Decoding the Brain to Help Build Machines

Jeffrey Mark Siskind, qobi@purdue.edu

PURDUE UNIVERSITY

University of Washington, Wednesday 11 May 2016

Outline

Outline

1. Verbs, Arguments, and Predication in the Human Brain
   - Experiment 1: hollywood2-text-speech
   - Experiment 2: compositionality-noninterleaved
   - Experiment 3: predication
2. Sentence Directed Video Object Codetection
3. Driving Under the Influence (of Language)
   - Grounding Language Semantics in Robotics
   - Object Codetection from Mobile Robot Video
4. Playing Checkers from English

Joint work

Andrei Barbu	Daniel Paul Barrett	Charles Roger Bradley
Scott Alan Bronikowski	Zachary Burchill	Jason J. Corso
Christiane D. Fellbaum	Catherine Hanson	Stephen José Hanson
Sébastien Hélie	Evguenia Malaia	Barak A. Pearlmutter
N. Siddharth	Thomas Michael Talavage	Ronnie B. Wilbur
Caiming Xiong	Haonan Yu	

Does the brain have a nondistributed representation?

Does the brain have a nondistributed representation?

Does the brain have a nondistributed representation?

Jonathan is distributed over the retina

Does the brain have a nondistributed representation?

Jonathan is distributed over the retina
pick up is distributed over the retina over time

Does the brain have a nondistributed representation?

Jonathan is distributed over the retina
pick up is distributed over the retina over time
chair is distributed over the retina

Does the brain have a nondistributed representation?

Jonathan is distributed over the retina
pick up is distributed over the retina over time
chair is distributed over the retina
some retinal points code both *Jonathan* and *pick up*

Does the brain have a nondistributed representation?

Jonathan is distributed over the retina
pick up is distributed over the retina over time
chair is distributed over the retina
some retinal points code both *Jonathan* and *pick up*
some retinal points code both *pick up* and *chair*

Does the brain have a nondistributed representation?

Jonathan is distributed over motor neurons
pick up is distributed over motor neurons over time
chair is distributed over motor neurons
some motor neurons code both *Jonathan* and *pick up*
some motor neurons code both *pick up* and *chair*

What underlies a symbolic representation?

What underlies a symbolic representation?

- categorical judgments

What underlies a symbolic representation?

- categorical judgments
  John or *Mary*

What underlies a symbolic representation?

- categorical judgments
  John or *Mary*, not 80% *John* and 20% *Mary*

What underlies a symbolic representation?

- categorical judgments
  John or *Mary*, not 80% *John* and 20% *Mary*
- modality neutrality

What underlies a symbolic representation?

- categorical judgments
  John or *Mary*, not 80% *John* and 20% *Mary*
- modality neutrality
  same brain activity evoked for seeing John, reading *John*, and hearing John's name

What underlies a symbolic representation?

- categorical judgments
  John or *Mary*, not 80% *John* and 20% *Mary*
- modality neutrality
  same brain activity evoked for seeing John, reading *John*, and hearing John's name
- factored

What underlies a symbolic representation?

- categorical judgments
  John or *Mary*, not 80% *John* and 20% *Mary*
- modality neutrality
  same brain activity evoked for seeing John, reading *John*, and hearing John's name
- factored

$$\left\{\begin{matrix} John \\ Mary \end{matrix}\right\} \times \left\{\begin{matrix} walks \\ talks \end{matrix}\right\}$$

What underlies a symbolic representation?

- categorical judgments
  John or *Mary*, not 80% *John* and 20% *Mary*
- modality neutrality
  same brain activity evoked for seeing John, reading *John*, and hearing John's name
- factored

$$\left\{\begin{array}{l}\textit{John}\\ \textit{Mary}\end{array}\right\}\times\left\{\begin{array}{l}\textit{walks}\\ \textit{talks}\end{array}\right\}\quad\text{not}\left\{\begin{array}{l}\textit{John-walks}\\ \textit{John-talks}\\ \textit{Mary-walks}\\ \textit{Mary-talks}\end{array}\right\}$$

What underlies a symbolic representation?

- categorical judgments
  John or *Mary*, not *80% John* and *20% Mary*
- modality neutrality
  same brain activity evoked for seeing John, reading *John*, and hearing John's name
- factored

$$\left\{\begin{array}{l}\textit{John}\\ \textit{Mary}\end{array}\right\}\times\left\{\begin{array}{l}\textit{walks}\\ \textit{talks}\end{array}\right\}\quad\text{not}\left\{\begin{array}{l}\textit{John-walks}\\ \textit{John-talks}\\ \textit{Mary-walks}\\ \textit{Mary-talks}\end{array}\right\}$$

- predication

What underlies a symbolic representation?

