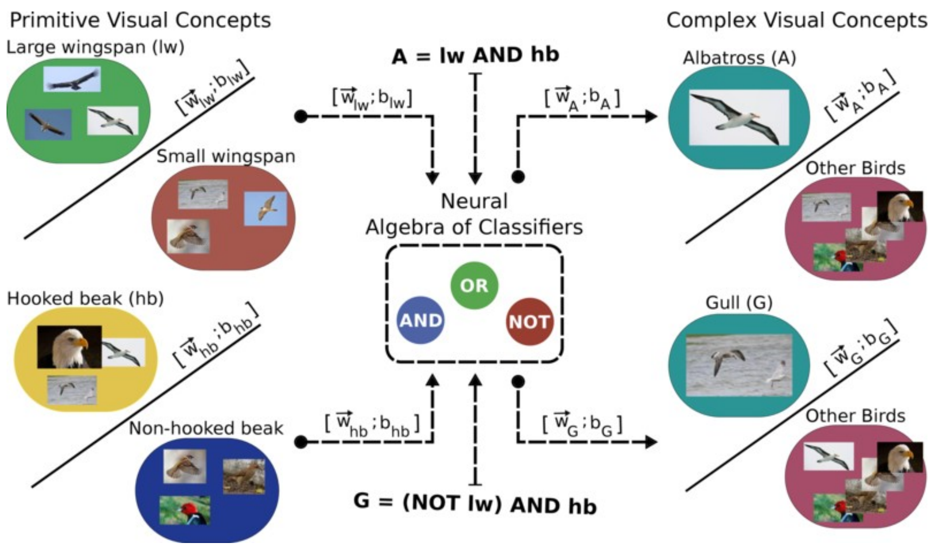
Problem Description

How to synthesize classifiers for arbitrary compositions of visual primitives ?



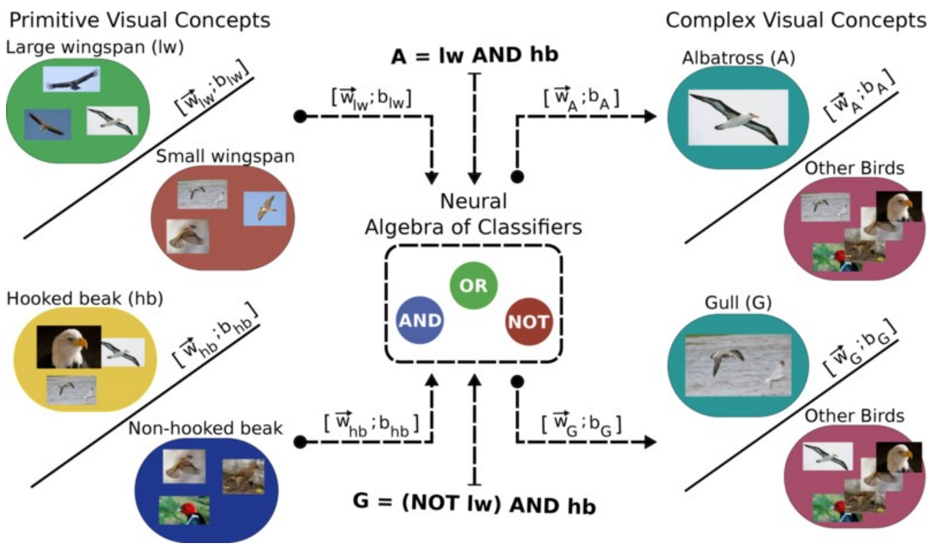
Problem Description

How to synthesize classifiers for arbitrary compositions of visual primitives ?



Problem Description

How to synthesize classifiers for arbitrary compositions of visual primitives ?

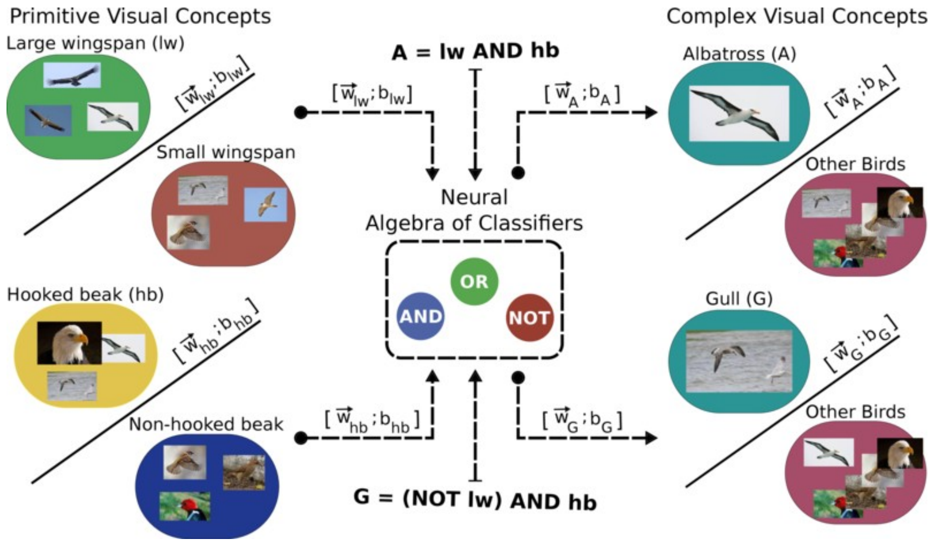


Problem Description

How to synthesize classifiers for arbitrary compositions of visual primitives ?

Large wingspan (lw)

Hooked beak (hb)



- Visual primitives (p): known simple visual concepts.

AUSTRALIAN CENTRE FOR ROBOTIC VISION

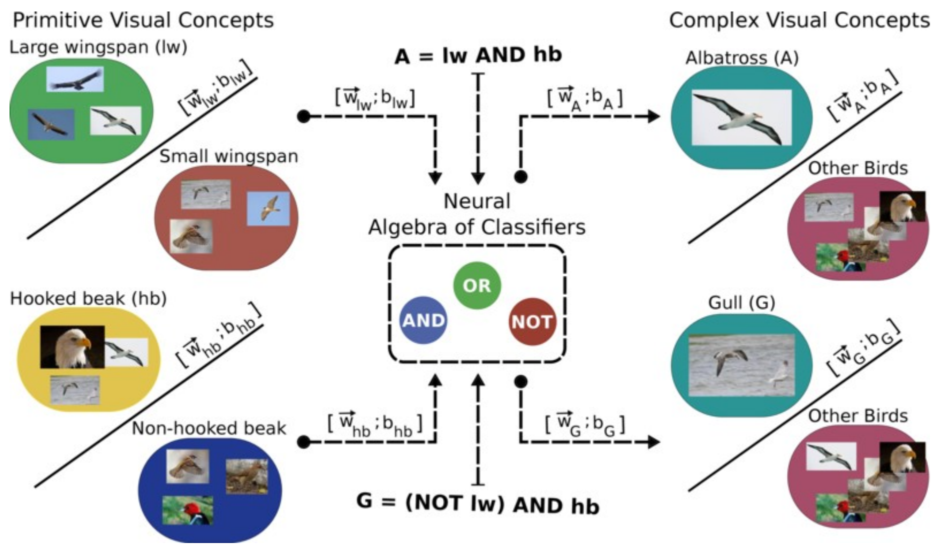
Australian National University

Problem Description

How to synthesize classifiers for arbitrary compositions of visual primitives ?

Large wingspan (lw)

Hooked beak (hb)



- Visual primitives (p): known simple visual concepts.
- Composition rules: (∧ , AND) conjunction, (∨ , OR) disjunction, and (¬, NOT) negation.

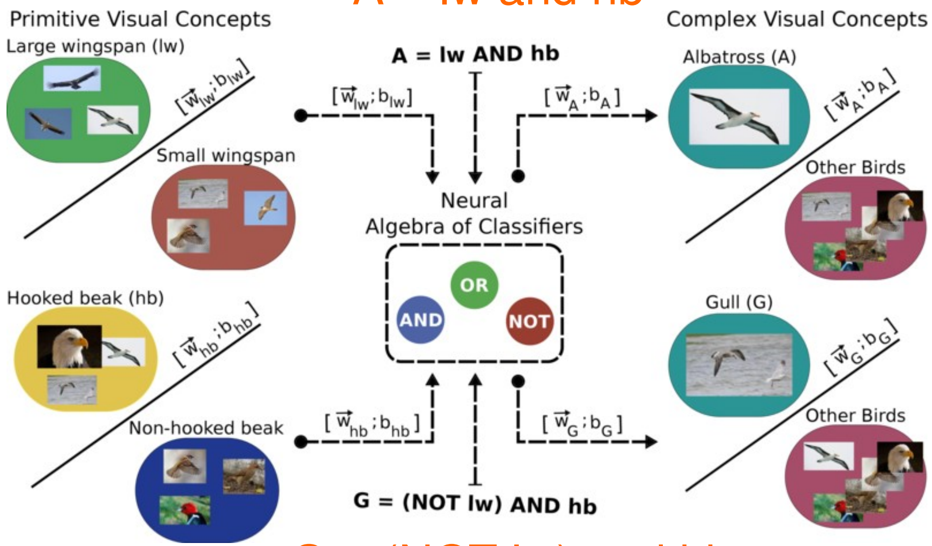
Problem Description

How to synthesize classifiers for arbitrary compositions of visual primitives ?

A = lw and hb

Large wingspan (lw)

Hooked beak (hb)



G = (NOT lw) and hb

- Visual primitives (p): known simple visual concepts.
- Composition rules: (∧ , AND) conjunction, (∨ , OR) disjunction, and (¬, NOT) negation.
- Expressions (e): visual concepts expressed as multiple compositions of primitives and composition rules.

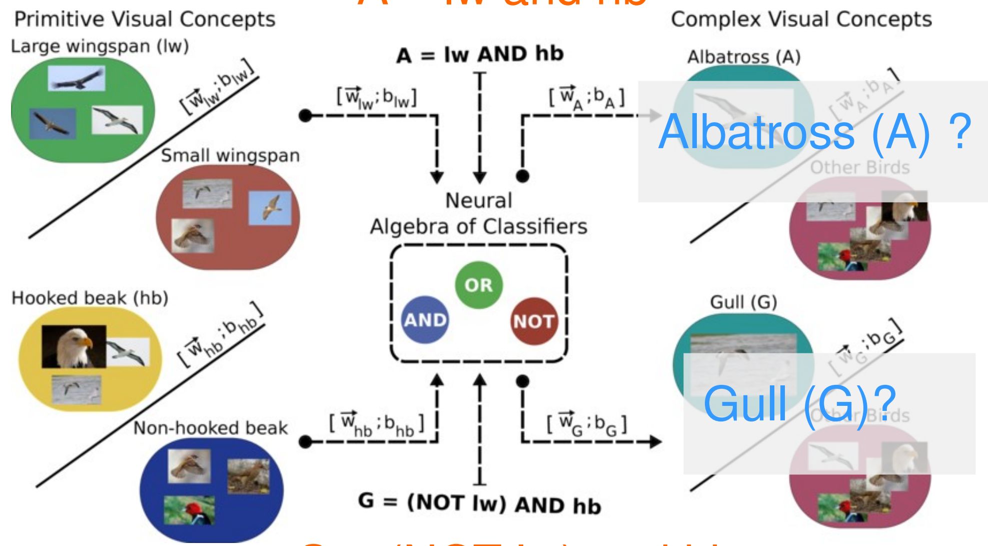
Problem Description

How to synthesize classifiers for arbitrary compositions of visual primitives ?

A = lw and hb

Large wingspan (lw)

Hooked beak (hb)



G = (NOT lw) and hb

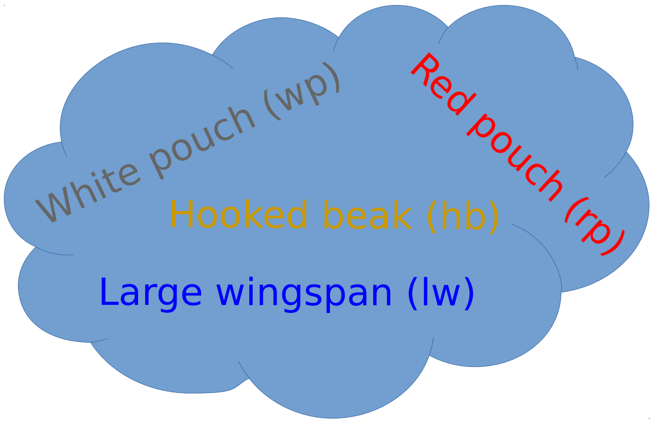
- Visual primitives (p): known simple visual concepts.
- Composition rules: (∧ , AND) conjunction, (∨ , OR) disjunction, and (¬, NOT) negation.
- Expressions (e): visual concepts expressed as multiple compositions of primitives and composition rules.

Traditional Vision System

The current state-of-the-art models for recognition follow a laborious data-driven approach, where complex concepts are learned using thousands of manually labeled examples.

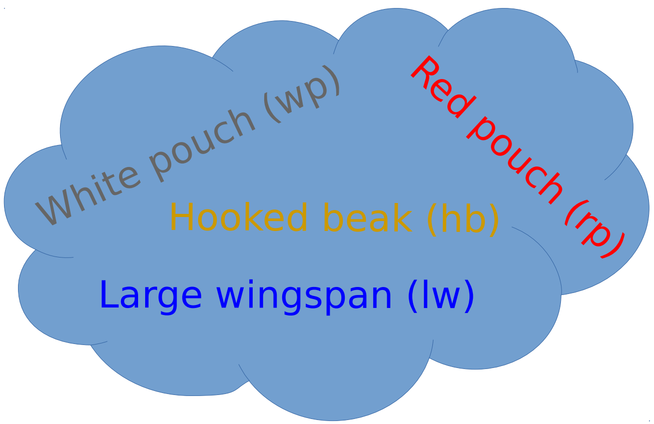
Traditional Vision System

The current state-of-the-art models for recognition follow a laborious data-driven approach, where complex concepts are learned using thousands of manually labeled examples.

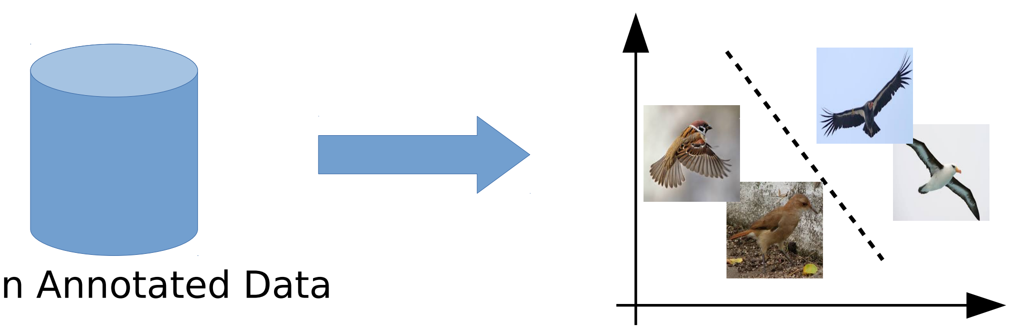


Traditional Vision System

The current state-of-the-art models for recognition follow a laborious data-driven approach, where complex concepts are learned using thousands of manually labeled examples.

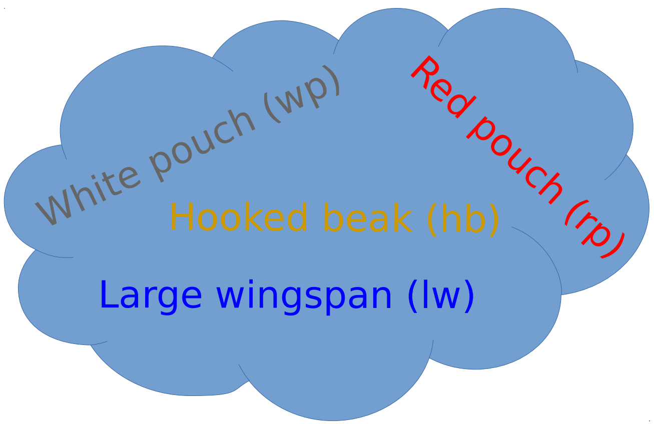


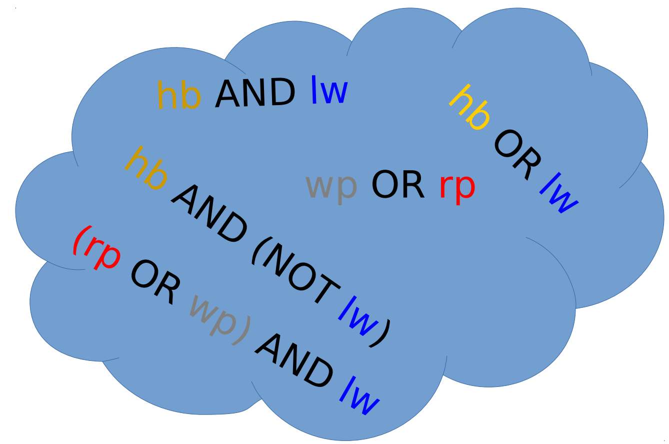
Collect Data + Train a classifier



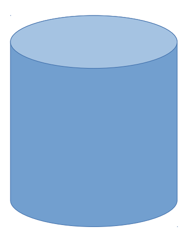
Traditional Vision System

The current state-of-the-art models for recognition follow a laborious data-driven approach, where complex concepts are learned using thousands of manually labeled examples.

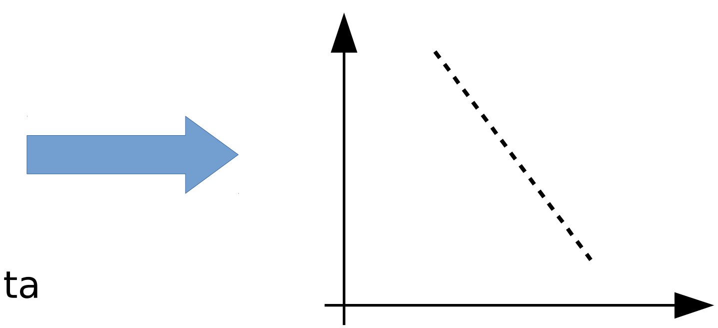




Collect Data + Train a classifier

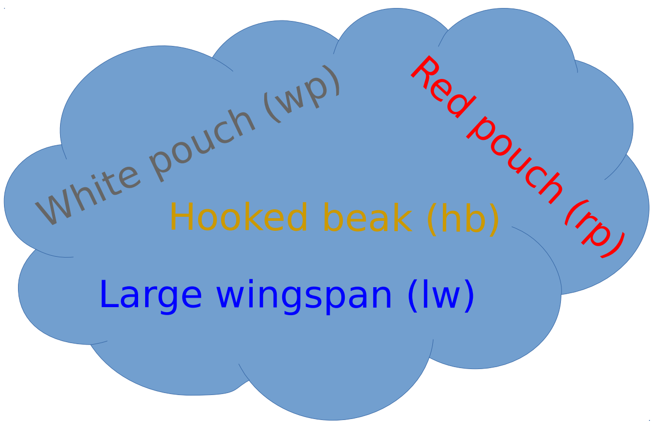


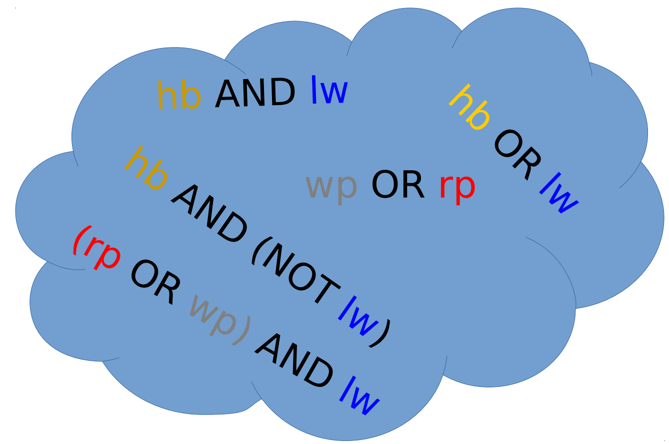
Human Annotated Data



Traditional Vision System

The current state-of-the-art models for recognition follow a laborious data-driven approach, where complex concepts are learned using thousands of manually labeled examples.

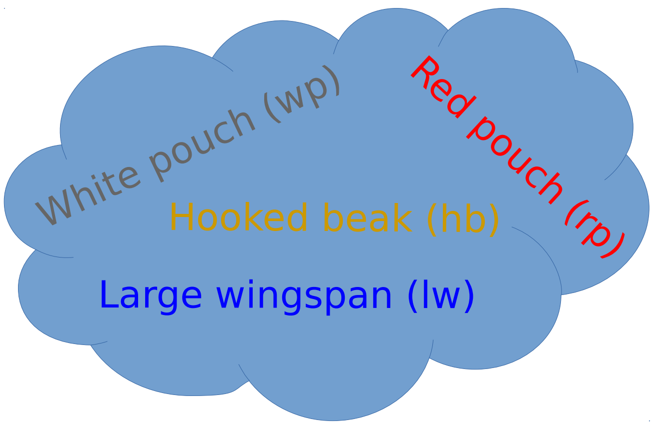


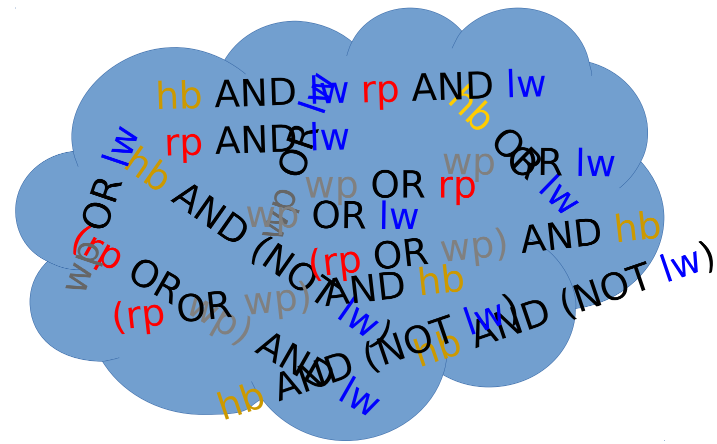


Collect Data + Train a classifier

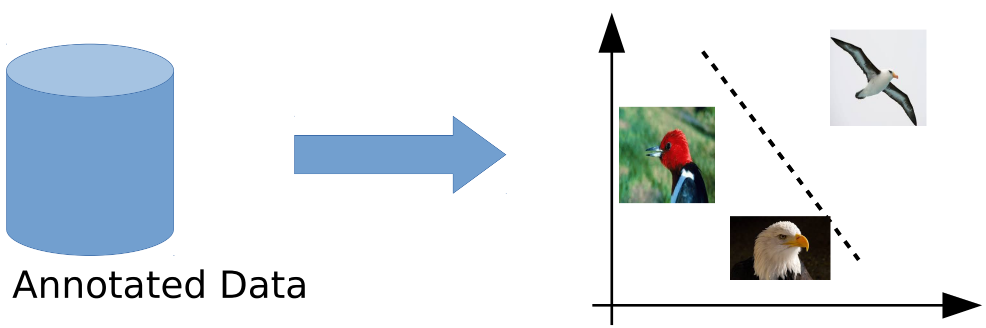
Traditional Vision System

The current state-of-the-art models for recognition follow a laborious data-driven approach, where complex concepts are learned using thousands of manually labeled examples.



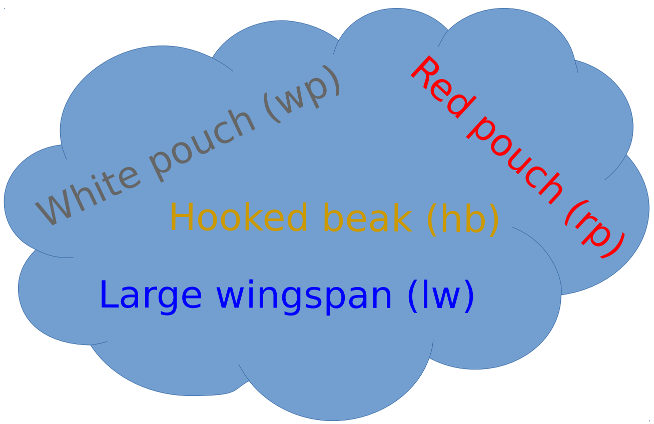


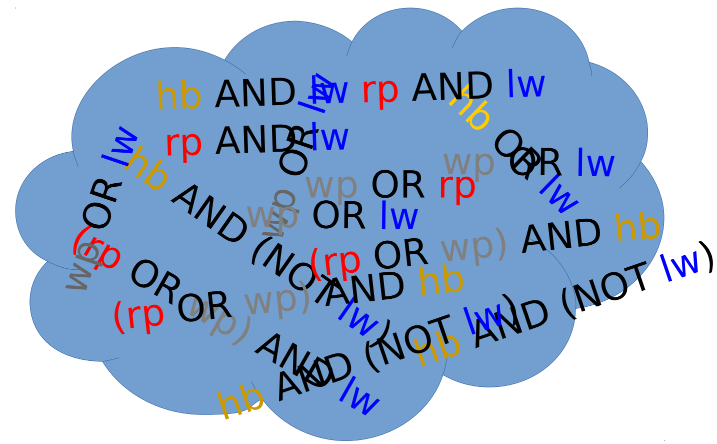
Collect Data + Train a classifier



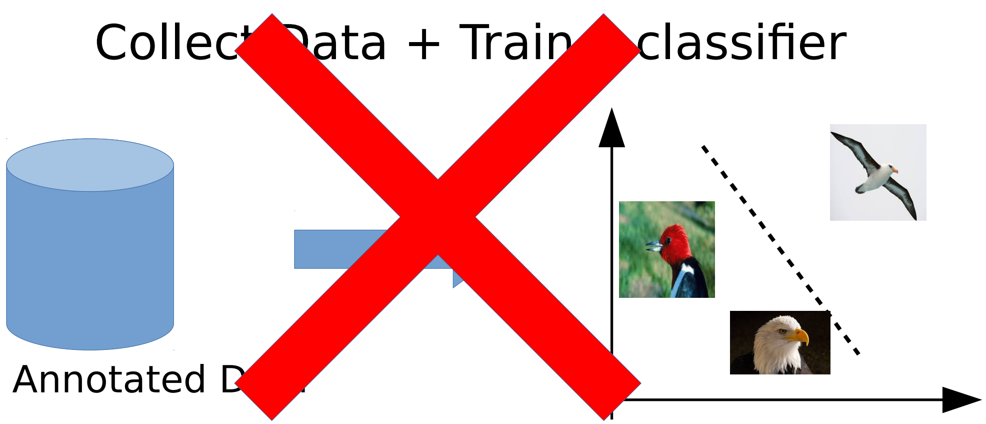
Traditional Vision System

The current state-of-the-art models for recognition follow a laborious data-driven approach, where complex concepts are learned using thousands of manually labeled examples.





Collect Data + Train classifier



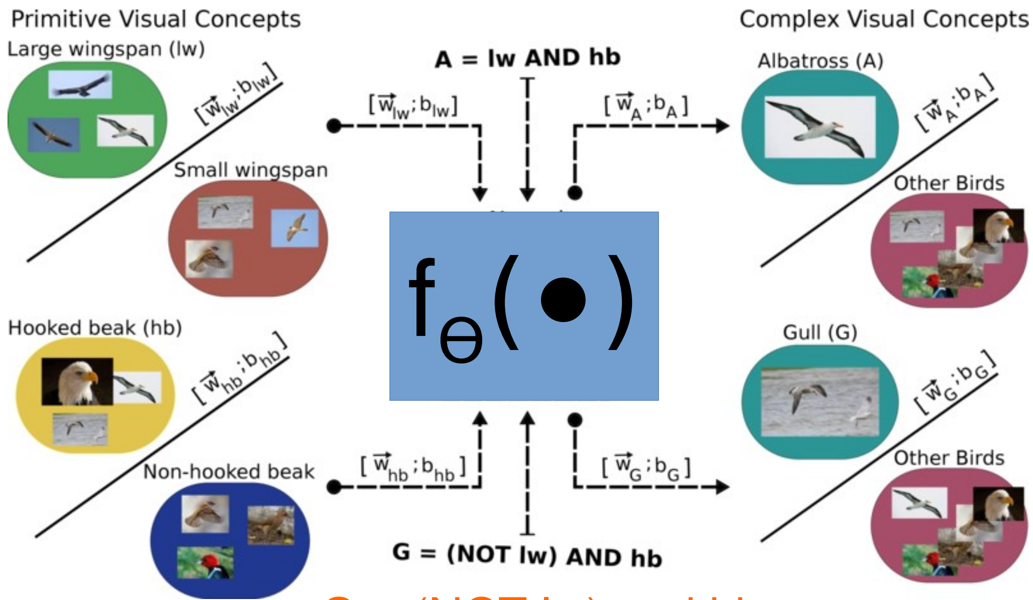
Compositional Model

We propose to learn a function $f_\Theta(\cdot)$ that maps the space of expressions to the space of binary classifiers:

A = lw and hb

Large wingspan (lw)

Hooked beak (hb)



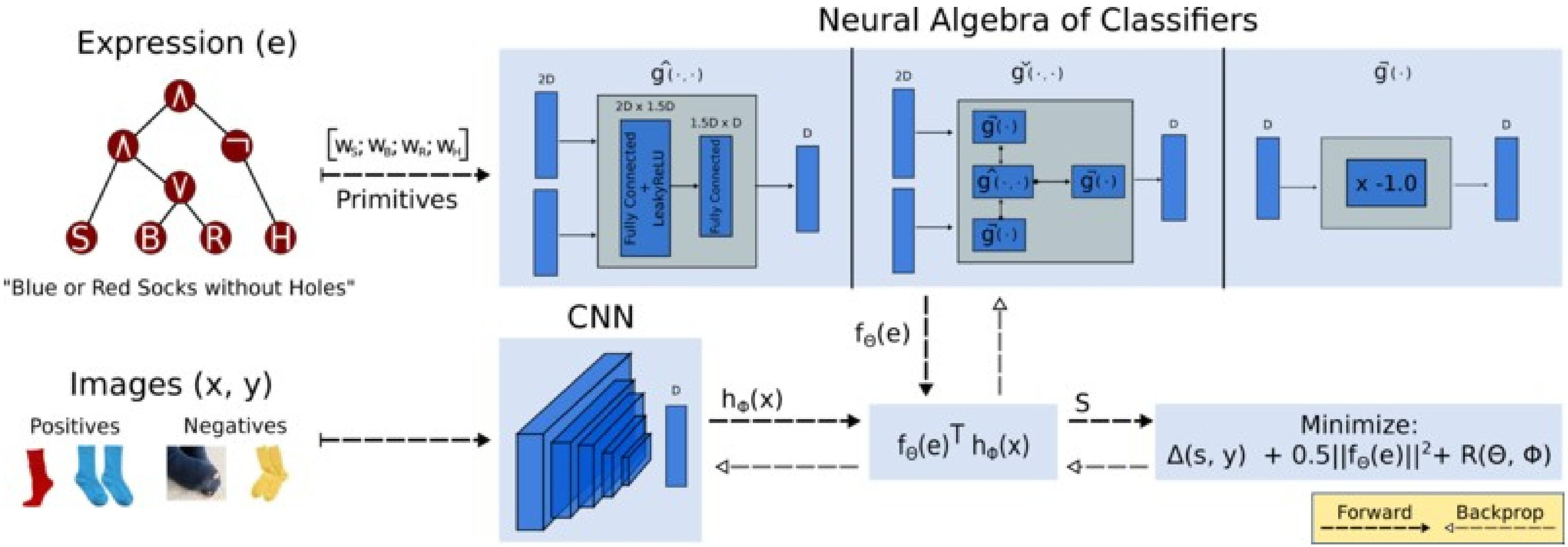
Albatross (A)

Gull (G)