- categorical judgments
  John or *Mary*, not 80% *John* and 20% *Mary*
- modality neutrality
  same brain activity evoked for seeing John, reading *John*, and hearing John's name
- factored

$$\left\{\begin{array}{l} John \\ Mary \end{array}\right\} \times \left\{\begin{array}{l} walks \\ talks \end{array}\right\} \quad \text{not} \left\{\begin{array}{l} John\text{-}walks \\ John\text{-}talks \\ Mary\text{-}walks \\ Mary\text{-}talks \end{array}\right\}$$

- predication

$$walk(John) \wedge talk(Mary)$$
$$walk(Mary) \wedge talk(John)$$

What underlies a symbolic representation?

- categorical judgments
  John or *Mary*, not 80% *John* and 20% *Mary*
- modality neutrality
  same brain activity evoked for seeing John, reading *John*, and hearing John's name
- factored

$$\left\{\begin{matrix} John \\ Mary \end{matrix}\right\} \times \left\{\begin{matrix} walks \\ talks \end{matrix}\right\} \quad \text{not} \left\{\begin{matrix} John\text{-}walks \\ John\text{-}talks \\ Mary\text{-}walks \\ Mary\text{-}talks \end{matrix}\right\}$$

- predication

$$\begin{matrix} walk(John) \wedge talk(Mary) \\ walk(Mary) \wedge talk(John) \end{matrix} \quad \text{not } \{John, Mary, walk, talk\}$$

Outline

1. Verbs, Arguments, and Predication in the Human Brain
   - Experiment 1: hollywood2-text-speech
   - Experiment 2: compositionality-noninterleaved
   - Experiment 3: predication
2. Sentence Directed Video Object Codetection
3. Driving Under the Influence (of Language)
   - Grounding Language Semantics in Robotics
   - Object Codetection from Mobile Robot Video
4. Playing Checkers from English

What underlies a symbolic representation?

- categorical judgments
  John or *Mary*, not 80% *John* and 20% *Mary*
- modality neutrality
  same brain activity evoked for seeing John, reading *John*, and hearing John's name

One Slide Tells it All

video

One Slide Tells it All

 → 

video computer

One Slide Tells it All

 $\rightarrow$  $\rightarrow$ $\approx 40\%$

video computer accuracy

One Slide Tells it All

video

$\longrightarrow$

computer

$\longrightarrow$

$\approx 40\%$

accuracy

video

One Slide Tells it All

 $\rightarrow$  $\rightarrow$ $\approx 40\%$

video — computer — accuracy

 $\rightarrow$ 

video — subject

One Slide Tells it All

 →  → $\approx$40%

video computer accuracy

 →  → 

video subject fMRI

One Slide Tells it All

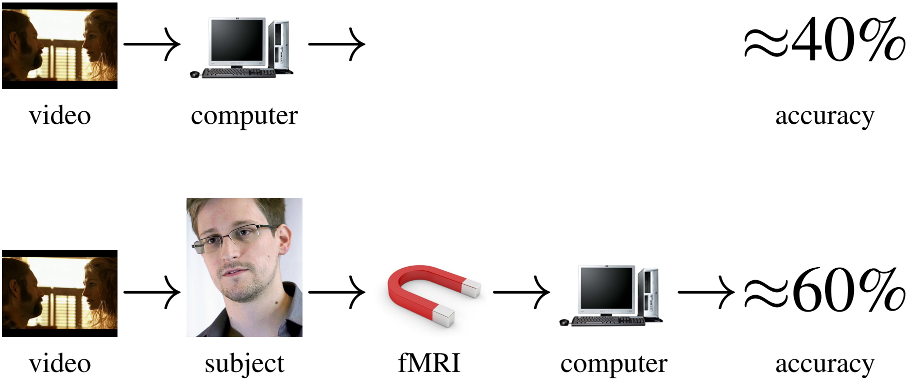
 →  → $\approx 40\%$

video — computer — accuracy

 →  →  → 

video — subject — fMRI — computer

One Slide Tells it All



Stimuli

HOLLYWOOD-2 Marszałek et al. (2009)

Stimuli

HOLLYWOOD-2 Marszałek et al. (2009)

Stimuli

$$\Longrightarrow \left\{ \begin{array}{l} \mathit{AnswerPhone}, \\ \mathit{DriveCar}, \\ \mathit{Eat}, \\ \mathit{FightPerson}, \\ \mathit{GetOutCar}, \\ \mathit{HandShake}, \\ \mathit{HugPerson}, \\ \mathit{Kiss}, \\ \mathit{Run}, \\ \mathit{SitDown}, \\ \mathit{SitUp}, \\ \mathit{StandUp} \end{array} \right.$$