G = (NOT lw) and hb

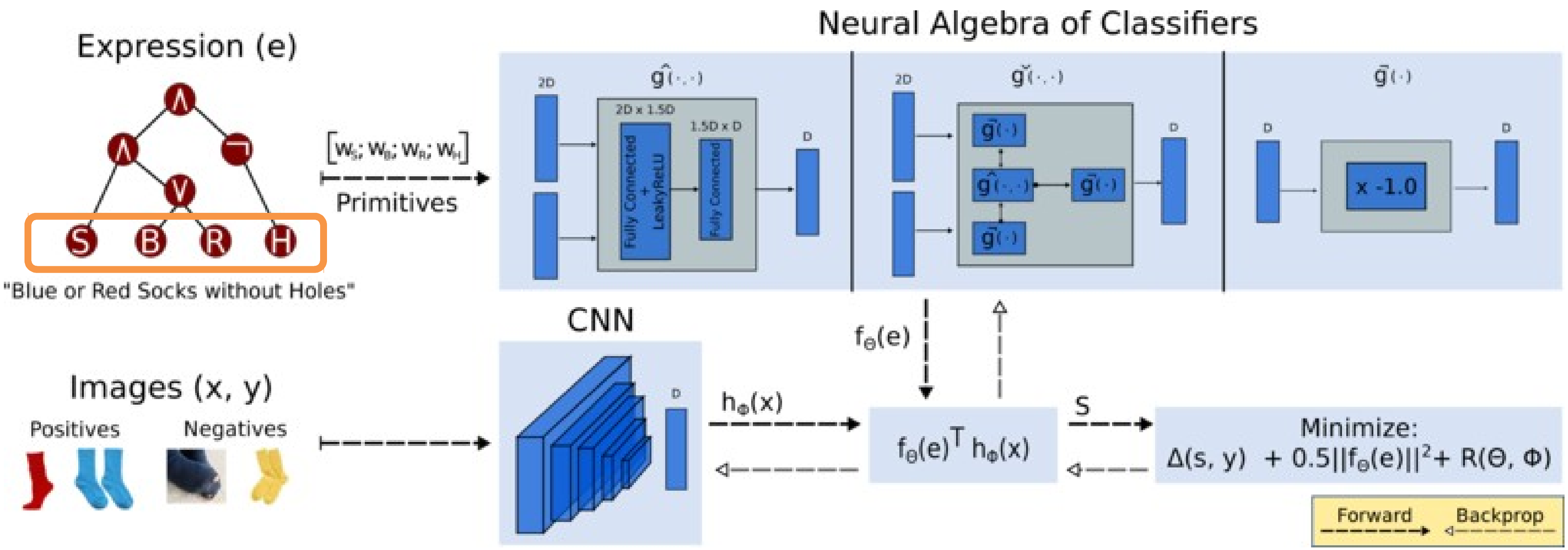
We use a relative small subset of training expressions and rely on the classifier similarity to generalize for unknown expressions.

Approach



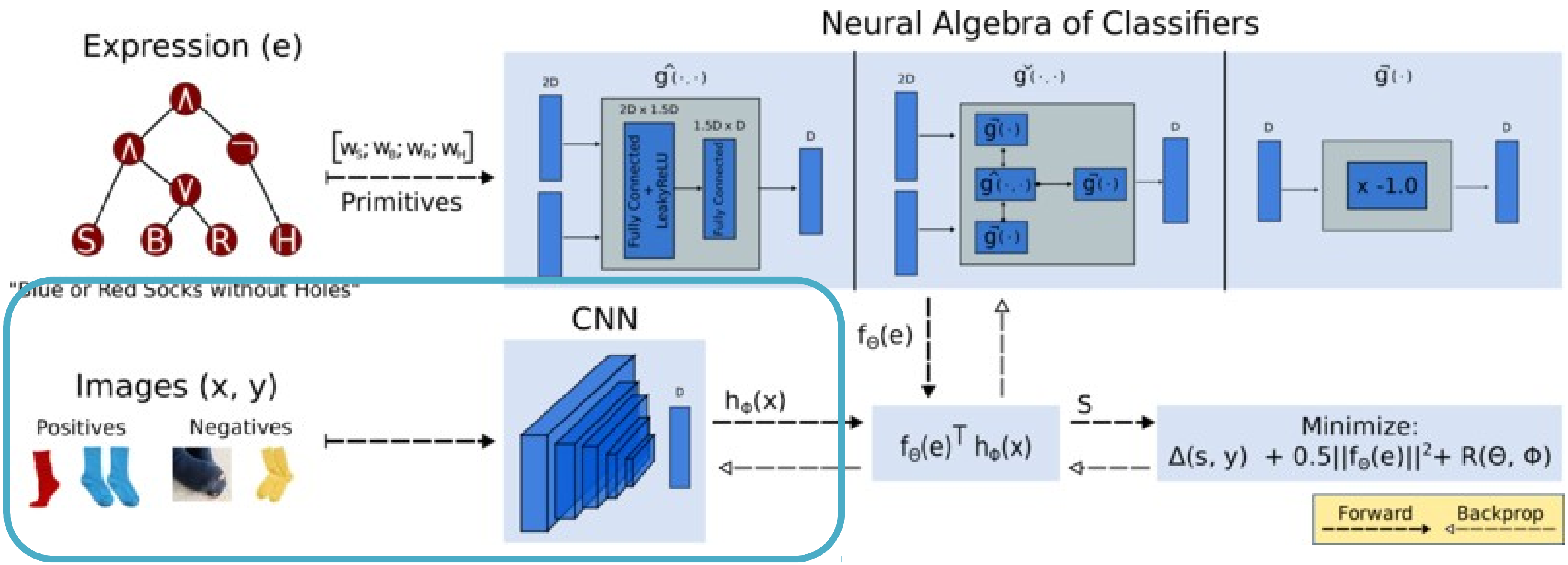
Expression (e)
"Blue or Red Socks without Holes"
Primitives
Neural Algebra of Classifiers
Fully Connected + LeakyReLU
Fully Connected
x -1.0
CNN
Images (x, y)
Positives
Negatives
Minimize:
Forward
Backprop

Approach



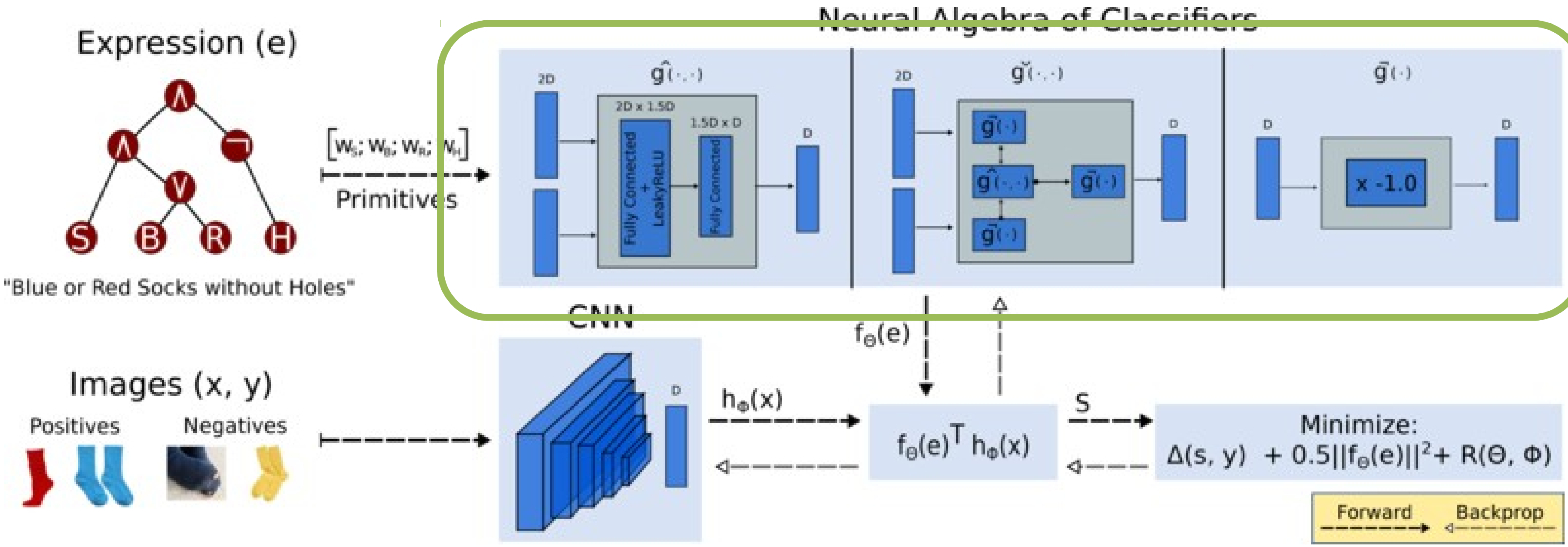
We represent primitives by the parameters of one-vs-all SVM classifiers trained on positives and negatives images of the primitives.

Approach



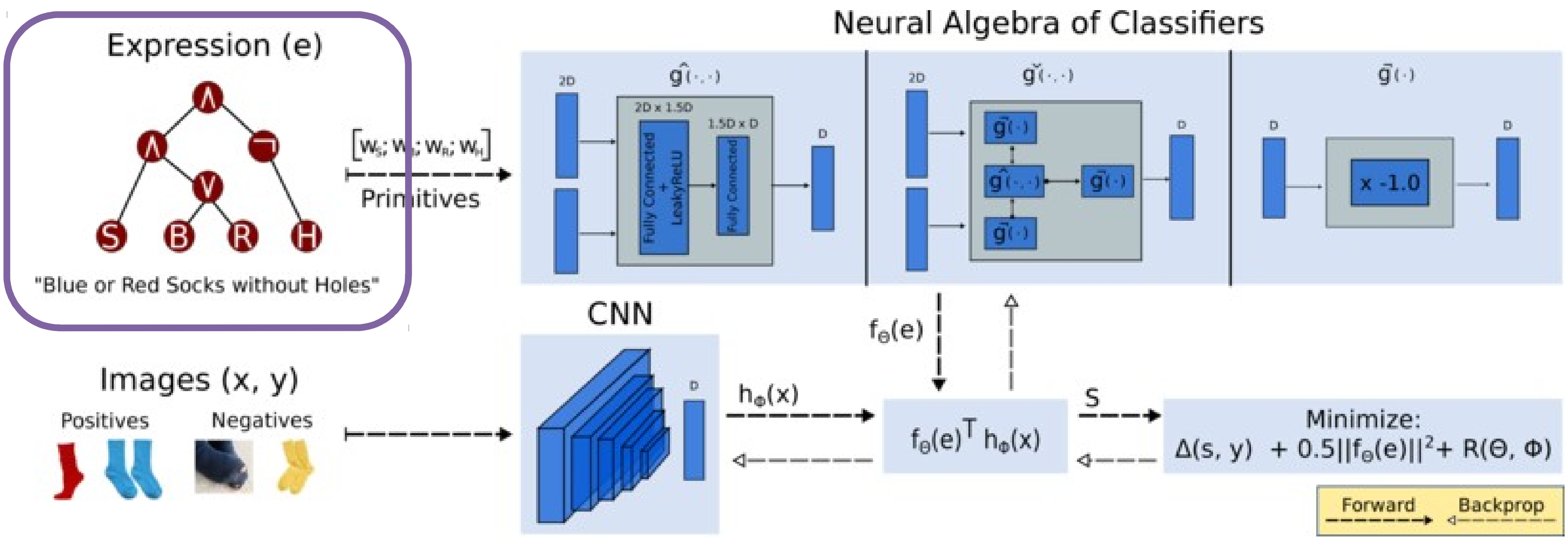
We represent images in a feature space, e.g., CNN features.

Approach



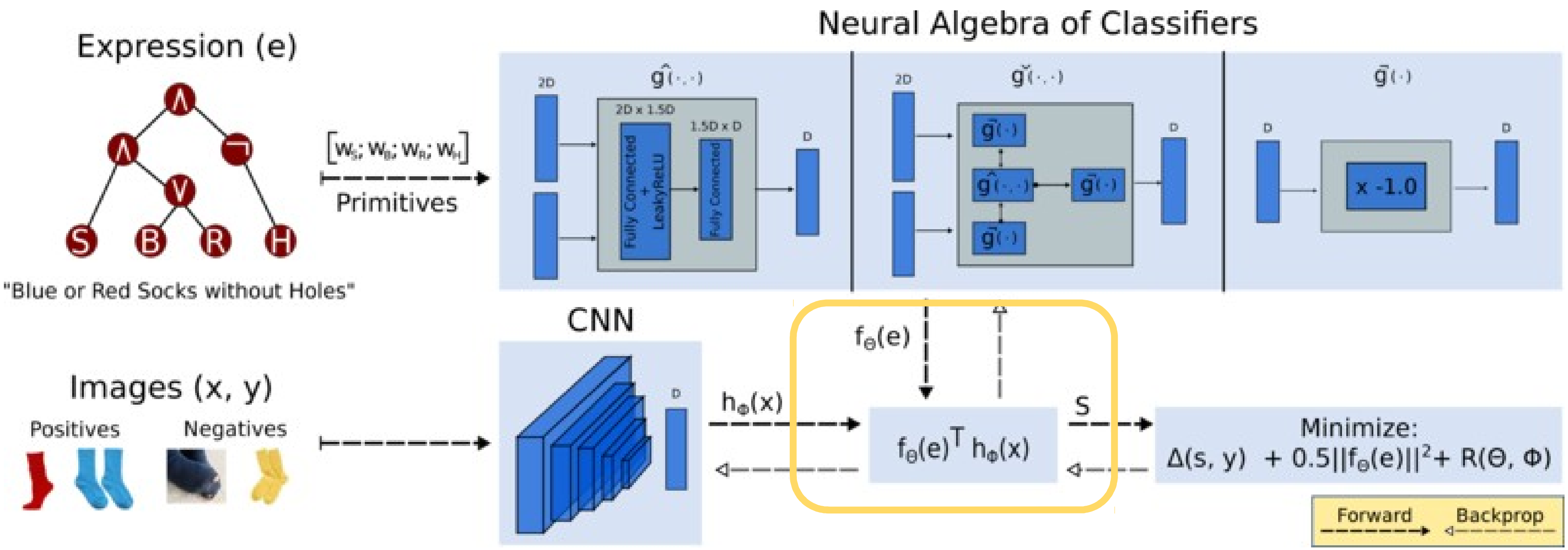
We model our function as a set of composition functions and simplify them using simple analytical relations and De Morgan's laws.

Approach



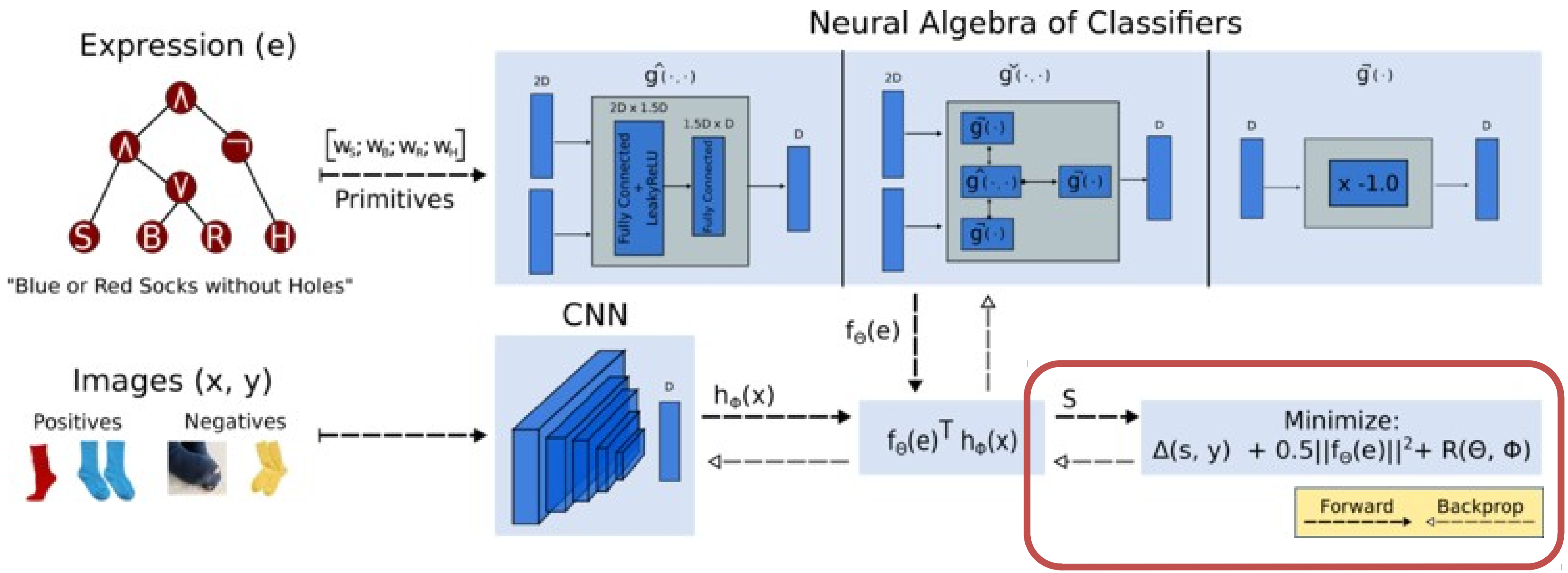
We parse the expression tree applying the composition functions recursively.