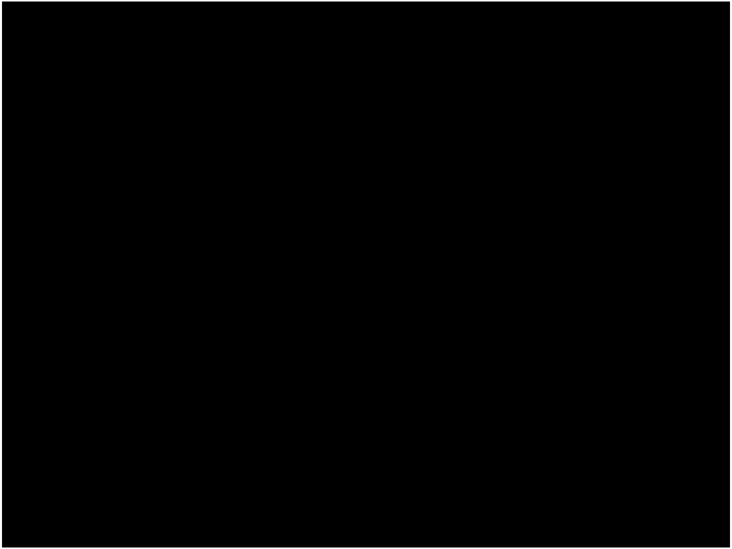
HOLLYWOOD-2 Marszałek et al. (2009)

Stimuli

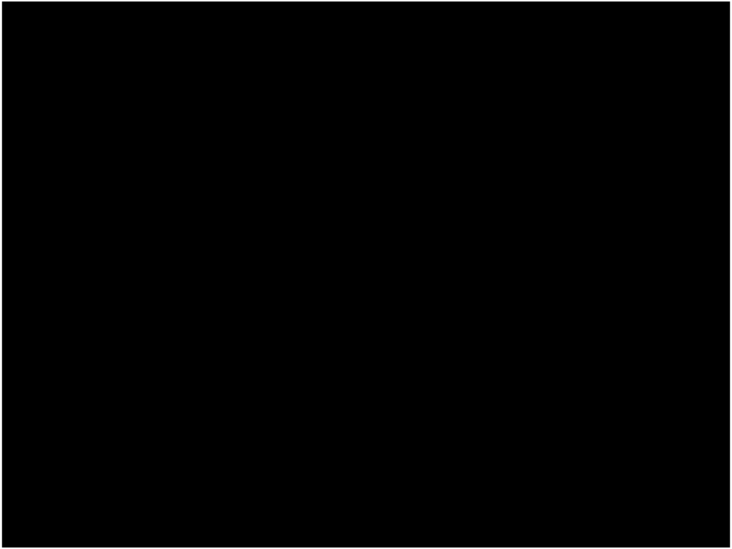
$$\Longrightarrow \left\{ \begin{array}{l} \mathit{AnswerPhone}, \\ \mathit{DriveCar}, \\ \mathit{Eat}, \\ \mathit{FightPerson}, \\ \mathit{GetOutCar}, \\ \mathit{HandShake}, \\ \mathit{HugPerson}, \\ \mathit{Kiss}, \\ \mathit{Run}, \\ \mathit{SitDown}, \\ \mathit{SitUp}, \\ \mathit{StandUp} \end{array} \right.$$

HOLLYWOOD-2 Marszałek et al. (2009)

Stimuli

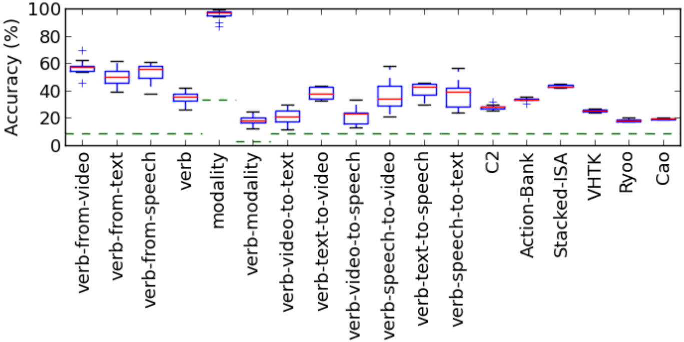


Stimuli



Classification Accuracies

fMRI from-video	57.0%
fMRI from-text	50.0%
fMRI from-speech	52.7%
fMRI video-to-text	21.1%
fMRI text-to-video	38.0%
fMRI video-to-speech	21.6%
fMRI speech-to-video	36.0%
fMRI text-to-speech	40.0%
fMRI speech-to-text	37.1%
C2	28.0%
Action Bank	33.6%
Stacked ISA	43.3%
VHTK	25.3%
Ryoo et al.	18.3%
Cao et al.	19.2%
chance	8.4%



Confusion Matrices

fMRI verb-from-video

	AnswerPhone	DriveCar	Eat	FightPerson	GetOutCar	HandShake	HugPerson	Kiss	Run	SitDown	SitUp	StandUp
AnswerPhone	159	13	32	3	10	26	17	10	5	19	13	13
DriveCar	8	189	16	4	14	21	8	7	22	9	7	15
Eat	27	20	176	5	5	18	19	20	3	7	10	10
FightPerson	1	6	2	237	8	5	14	11	9	8	8	11
GetOutCar	6	15	4	10	196	5	4	3	23	14	12	28
HandShake	22	5	20	7	5	193	17	14	3	12	5	17
HugPerson	18	16	15	11	10	16	155	44	4	8	11	12
Kiss	13	9	12	11	3	21	38	178	2	13	12	8
Run	3	28	1	7	15	2	7	2	225	12	3	15
SitDown	13	9	5	7	17	13	6	7	6	155	29	53
SitUp	12	14	7	13	9	5	9	8	4	21	191	27
StandUp	6	13	1	12	25	24	8	9	14	52	21	135

Confusion Matrices

fMRI verb-from-text

	AnswerPhone	DriveCar	Eat	FightPerson	GetOutCar	HandShake	HugPerson	Kiss	Run	SitDown	SitUp	StandUp
AnswerPhone	151	16	13	25	10	25	14	10	13	17	9	17
DriveCar	26	149	10	22	9	15	11	10	15	16	14	23
Eat	16	12	171	7	8	9	10	17	30	12	13	15
FightPerson	15	18	10	161	18	23	12	11	8	14	8	22
GetOutCar	19	20	15	15	157	15	18	7	12	14	7	21
HandShake	18	18	8	22	12	157	14	10	13	11	14	23
HugPerson	20	16	14	19	13	17	168	12	10	13	8	10
Kiss	2	14	27	8	5	12	7	177	20	19	13	16
Run	13	10	24	9	13	8	8	19	181	11	12	12
SitDown	11	12	13	15	13	18	11	20	15	151	21	20
SitUp	16	15	10	9	15	15	10	13	13	30	159	15
StandUp	16	25	8	22	15	20	8	18	16	17	15	140

Confusion Matrices

fMRI verb-from-speech

	AnswerPhone	DriveCar	Eat	FightPerson	GetOutCar	HandShake	HugPerson	Kiss	Run	SitDown	SitUp	StandUp
AnswerPhone	161	15	11	14	15	14	24	12	4	17	14	19
DriveCar	18	174	5	11	14	13	18	8	9	20	15	15
Eat	6	11	177	10	5	17	8	27	21	9	13	16
FightPerson	20	18	12	157	19	11	20	11	10	12	14	16
GetOutCar	22	18	4	15	189	15	11	3	13	10	9	11
HandShake	14	13	11	16	17	157	28	17	12	15	13	7
HugPerson	12	14	6	20	16	24	165	14	16	6	10	17
Kiss	6	5	24	1	2	19	14	190	17	16	10	16
Run	9	8	15	11	6	10	16	24	176	13	17	15
SitDown	14	31	15	12	14	17	11	10	16	165	7	8
SitUp	12	15	22	16	11	11	11	12	17	27	154	12
StandUp	16	18	14	18	15	12	11	20	14	11	10	161

Confusion Matrices

C2

	AnswerPhone	DriveCar	Eat	FightPerson	GetOutCar	HandShake	HugPerson	Kiss	Run	SitDown	SitUp	StandUp
AnswerPhone	49	7	99	2	13	29	8	30	20	29	15	19
DriveCar	12	75	30	28	41	17	13	17	61	7	16	3
Eat	25	15	169	1	18	13	19	12	11	22	10	5
FightPerson	3	21	7	186	14	2	10	8	17	36	9	7
GetOutCar	8	44	34	23	67	25	8	9	49	14	21	18
HandShake	16	27	30	19	7	82	17	41	19	28	12	22
HugPerson	20	29	24	43	11	27	28	27	27	29	24	31
Kiss	35	30	11	16	21	38	23	60	10	50	16	10
Run	11	4	15	86	29	17	6	13	87	34	5	13
SitDown	20	3	21	16	10	25	4	21	15	157	10	18
SitUp	24	21	46	48	24	12	19	16	13	26	32	39
StandUp	22	3	32	17	30	21	25	18	10	26	32	84

Confusion Matrices

Action Bank

	AnswerPhone	DriveCar	Eat	FightPerson	GetOutCar	HandShake	HugPerson	Kiss	Run	SitDown	SitUp	StandUp
AnswerPhone	97	19	52	10	12	25	31	11	0	39	24	0
DriveCar	1	142	11	21	31	42	10	12	39	10	0	1
Eat	42	27	122	10	14	32	14	31	9	13	5	1
FightPerson	0	28	0	193	16	10	15	4	28	1	11	14
GetOutCar	25	29	27	43	104	9	1	0	15	0	39	28
HandShake	30	27	13	6	12	145	27	22	20	11	0	7
HugPerson	35	11	7	57	0	38	55	53	0	26	23	15
Kiss	44	0	23	34	21	25	26	89	4	31	3	20
Run	21	36	6	89	30	11	7	0	91	18	0	11
SitDown	27	16	58	14	0	10	23	34	19	91	1	27
SitUp	30	10	30	16	21	10	11	40	2	0	73	77
StandUp	13	2	11	20	41	16	11	1	21	21	74	89