Approach



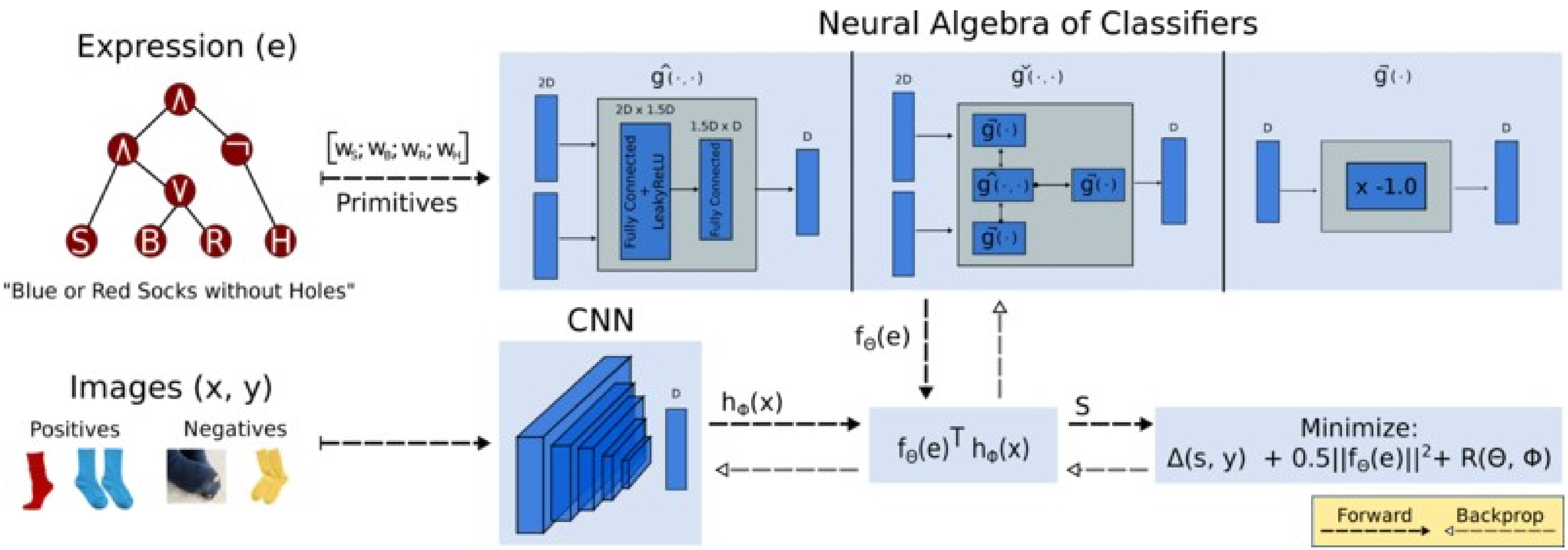
We score images according to the “predicted classifier” for a given expressions.

Approach



We minimize the classification loss of batches of positive and negative images for different training expressions.

Approach

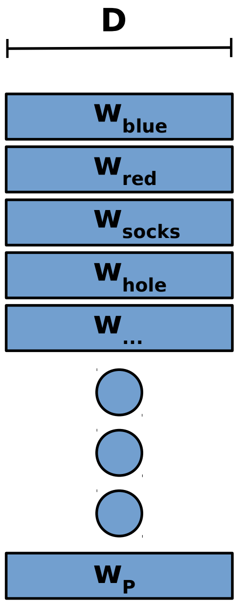


Our method...

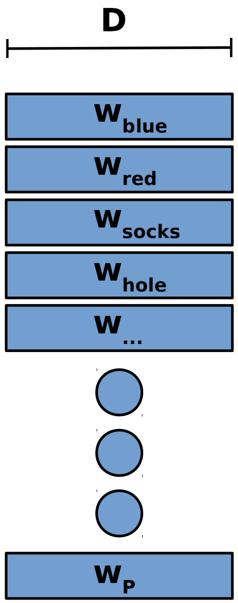
- Synthesizes classifiers for any concept that can be expressed as a boolean expression of primitives.
- Explores correlations, cooccurrences, and contextuality between visual primitives.
- Leverages semantic similarity and compositionality.
- Learns from a subset of expressions and relies on the classifier similarity to generalize for unknown expressions.

Approach

Approach



Approach

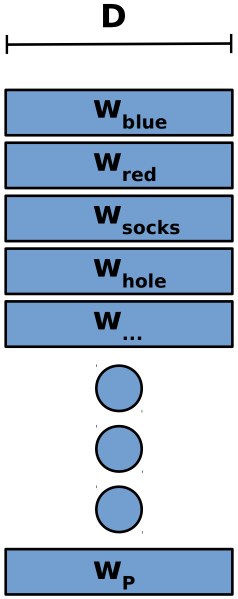


$g^{not}(\bullet): R^D \rightarrow R^D$

$g^{\wedge}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$

$g^{\vee}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$

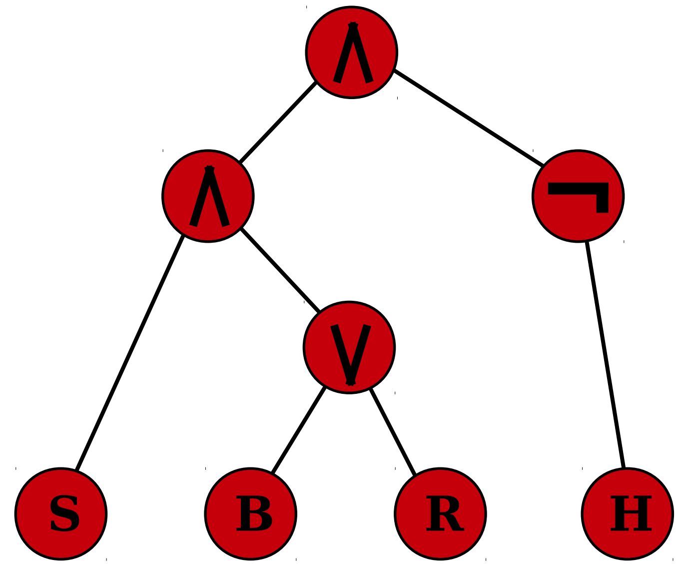
Approach



$g^{not}(\bullet): R^D \rightarrow R^D$

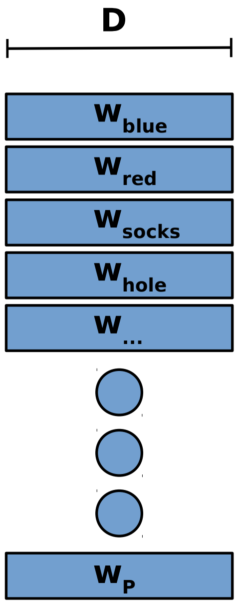
$g^{\wedge}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$

$g^{\vee}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$



Blue or **R**ed **S**ocks Without **H**oles

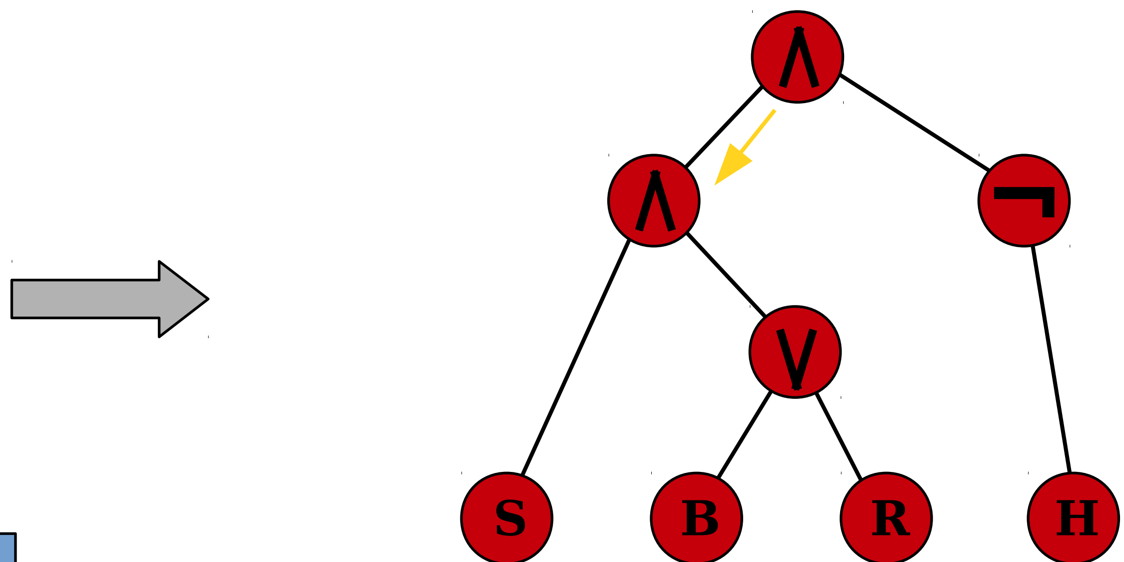
Approach



$g^{not}(\bullet): R^D \rightarrow R^D$

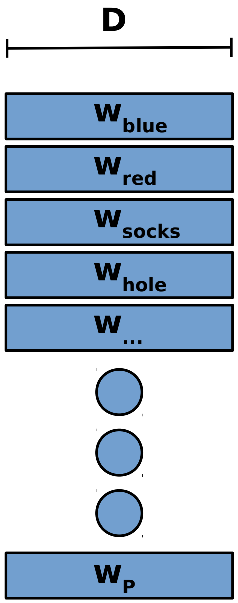
$g^{\wedge}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$

$g^{\vee}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$



Blue or **R**ed **S**ocks Without **H**oles

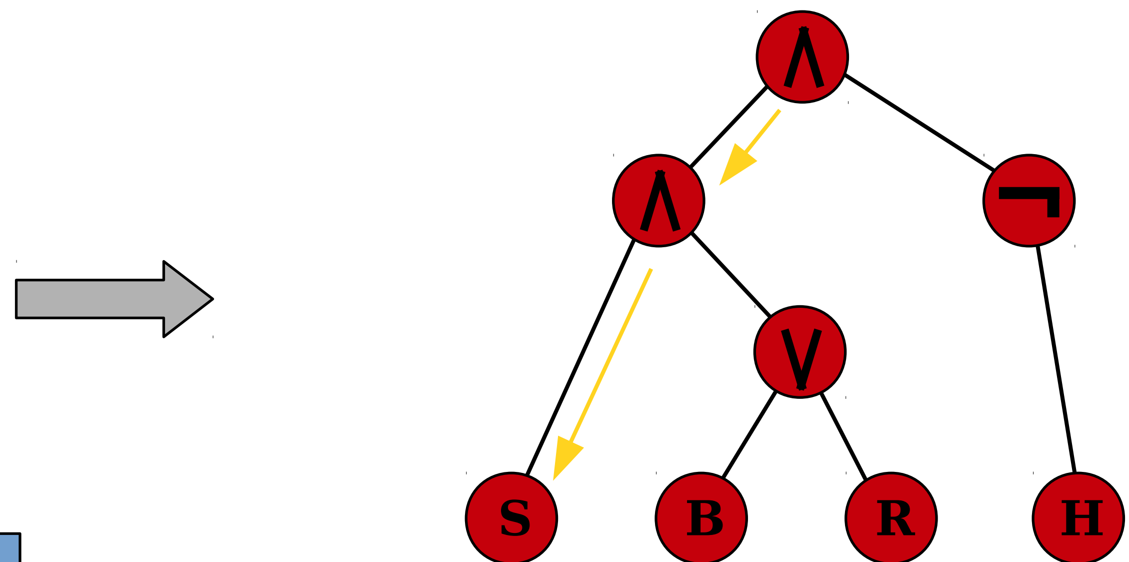
Approach



$g^{not}(\bullet)$: $R^D \to R^D$

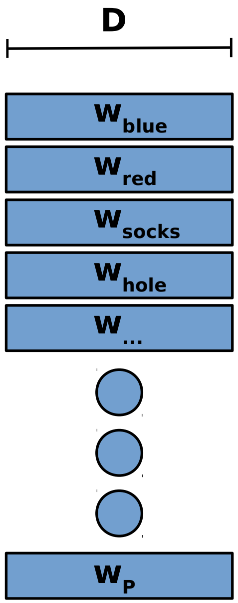
$g^{\wedge}(\bullet, \bullet)$: R^D x $R^D \to R^D$

$g^{\vee}(\bullet, \bullet)$: R^D x $R^D \to R^D$



Blue or **R**ed **S**ocks Without **H**oles

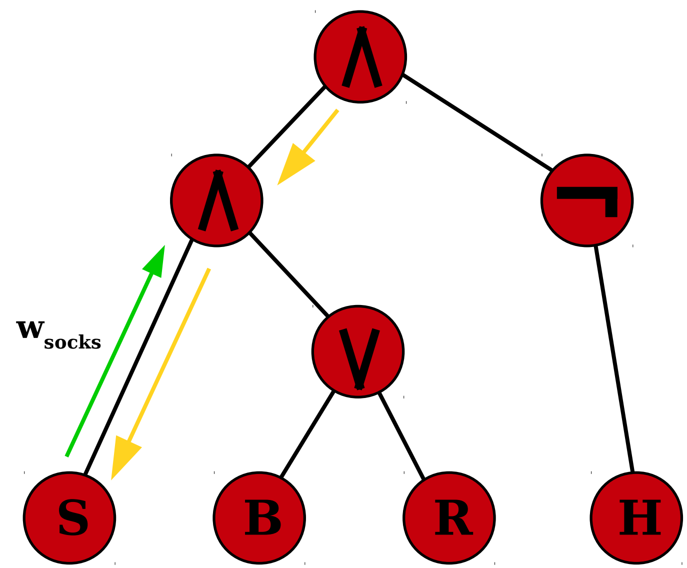
Approach



$g^{not}(\bullet): R^D \rightarrow R^D$

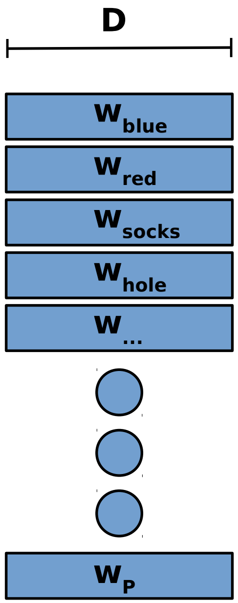
$g^{\wedge}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$