Confusion Matrices

Stacked ISA

	AnswerPhone	DriveCar	Eat	FightPerson	GetOutCar	HandShake	HugPerson	Kiss	Run	SitDown	SitUp	StandUp
AnswerPhone	65	0	72	3	3	60	4	29	0	49	21	14
DriveCar	0	258	10	15	12	0	0	0	22	0	0	3
Eat	17	24	174	0	27	16	17	7	4	20	14	0
FightPerson	8	6	8	187	11	12	10	0	36	19	13	10
GetOutCar	2	39	22	29	147	5	0	5	22	10	4	35
HandShake	55	11	12	0	17	133	4	42	11	12	14	9
HugPerson	31	1	10	18	7	39	42	73	9	21	52	17
Kiss	20	2	10	6	7	53	37	122	0	27	34	2
Run	8	45	3	71	14	0	3	0	132	21	10	13
SitDown	29	0	7	1	3	26	6	36	12	143	15	42
SitUp	40	2	16	13	11	25	29	21	7	11	82	63
StandUp	11	0	0	2	18	6	40	4	9	25	26	179

Confusion Matrices

VHTK

	AnswerPhone	DriveCar	Eat	FightPerson	GetOutCar	HandShake	HugPerson	Kiss	Run	SitDown	SitUp	StandUp
AnswerPhone	36	1	179	14	2	7	16	43	2	2	9	9
DriveCar	0	132	18	73	5	33	0	32	24	0	3	0
Eat	21	1	207	10	4	28	8	22	0	10	5	4
FightPerson	0	17	15	143	5	5	16	31	30	20	38	0
GetOutCar	17	7	53	63	5	34	0	42	31	12	28	28
HandShake	26	10	79	1	11	60	12	66	8	15	20	12
HugPerson	27	25	69	61	7	10	17	60	13	9	22	0
Kiss	32	24	42	32	5	36	13	103	8	0	25	0
Run	7	31	43	94	2	8	12	7	64	10	30	12
SitDown	9	2	81	44	5	19	0	32	10	100	12	6
SitUp	5	8	48	60	2	43	4	75	19	1	29	26
StandUp	9	10	59	50	5	15	4	11	47	10	24	76

Confusion Matrices

Ryoo

	AnswerPhone	DriveCar	Eat	FightPerson	GetOutCar	HandShake	HugPerson	Kiss	Run	SitDown	SitUp	StandUp
AnswerPhone	5	2	87	41	16	29	1	113	0	7	11	8
DriveCar	2	11	73	54	17	17	0	44	22	2	0	78
Eat	0	14	189	25	11	1	0	65	3	7	0	5
FightPerson	10	19	3	186	10	18	2	11	10	0	3	48
GetOutCar	0	9	55	58	42	19	0	38	10	10	31	48
HandShake	10	30	10	16	0	26	4	191	0	0	0	33
HugPerson	3	0	40	128	3	20	0	65	7	0	13	41
Kiss	2	13	46	44	19	30	2	112	1	4	4	43
Run	12	11	50	129	4	9	0	66	11	12	5	11
SitDown	0	0	43	87	24	14	0	99	16	9	3	25
SitUp	0	16	56	39	30	8	3	63	16	0	29	60
StandUp	10	11	20	77	15	1	12	47	1	0	43	83

Confusion Matrices

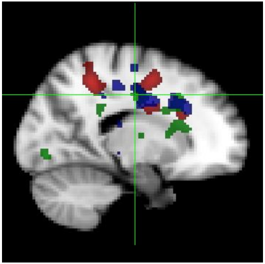
Cao

	AnswerPhone	DriveCar	Eat	FightPerson	GetOutCar	HandShake	HugPerson	Kiss	Run	SitDown	SitUp	StandUp
AnswerPhone	21	2	62	60	35	31	18	64	7	8	12	0
DriveCar	26	2	33	66	51	13	12	39	29	9	39	1
Eat	16	21	172	20	3	8	4	41	0	20	8	7
FightPerson	6	1	0	221	37	9	3	10	19	0	4	10
GetOutCar	10	7	38	63	100	18	17	25	0	3	22	17
HandShake	27	10	11	27	29	41	27	89	2	24	23	10
HugPerson	17	0	21	168	11	21	17	26	9	21	0	9
Kiss	33	0	20	64	32	44	11	55	5	35	5	16
Run	32	8	25	120	17	9	1	43	1	42	10	12
SitDown	15	0	30	74	10	24	37	44	0	76	10	0
SitUp	0	11	42	87	78	23	6	31	1	8	9	24
StandUp	11	15	18	126	48	22	1	20	8	9	16	26

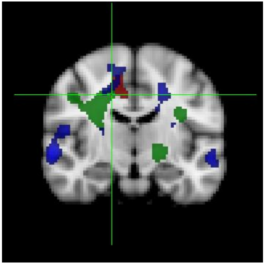
Hot Off The Press

	SVM	NN
fMRI from-video	57.0%	60.6%
fMRI from-text	50.0%	53.1%
fMRI from-speech	52.7%	56.0%
fMRI from-all	34.8%	51.8%
fMRI video-to-text	21.1%	26.6%
fMRI text-to-video	38.0%	44.6%
fMRI video-to-speech	21.6%	28.0%
fMRI speech-to-video	36.0%	49.4%
fMRI text-to-speech	40.0%	44.0%
fMRI speech-to-text	37.1%	45.6%

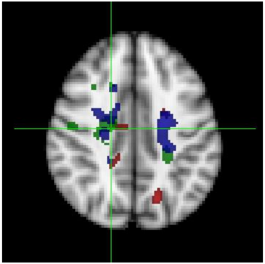
Brain Regions for Video, Text, and Speech



slice 36 (X = 18.00)



slice 56 (Y = -14.00)



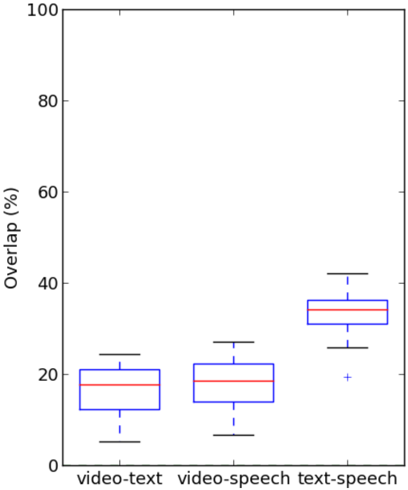
slice 56 (Z = 40.00)

- video
- text
- speech

Subject 05, MNI_152

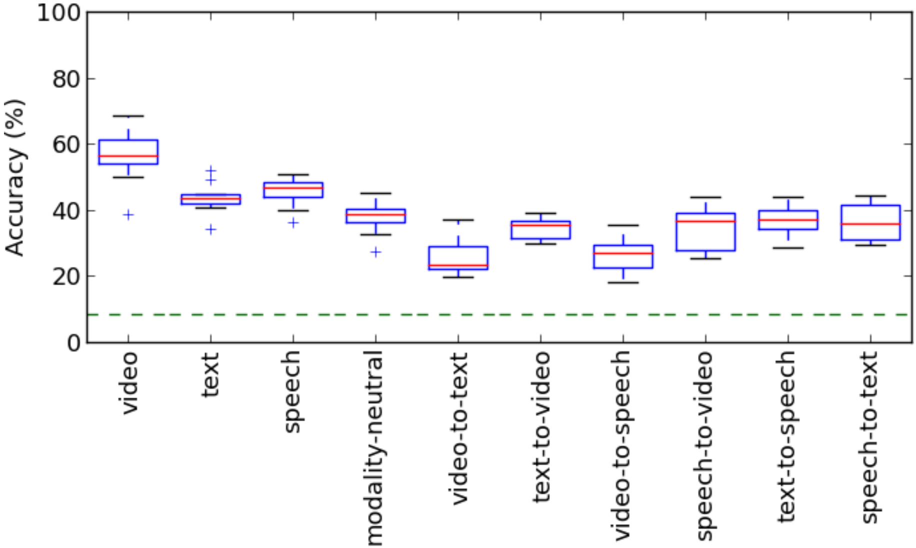
Quantifying Overlap Between Video, Text, and Speech

$$\frac{|X \cap Y|}{|X \cup Y|}$$

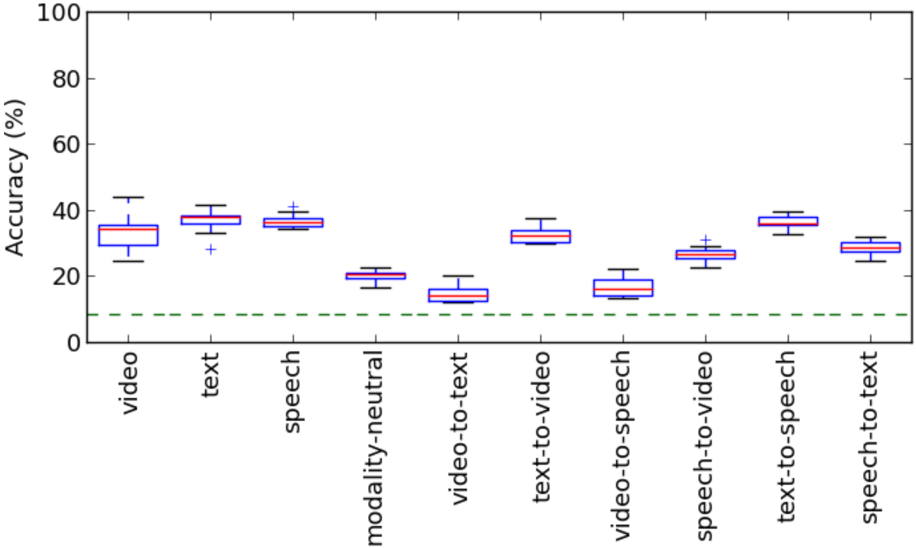


In 4000 voxels (≈10% of brain volume) with highest magnitude of SVM weight.