$g^{\vee}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$



Blue or **R**ed **S**ocks Without **H**oles

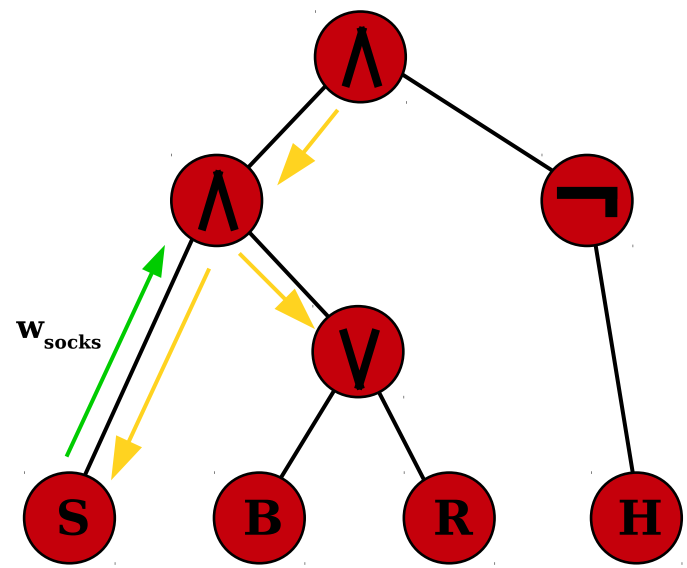
Approach



$g^{not}(\bullet): R^D \rightarrow R^D$

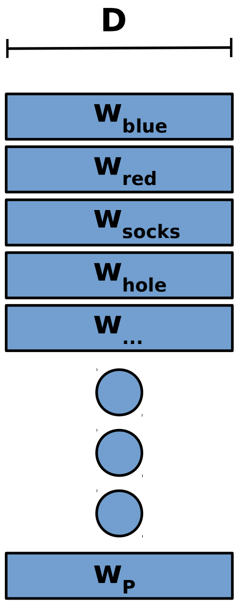
$g^{\wedge}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$

$g^{\vee}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$



Blue or **R**ed **S**ocks Without **H**oles

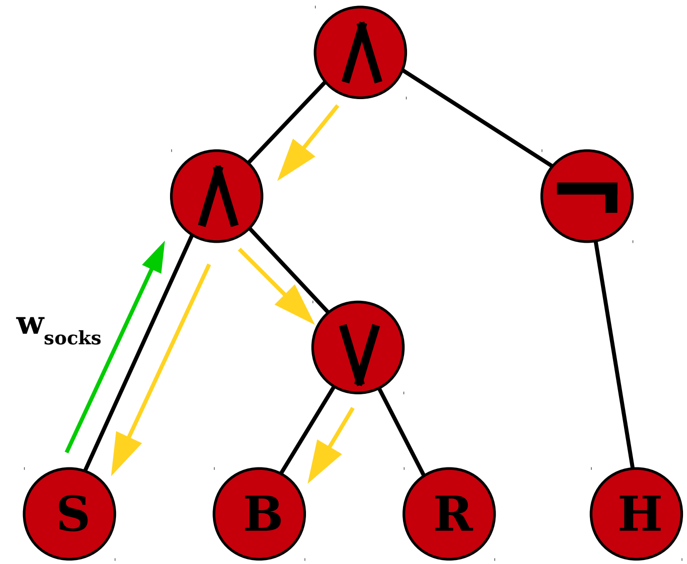
Approach



$g^{not}(\bullet)$: $R^D \rightarrow R^D$

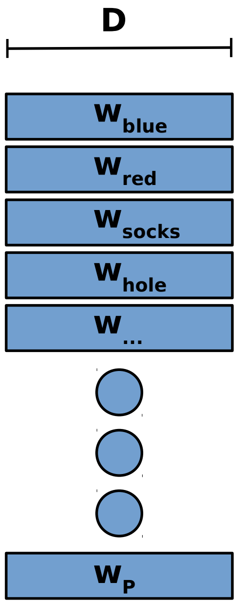
$g^{\wedge}(\bullet, \bullet)$: $R^D \times R^D \rightarrow R^D$

$g^{\vee}(\bullet, \bullet)$: $R^D \times R^D \rightarrow R^D$



Blue or **R**ed **S**ocks Without **H**oles

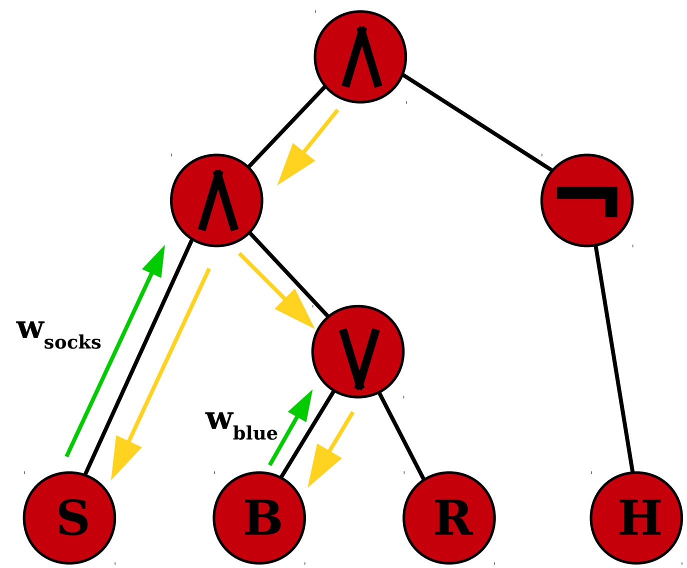
Approach



$g^{not}(\bullet): R^D \rightarrow R^D$

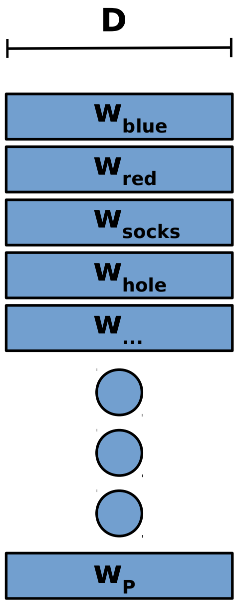
$g^{\wedge}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$

$g^{\vee}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$



Blue or **R**ed **S**ocks Without **H**oles

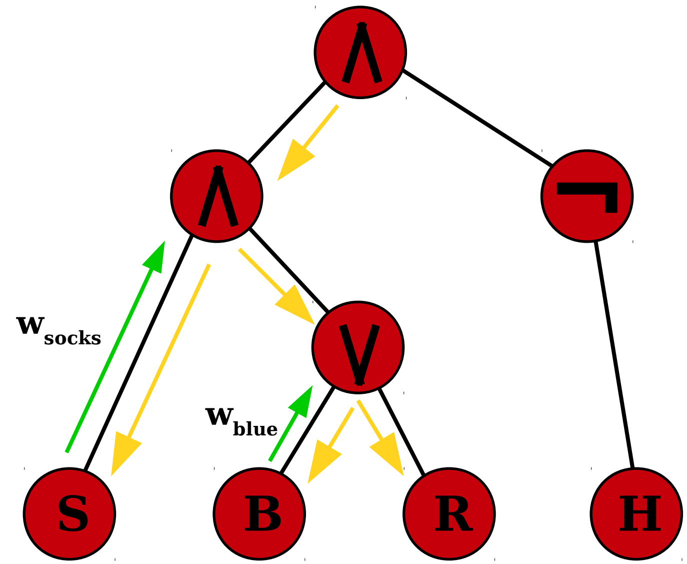
Approach



$g^{not}(\bullet): R^D \rightarrow R^D$

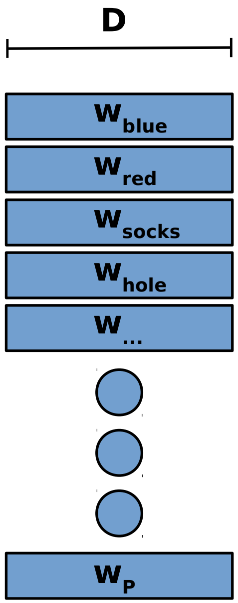
$g^{\wedge}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$

$g^{\vee}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$



Blue or **R**ed **S**ocks Without **H**oles

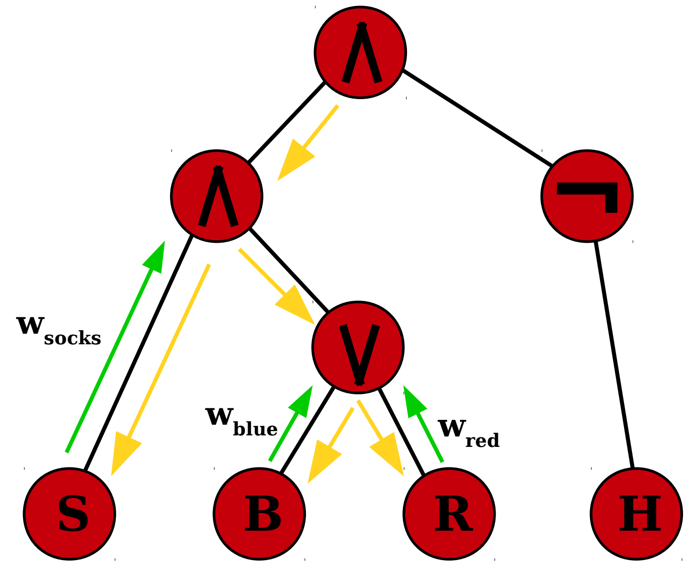
Approach



$$g^{not}(\bullet): R^D \to R^D$$

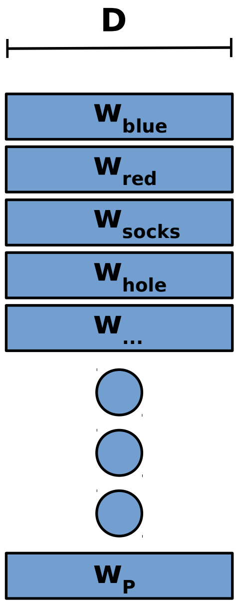
$$g^{\wedge}(\bullet, \bullet): R^D \times R^D \to R^D$$

$$g^{\vee}(\bullet, \bullet): R^D \times R^D \to R^D$$



Blue or **R**ed **S**ocks Without **H**oles

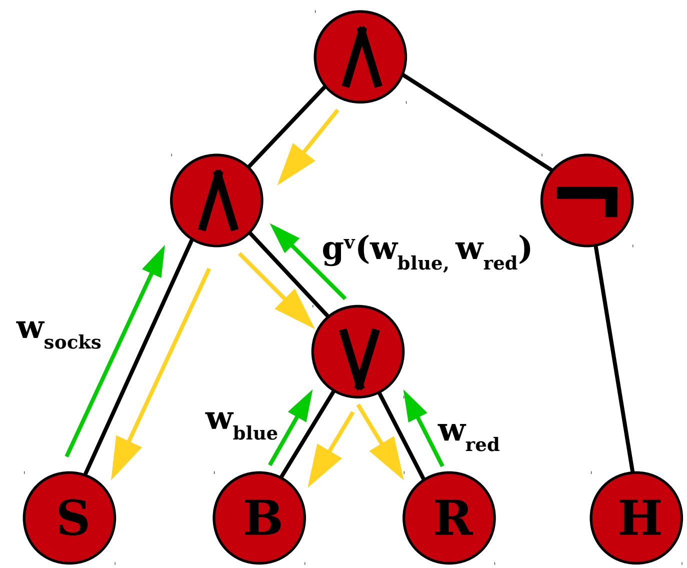
Approach



$g^{not}(\bullet): R^D \rightarrow R^D$

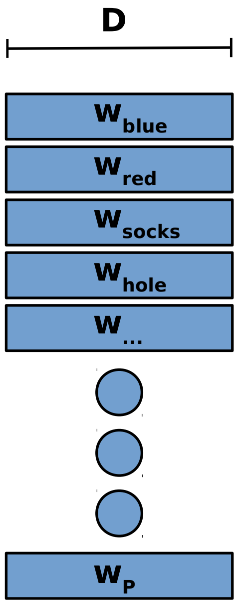
$g^{\wedge}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$

$g^{\vee}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$



Blue or **R**ed **S**ocks Without **H**oles

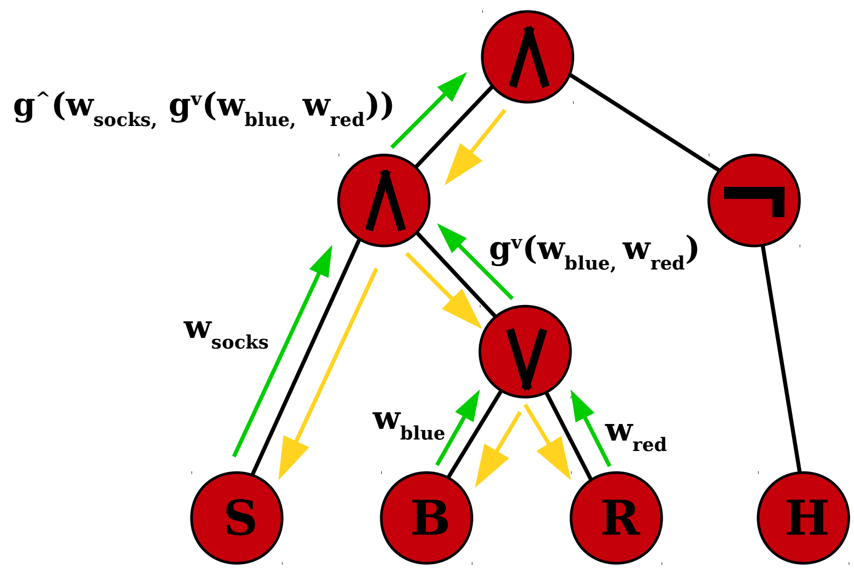
Approach



$g^{not}(\bullet): R^D \rightarrow R^D$

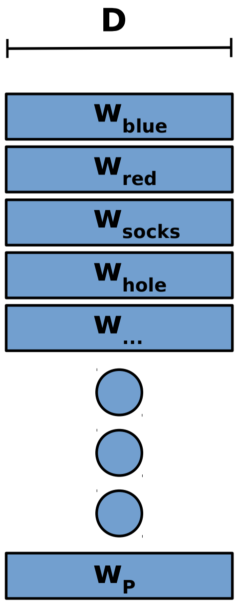
$g^{\wedge}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$

$g^{\vee}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$



Blue or **R**ed **S**ocks Without **H**oles

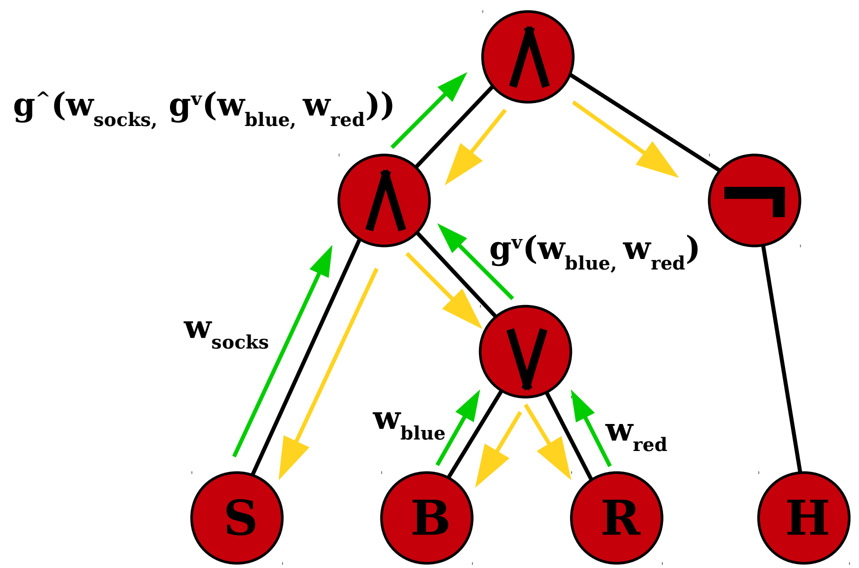
Approach



$g^{not}(\bullet)$: $R^D \rightarrow R^D$

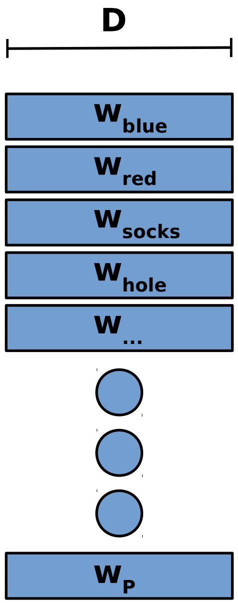
$g^{\wedge}(\bullet, \bullet)$: $R^D \times R^D \rightarrow R^D$

$g^{v}(\bullet, \bullet)$: $R^D \times R^D \rightarrow R^D$



Blue or **R**ed **S**ocks Without **H**oles

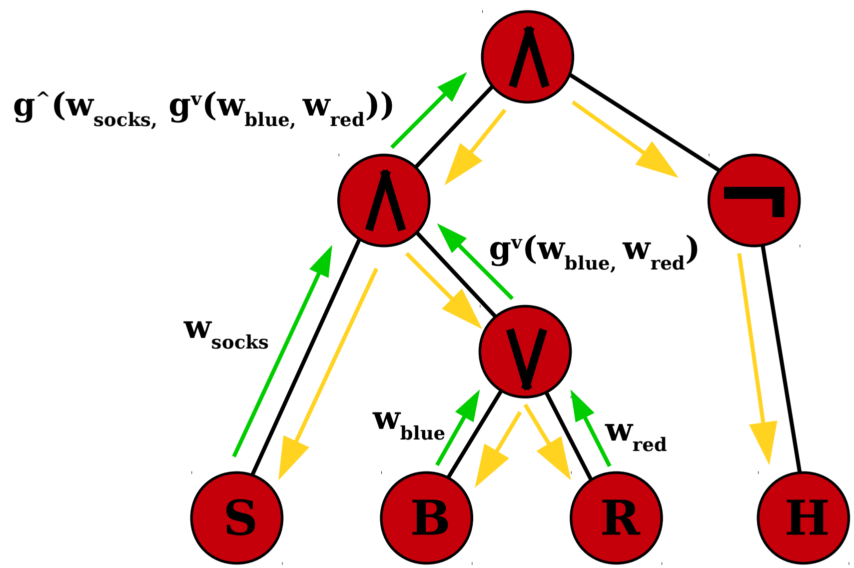
Approach



$g^{not}(\bullet): R^D \rightarrow R^D$

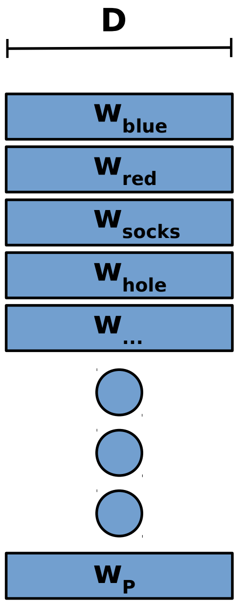
$g^{\wedge}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$

$g^{\vee}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$



Blue or **R**ed **S**ocks Without **H**oles

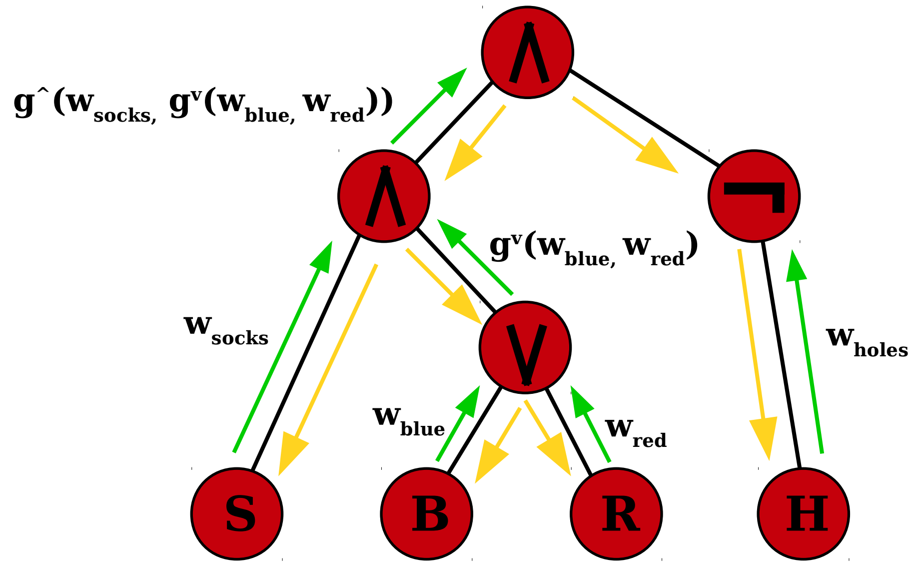
Approach



$g^{not}(\bullet): R^D \rightarrow R^D$

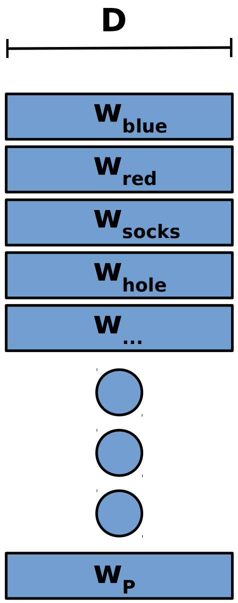
$g^{\wedge}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$

$g^{\vee}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$



Blue or **R**ed **S**ocks Without **H**oles

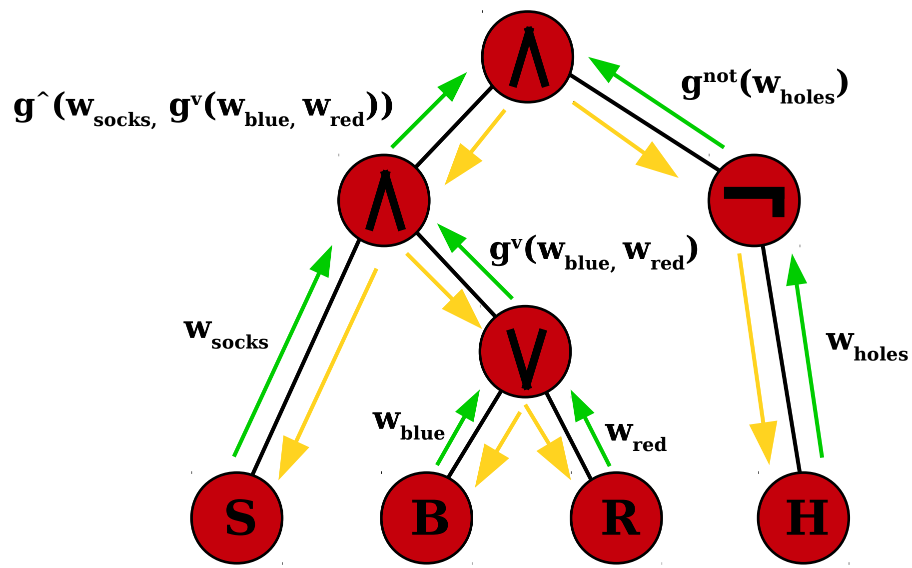
Approach



$g^{not}(\bullet): R^D \rightarrow R^D$

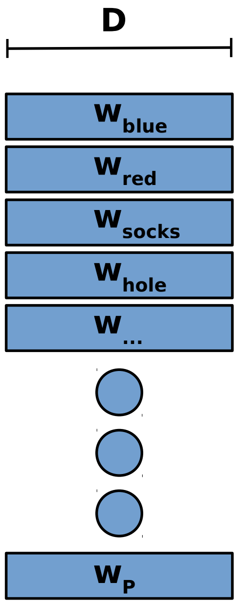
$g^{\wedge}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$

$g^{\vee}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$



Blue or **R**ed **S**ocks Without **H**oles

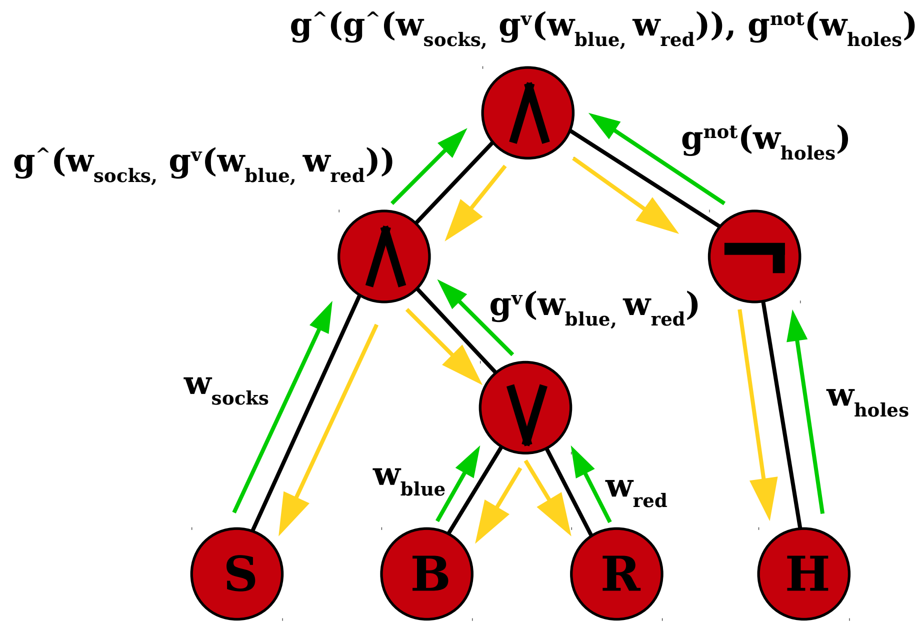
Approach



$$g^{not}(\bullet): R^D \rightarrow R^D$$

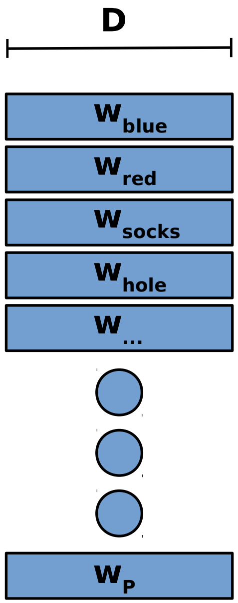
$$g^{\wedge}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$$

$$g^{\vee}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$$



Blue or **R**ed **S**ocks Without **H**oles

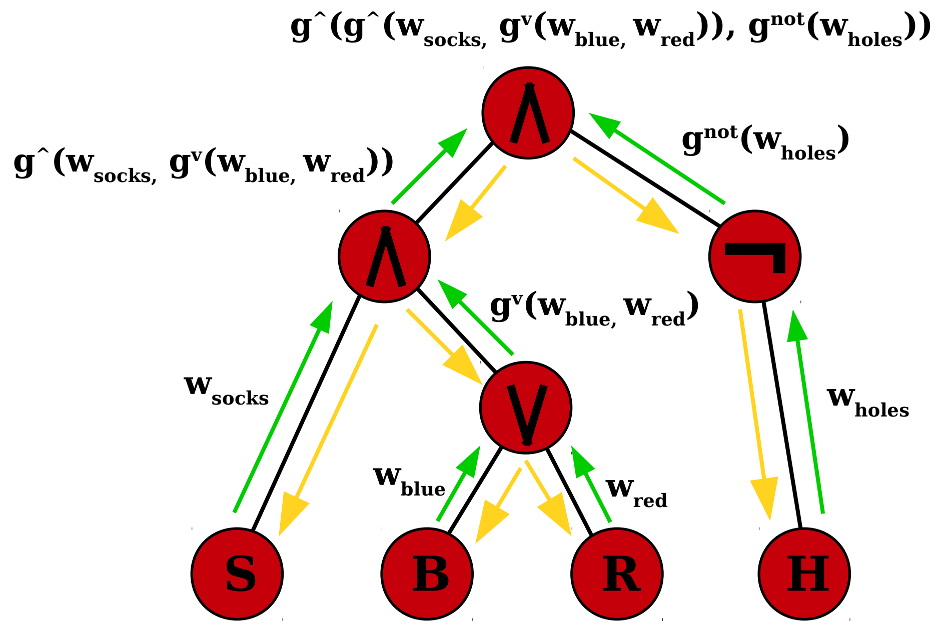
Approach



$g^{not}(\bullet): R^D \to R^D$

$g^{\wedge}(\bullet, \bullet): R^D \times R^D \to R^D$

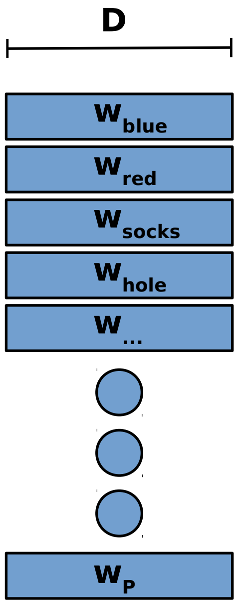
$g^{\vee}(\bullet, \bullet): R^D \times R^D \to R^D$



Blue or **R**ed **S**ocks Without **H**oles

$$f(e) = g^{\wedge}(g^{\wedge}(w_{socks}, g^{v}(w_{blue}, w_{red})), g^{not}(w_{holes}))$$

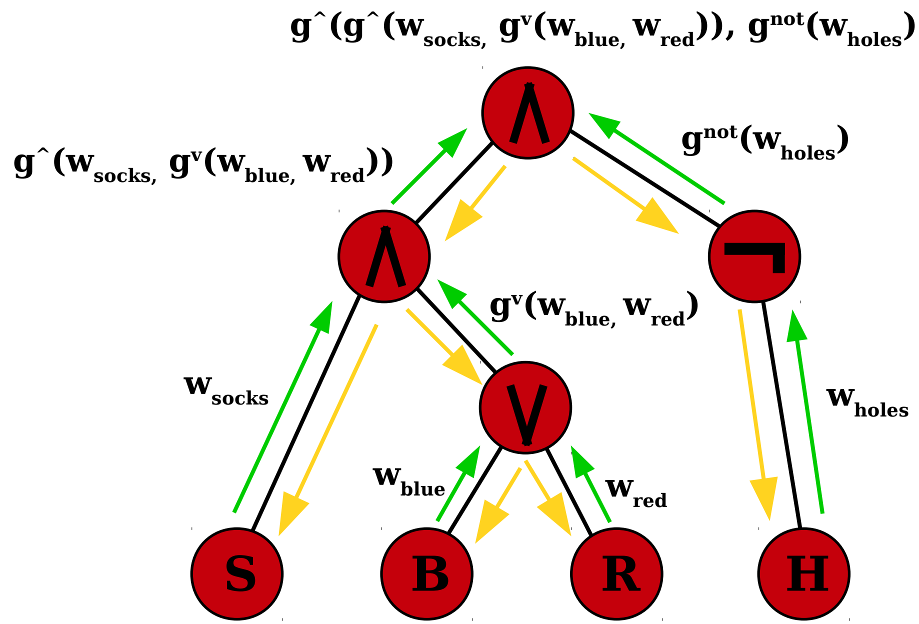
Approach



$g^{not}(\bullet): R^D \rightarrow R^D$

$g^{\wedge}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$

$g^{v}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$



Blue or **R**ed **S**ocks Without **H**oles

$$f(e) = g^{\wedge}(g^{\wedge}(w_{socks}, g^{v}(w_{blue}, w_{red})), g^{not}(w_{holes})) = w_e \in R^D$$