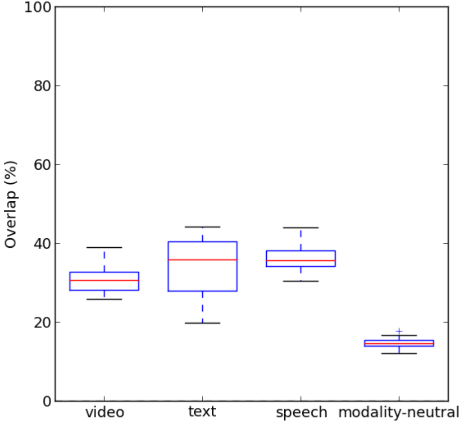
Within-Subject Classification



Cross-Subject Classification



Cross-Subject Overlap



Outline

What underlies a symbolic representation?

- factored

$$\left\{\begin{array}{l}\textit{John}\\\textit{Mary}\end{array}\right\}\times\left\{\begin{array}{l}\textit{walks}\\\textit{talks}\end{array}\right\}\quad\text{not}\quad\left\{\begin{array}{l}\textit{John-walks}\\\textit{John-talks}\\\textit{Mary-walks}\\\textit{Mary-talks}\end{array}\right\}$$

$$\left\{\begin{array}{l} \textit{Dan} \\ \textit{Haonan} \\ \textit{Jeff} \\ \textit{Scott} \end{array}\right\} \times \left\{\begin{array}{l} \textit{carried} \\ \textit{folded} \\ \textit{left} \end{array}\right\} \times \left\{\begin{array}{l} \textit{the chair} \\ \textit{the paper} \\ \textit{the shirt} \end{array}\right\} \times \left\{\begin{array}{l} [\textit{on the}]\textit{left}[\textit{ward}] \\ [\textit{on the}]\textit{right}[\textit{ward}] \end{array}\right\}$$

$$\begin{Bmatrix} \textit{Dan} \\ \textit{Haonan} \\ \textit{Jeff} \\ \textit{Scott} \end{Bmatrix}$$

actor

$$\left\{\begin{array}{l} carried \\ folded \\ left \end{array}\right\}$$

verb

$$\left\{\begin{array}{l} \textit{the chair} \\ \textit{the paper} \\ \textit{the shirt} \end{array}\right\}$$

object

$$\left\{ \begin{array}{r} \textit{left}[\textit{ward}] \\ \textit{right}[\textit{ward}] \end{array} \right\}$$

direction

$$\left\{\begin{array}{l} [\textit{on the}]\textit{left} \\ [\textit{on the}]\textit{right} \end{array}\right\}$$

location

$$\left\{\begin{array}{l} \textit{Dan} \\ \textit{Haonan} \\ \textit{Jeff} \\ \textit{Scott} \end{array}\right\} \times \left\{\begin{array}{l} \textit{carried} \\ \textit{folded} \\ \textit{left} \end{array}\right\}$$

actor-verb

$$\left\{\begin{array}{l} \textit{Dan} \\ \textit{Haonan} \\ \textit{Jeff} \\ \textit{Scott} \end{array}\right\} \times \left\{\begin{array}{l} \textit{the chair} \\ \textit{the paper} \\ \textit{the shirt} \end{array}\right\}$$

actor-object

$$\begin{Bmatrix} \mathit{Dan} \\ \mathit{Haonan} \\ \mathit{Jeff} \\ \mathit{Scott} \end{Bmatrix} \times \begin{Bmatrix} \mathit{left}[\mathit{ward}] \\ \mathit{right}[\mathit{ward}] \end{Bmatrix}$$

actor-direction

$$\left\{\begin{array}{l} \textit{Dan} \\ \textit{Haonan} \\ \textit{Jeff} \\ \textit{Scott} \end{array}\right\} \times \left\{\begin{array}{l} [\textit{on the}]\,\textit{left} \\ [\textit{on the}]\,\textit{right} \end{array}\right\}$$

actor-location

$$\left\{\begin{array}{l} \textit{carried} \\ \textit{folded} \\ \textit{left} \end{array}\right\} \times \left\{\begin{array}{l} \textit{the chair} \\ \textit{the paper} \\ \textit{the shirt} \end{array}\right\}$$

verb-object

$$\left\{\begin{matrix} \textit{carried} \\ \\ \textit{left} \end{matrix}\right\} \times \left\{\begin{matrix} \textit{left}[\textit{ward}] \\ \textit{right}[\textit{ward}] \end{matrix}\right\}$$

verb-direction

$$\left\{\begin{array}{l}\textit{the chair}\\ \textit{the paper}\\ \textit{the shirt}\end{array}\right\} \times \left\{\begin{array}{r}\textit{left}[\textit{ward}]\\ \textit{right}[\textit{ward}]\end{array}\right\}$$

object-direction

$$\left\{\begin{array}{l} \textit{the chair} \\ \textit{the paper} \\ \textit{the shirt} \end{array}\right\} \times \left\{\begin{array}{l} [\textit{on the}]\,\textit{left} \\ [\textit{on the}]\,\textit{right} \end{array}\right\}$$