Approach

D

w_{blue}

w_{red}

w_{socks}

w_{hole}

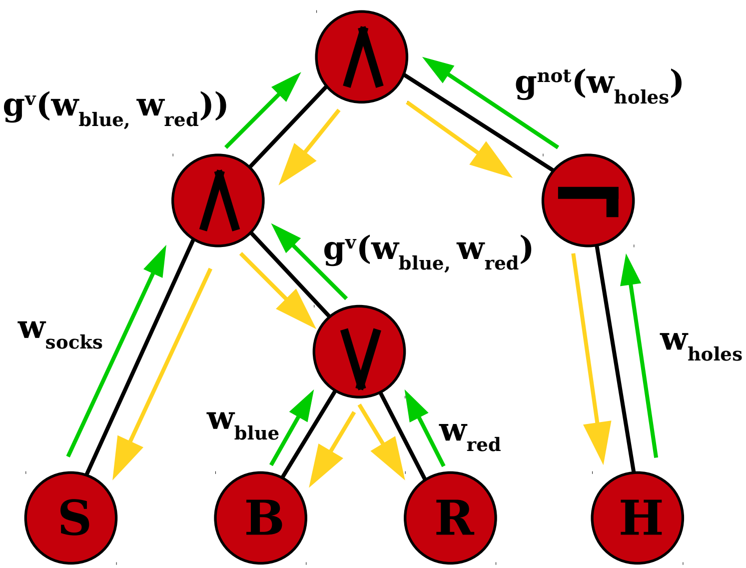
$w_{...}$

w_{P}

$g^{not}(\bullet): R^D \rightarrow R^D$

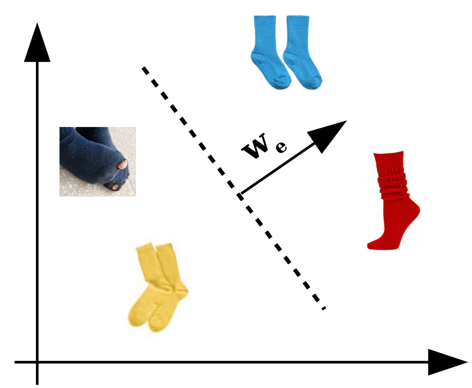
$g^{\wedge}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$

$g^{\vee}(\bullet, \bullet): R^D \times R^D \rightarrow R^D$



Blue or **R**ed **S**ocks Without **H**oles

$$f(e) = g^{\wedge}(g^{\wedge}(w_{socks}, g^{\vee}(w_{blue}, w_{red})), g^{not}(w_{holes}))$$
$$= w_e \in R^D$$



Approach

We model our function as a set of composition functions and simplify them using simple analytical relations and De Morgan's laws.

$$g_\theta^\wedge(w_a, w_b) = \text{Neural Network}(w_a, w_b)$$

$$g^\neg(w) = -w$$

$$g^\vee(w_a, w_b) = g^\neg\left(g^\wedge\left(g^\neg(w_a), g^\neg(w_b)\right)\right)$$

Recap

We propose to learn a function $f_\Theta(\cdot)$ that maps the space of expressions to the space of binary classifiers:

Recap

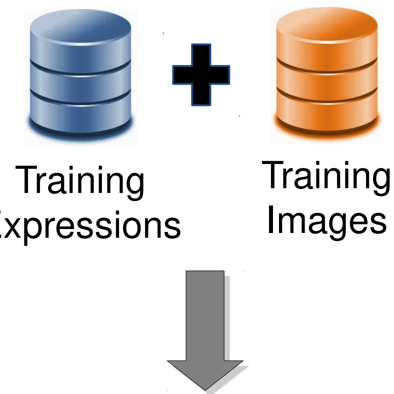
We propose to learn a function $f_\theta(\cdot)$ that maps the space of expressions to the space of binary classifiers:

Training:

Recap

We propose to learn a function $f_\Theta(\cdot)$ that maps the space of expressions to the space of binary classifiers:

Training:



Training Expressions

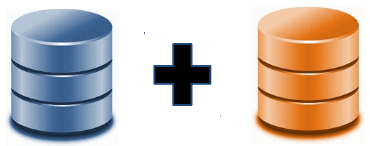
Training Images

Learn...
1) Primitives: $w_p \in R^D$
2) Image representation: $h_\phi(x) \in R^D$
3) Mapping function: $f_\Theta(e) \in R^D$

Recap

We propose to learn a function $f_\Theta(\cdot)$ that maps the space of expressions to the space of binary classifiers:

Training:



Training Expressions

Training Images

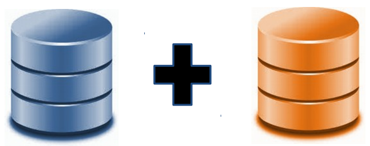
Learn...
1) Primitives: $w_p \in R^D$
2) Image representation: $h_\phi(x) \in R^D$
3) Mapping function: $f_\Theta(e) \in R^D$

Test:

Recap

We propose to learn a function $f_\Theta(\cdot)$ that maps the space of expressions to the space of binary classifiers:

Training:



Training Expressions | Training Images

Learn...
1) Primitives: $w_p \in R^D$
2) Image representation: $h_\phi(x) \in R^D$
3) Mapping function: $f_\Theta(e) \in R^D$

Test:

Training Expressions | Validation Images

Recap

We propose to learn a function $f_{\theta}(\cdot)$ that maps the space of expressions to the space of binary classifiers:

Training:

Training Expressions

Training Images

Test:

Training Expressions

Validation Images

Test Expressions

Test Images

Learn...
1) Primitives: $w_p \in R^D$
2) Image representation: $h_{\phi}(x) \in R^D$
3) Mapping function: $f_{\theta}(e) \in R^D$

Recap

We propose to learn a function $f_\Theta(\cdot)$ that maps the space of expressions to the space of binary classifiers:

Training:

Training Expressions + Training Images

Learn...
1) Primitives: $w_p \in R^D$
2) Image representation: $h_\Phi(x) \in R^D$
3) Mapping function: $f_\Theta(e) \in R^D$

Test:

Training Expressions + Validation Images

Test Expressions + Test Images

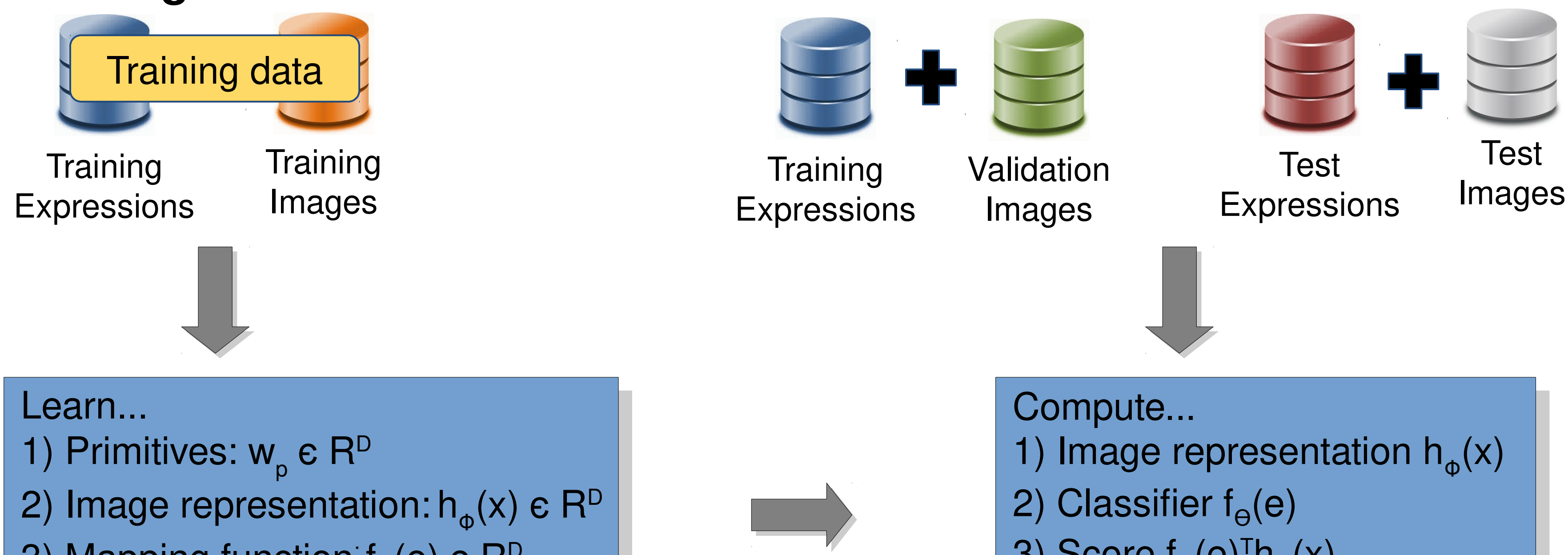
Compute...
1) Image representation $h_\Phi(x)$
2) Classifier $f_\Theta(e)$
3) Score $f_\Theta(e)^T h_\Phi(x)$

Recap

We propose to learn a function $f_\Theta(\cdot)$ that maps the space of expressions to the space of binary classifiers:

Training:

Test:



Learn...
1) Primitives: $w_p \in R^D$
2) Image representation: $h_\phi(x) \in R^D$
3) Mapping function: $f_\Theta(e) \in R^D$

Compute...
1) Image representation $h_\phi(x)$
2) Classifier $f_\Theta(e)$
3) Score $f_\Theta(e)^T h_\phi(x)$

Recap

We propose to learn a function $f_\Theta(\cdot)$ that maps the space of expressions to the space of binary classifiers:

Training:

Training data

Training Expressions

Training Images

Test:

Training Expressions + Validation Images

Test Expressions + Test Images

Learn...
1) Primitives: $w_p \in R^D$
2) Image representation: $h_\phi(x) \in R^D$
3) Mapping function: $f_\Theta(e) \in R^D$

Generalize

Compute...
1) Image representation $h_\phi(x)$
2) Classifier $f_\Theta(e)$
3) Score $f_\Theta(e)^T h_\phi(x)$

Recap

We propose to learn a function $f_\Theta(\cdot)$ that maps the space of expressions to the space of binary classifiers:

Training:

Training data

Training Expressions

Training Images

Learn...
1) Primitives: $w_p \in R^D$
2) Image representation: $h_\Phi(x) \in R^D$
3) Mapping function: $f_\Theta(e) \in R^D$

Generalize

Test:

Known exp.

Training Expressions

Validation Images

Unknown exp.

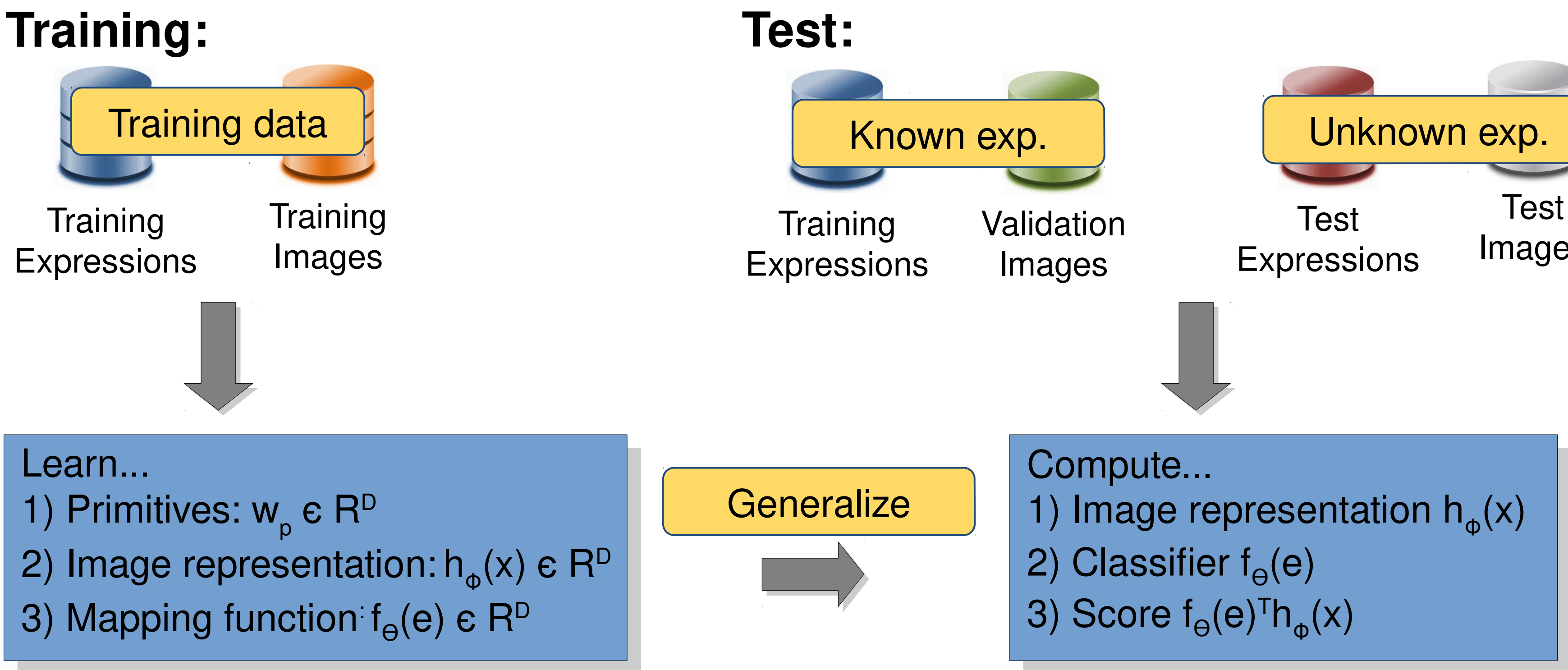
Test Expressions

Test Images

Compute...
1) Image representation $h_\Phi(x)$
2) Classifier $f_\Theta(e)$
3) Score $f_\Theta(e)^T h_\Phi(x)$

Recap

We propose to learn a function $f_\theta(\cdot)$ that maps the space of expressions to the space of binary classifiers:



We use a relative small subset of training expressions and rely on the classifier similarity to generalize for unknown expressions.

Quantitative Experiments

Quantitative Experiments

Baselines:

Chance: Random guess.

Supervised: SVMs trained to discriminate images according to training expressions.

Independent Classifiers:

- P(a AND b) = P(a) x P(b)
- P(a OR b) = P(a) + P(b) - (P(a) x P(b))
- P(NOT a) = 1 - P(a)

Quantitative Experiments

Baselines:

Chance: Random guess.

Supervised: S… inate images accordi… ons.

Cannot generalize

Independent Classifiers:

- P(a AND b) = P(a) x P(b)
- P(a … a) x P(b))
- P(NOT a) = 1 - P(a)

Assume independence

Quantitative Experiments

Baselines:

Chance: Random guess.

Supervised: S… …inate images accordi… …ons.

Cannot generalize

Independent Classifiers:

- P(a AND b) = P(a) x P(b)
- P(a … a) x P(b))
- P(NOT a) = 1 - P(a)

Assume independence

a OR b

a AND b

Quantitative Experiments

Baselines:

Chance: Random guess.

Supervised: S … inate images accordi … ons.

Cannot generalize

Independent Classifiers:

- P(a AND b) = P(a) x P(b)
- P(a … a) x P(b))
- P(NOT a) = 1 - P(a)

Assume independence

a OR b

Known | Unknown

a AND b

Known | Unknown

Quantitative Experiments

AUSTRALIAN CENTRE FOR ROBOTIC VISION

Australian National University

Baselines:

Chance: Random guess.

Supervised: S inate images accordi ons.

Cannot generalize

Independent Classifiers:

- P(a AND b) = P(a) x P(b)
- P(a a) x P(b))
- P(NOT a) = 1 - P(a)

Assume independence

Table 1. Evaluating known/unknown disjunctive and conjunctive expressions on the CUB-200 Birds dataset.

	a OR b						a AND b					
	Known			Unknown			Known			Unknown		
Metrics	MAP	AUC	EER	MAP	AUC	EER	MAP	AUC	EER	MAP	AUC	EER
Chance	39.70	50.00	50.0	40.60	50.00	50.0	4.55	50.0	50.0	4.59	50.0	50.0
Supervised	65.25	74.76	31.58	-	-	-	22.87	78.02	29.69	-	-	-
Independent	58.73	68.39	36.76	60.66	69.28	36.10	17.23	77.22	29.94	19.16	78.00	29.28
Neural Alg. Classifiers	**70.10**	**77.36**	**29.44**	**71.18**	**77.76**	**29.04**	**23.09**	**81.54**	**26.36**	**23.87**	**81.98**	**25.85**

Table 2. Evaluating known/unknown disjunctive and conjunctive expressions on the AwA2 dataset.

	Disjunctive Expressions						Conjunctive Expressions					
	Known Exp.			Unknown Exp.			Known Exp.			Unknown Exp.		
Metrics	MAP	AUC	EER	MAP	AUC	EER	MAP	AUC	EER	MAP	AUC	EER
Chance	53.19	50.0	50.0	53.04	50.0	50.0	18.77	50.0	50.0	21.17	50.0	50.0
Supervised	97.47	97.20	8.13	-	-	-	94.90	98.53	6.00	-	-	-
Independent	97.28	97.12	8.70	97.86	97.58	6.77	93.95	98.13	6.80	93.90	97.87	7.36
Neural Alg. Classifiers	**98.84**	**98.67**	**5.84**	**99.05**	**98.91**	**5.24**	**95.95**	**98.79**	**5.29**	**96.50**	**98.81**	**5.34**

Quantitative Experiments

Baselines:

Chance: Random guess.

Supervised: S… …inate images accordi… …ons.

Cannot generalize

Independent Classifiers:

- P(a AND b) = P(a) x P(b)
- P(a … a) x P(b))
- P(NOT a) = 1 - P(a)

Assume independence

a OR b

a AND b

Table 1. Evaluating known/unknown disjunctive and conjunctive expressions on the CUB-200 Birds dataset.

	Known			Unknown			Known			Unknown		
Metrics	MAP	AUC	EER	MAP	AUC	EER	MAP	AUC	EER	MAP	AUC	EER
Chance	39.70	50.00	50.0	40.60	50.00	50.0	4.55	50.0	50.0	4.59	50.0	50.0
Supervised	65.25	74.76	31.58	-	-	-	22.87	78.02	29.69	-	-	-
Independent	58.73	68.39	36.76	60.66	69.28	36.10	17.23	77.22	29.94	19.16	78.00	29.28
Neural Alg. Classifiers	**70.10**	**77.36**	**29.44**	**71.18**	**77.76**	**29.04**	**23.09**	**81.54**	**26.36**	**23.87**	**81.98**	**25.85**

Table 2. Evaluating known/unknown disjunctive and conjunctive expressions on the AwA2 dataset.

	Disjunctive Expressions						Conjunctive Expressions					
	Known Exp.			Unknown Exp.			Known Exp.			Unknown Exp.		
Metrics	MAP	AUC	EER	MAP	AUC	EER	MAP	AUC	EER	MAP	AUC	EER
Chance	53.19	50.0	50.0	53.04	50.0	50.0	18.77	50.0	50.0	21.17	50.0	50.0
Supervised	97.47	97.20	8.13	-	-	-	94.90	98.53	6.00	-	-	-
Independent	97.28	97.12	8.70	97.86	97.58	6.77	93.95	98.13	6.80	93.90	97.87	7.36
Neural Alg. Classifiers	**98.84**	**98.67**	**5.84**	**99.05**	**98.91**	**5.24**	**95.95**	**98.79**	**5.29**	**96.50**	**98.81**	**5.34**

Our method consistently outperforms the baselines in two attributes datasets (CUB200 and AWA2).

Quantitative Experiments

Baselines:

Chance: Random guess.

Supervised: S... inate images accordi... ns.

Cannot generalize

Independent Classifiers:

- P(a AND b) = P(a) x P(b)
- P(a ... a) x P(b))
- P(NOT a) = 1 - P(a)

Assume independence

a OR b

a AND b

Table 1. Evaluating known/unknown disjunctive and conjunctive expressions on the CUB-200 Birds dataset.

	Known			Unknown			Known			Unknown		
Metrics	MAP	AUC	EER	MAP	AUC	EER	MAP	AUC	EER	MAP	AUC	EER
Chance	39.70	50.00	50.0	40.60	50.00	50.0	4.55	50.0	50.0	4.59	50.0	50.0
Supervised	65.25	74.76	31.58	-	-	-	22.87	78.02	29.69	-	-	-
Independent	58.73	68.39	36.76	60.66	69.28	36.10	17.23	77.22	29.94	19.16	78.00	29.28
Neural Alg. Classifiers	**70.10**	**77.36**	**29.44**	**71.18**	**77.76**	**29.04**	**23.09**	**81.54**	**26.36**	**23.87**	**81.98**	**25.85**

Table 2. Evaluating known/unknown disjunctive and conjunctive expressions on the AwA2 dataset.

	Disjunctive Expressions						Conjunctive Expressions					
	Known Exp.			Unknown Exp.			Known Exp.			Unknown Exp.		
Metrics	MAP	AUC	EER	MAP	AUC	EER	MAP	AUC	EER	MAP	AUC	EER
Chance	53.19	50.0	50.0	53.04	50.0	50.0	18.77	50.0	50.0	21.17	50.0	50.0
Supervised	97.47	97.20	8.13	-	-	-	94.90	98.53	6.00	-	-	-
Independent	97.28	97.12	8.70	97.86	97.58	6.77	93.95	98.13	6.80	93.90	97.87	7.36
Neural Alg. Classifiers	**98.84**	**98.67**	**5.84**	**99.05**	**98.91**	**5.24**	**95.95**	**98.79**	**5.29**	**96.50**	**98.81**	**5.34**

It works as well as the supervised approach with known expressions.

Quantitative Experiments

Baselines:

Chance: Random guess.

Supervised: S... inate images accordi... ns.

Cannot generalize

Independent Classifiers:

- P(a AND b) = P(a) x P(b)
- P(a ... a) x P(b))
- P(NOT a) = 1 - P(a)

Assume independence

a OR b | a AND b

Table 1. Evaluating known/unknown disjunctive and conjunctive expressions on the CUB-200 Birds dataset.

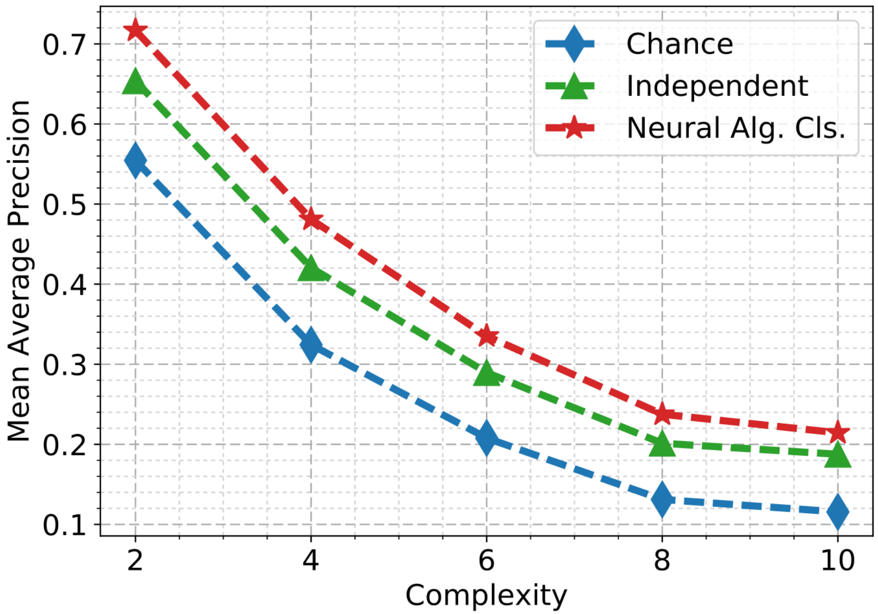
	Known			Unknown			Known			Unknown		
Metrics	MAP	AUC	EER	MAP	AUC	EER	MAP	AUC	EER	MAP	AUC	EER
Chance	39.70	50.00	50.0	40.60	50.00	50.0	4.55	50.0	50.0	4.59	50.0	50.0
Supervised	65.25	74.76	31.58	-	-	-	22.87	78.02	29.69	-	-	-
Independent	58.73	68.39	36.76	60.66	69.28	36.10	17.23	77.22	29.94	19.16	78.00	29.28
Neural Alg. Classifiers	**70.10**	**77.36**	**29.44**	**71.18**	**77.76**	**29.04**	**23.09**	**81.54**	**26.36**	**23.87**	**81.98**	**25.85**

Table 2. Evaluating known/unknown disjunctive and conjunctive expressions on the AwA2 dataset.

	Disjunctive Expressions						Conjunctive Expressions					
	Known Exp.			Unknown Exp.			Known Exp.			Unknown Exp.		
Metrics	MAP	AUC	EER	MAP	AUC	EER	MAP	AUC	EER	MAP	AUC	EER
Chance	53.19	50.0	50.0	53.04	50.0	50.0	18.77	50.0	50.0	21.17	50.0	50.0
Supervised	97.47	97.20	8.13	-	-	-	94.90	98.53	6.00	-	-	-
Independent	97.28	97.12	8.70	97.86	97.58	6.77	93.95	98.13	6.80	93.90	97.87	7.36
Neural Alg. Classifiers	**98.84**	**98.67**	**5.84**	**99.05**	**98.91**	**5.24**	**95.95**	**98.79**	**5.29**	**96.50**	**98.81**	**5.34**

Quantitative Experiments

Complex Unknown Expressions: $(p_1 \vee q_1) \wedge (p_2 \vee q_2) \wedge \ldots$ where p and q are visual primitives which may appear negated and c (complexity) is the number of simple terms in those expressions.



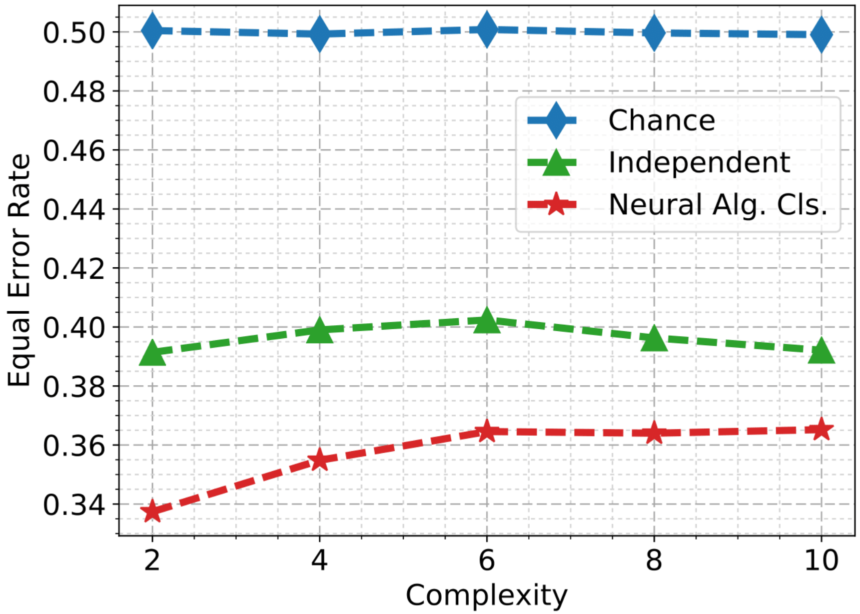


Figure: Performance of the proposed method and baselines on classifying images of **CUB-200** dataset according to unknown expressions of different complexity in conjunctive normal form.

Quantitative Experiments

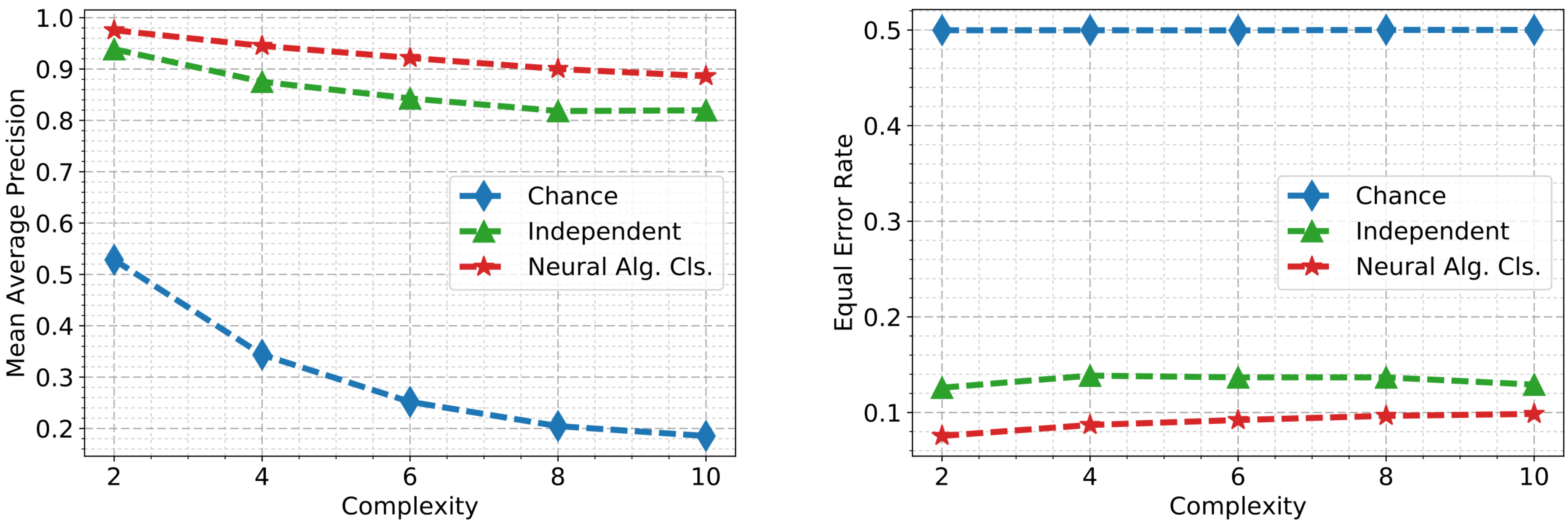


Figure: Performance of the proposed method and baselines on classifying images of **AWA2** dataset according to unknown expressions of different complexity in conjunctive normal form.

Qualitative Experiments

Qualitative Experiments

Birds with crown and breast of **the same** color (e.g., blue, yellow, or red.)

Qualitative Experiments

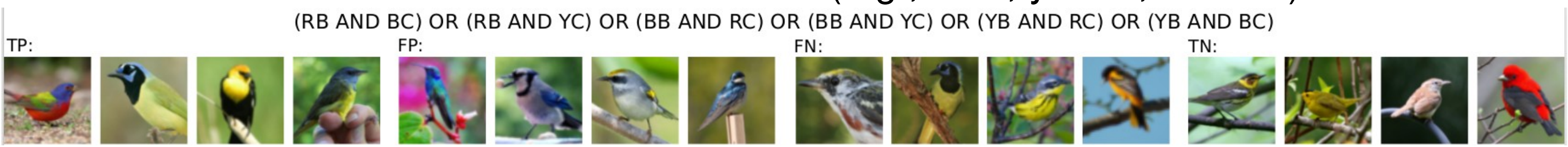
Birds with crown and breast of **the same** color (e.g., blue, yellow, or red.)

Birds with crown and breast of **different** color (e.g., blue, yellow, or red.)

Qualitative Experiments

Birds with crown and breast of **the same** color (e.g., blue, yellow, or red.)

Birds with crown and breast of **different** color (e.g., blue, yellow, or red.)



Qualitative Experiments

Birds with crown and breast of **the same** color (e.g., blue, yellow, or red.)

Birds with crown and breast of **different** color (e.g., blue, yellow, or red.)

Qualitative Experiments

Birds with crown and breast of **the same** color (e.g., blue, yellow, or red.)

Birds with crown and breast of **different** color (e.g., blue, yellow, or red.)

Big and fast animals that are not hunters:

(B AND F) AND (NOT H) = (NOT (S OR SL)) AND (NOT H)

Qualitative Experiments

Birds with crown and breast of **the same** color (e.g., blue, yellow, or red.)

Birds with crown and breast of **different** color (e.g., blue, yellow, or red.)

Big and fast animals that are not hunters:

(B AND F) AND (NOT H) = (NOT (S OR SL)) AND (NOT H)

Qualitative Experiments

Birds with crown and breast of **the same** color (e.g., blue, yellow, or red.)

Birds with crown and breast of **different** color (e.g., blue, yellow, or red.)

Big and fast animals that are not hunters:

(B AND F) AND (NOT H) = (NOT (S OR SL)) AND (NOT H)

Neural Algebra of Classifiers

A deep learning framework for composition of classifiers

Rodrigo Santa Cruz[1], Basura Fernando[1], Anoop Cherian[1,2], and Stephen Gould[1]
[1]Australian Centre for Robotic Vision, Australian National University, Canberra, Australia
[2]Mitsubishi Electric Research Labs, 201 Broadway, Cambridge, MA