object-location

$$\left\{\begin{array}{l} \textit{Dan} \\ \textit{Haonan} \\ \textit{Jeff} \\ \textit{Scott} \end{array}\right\} \times \left\{\begin{array}{l} \textit{carried} \\ \textit{folded} \\ \textit{left} \end{array}\right\} \times \left\{\begin{array}{l} \textit{the chair} \\ \textit{the paper} \\ \textit{the shirt} \end{array}\right\}$$

actor-verb-object

$$\left\{\begin{array}{l} Dan \\ Haonan \\ Jeff \\ Scott \end{array}\right\} \times \left\{\begin{array}{l} carried \\ \\ left \end{array}\right\} \qquad \times \left\{\begin{array}{r} left[ward] \\ right[ward] \end{array}\right\}$$

actor-verb-direction

$$\begin{Bmatrix} \textit{Dan} \\ \textit{Haonan} \\ \textit{Jeff} \\ \textit{Scott} \end{Bmatrix} \times \begin{Bmatrix} \textit{the chair} \\ \textit{the paper} \\ \textit{the shirt} \end{Bmatrix} \times \begin{Bmatrix} \textit{left}[\textit{ward}] \\ \textit{right}[\textit{ward}] \end{Bmatrix}$$

actor-object-direction

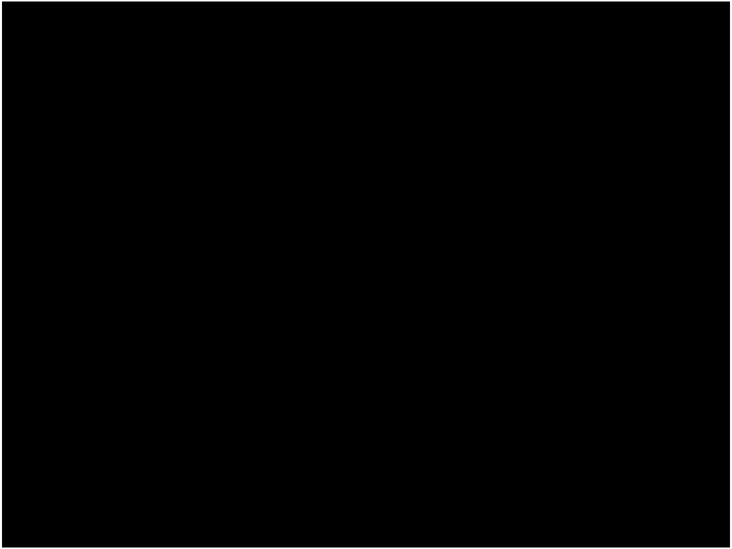
$$\left\{\begin{matrix} \textit{carried} \\ \\ \textit{left} \end{matrix}\right\} \times \left\{\begin{matrix} \textit{the chair} \\ \textit{the paper} \\ \textit{the shirt} \end{matrix}\right\} \times \left\{\begin{matrix} \textit{left[ward]} \\ \textit{right[ward]} \end{matrix}\right\}$$

verb-object-direction

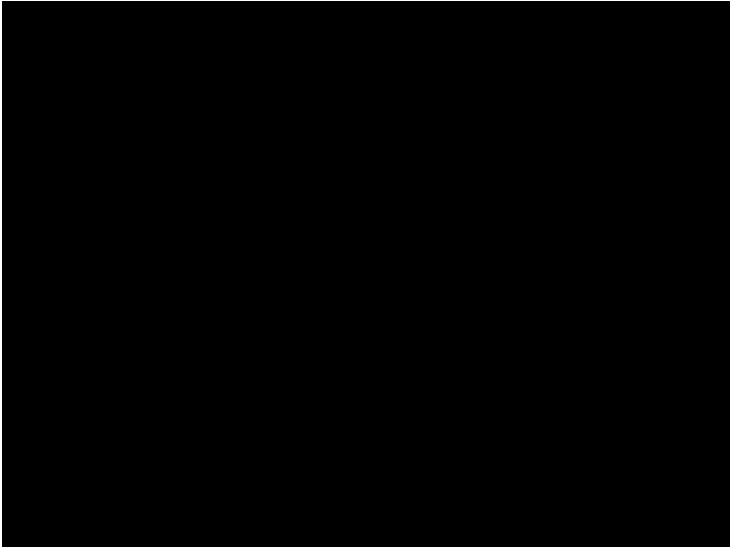
$$\left\{\begin{array}{l} \textit{Dan} \\ \textit{Haonan} \\ \textit{Jeff} \\ \textit{Scott} \end{array}\right\} \times \left\{\begin{array}{l} \textit{carried} \\ \textit{folded} \\ \textit{left} \end{array}\right\} \times \left\{\begin{array}{l} \textit{the chair} \\ \textit{the paper} \\ \textit{the shirt} \end{array}\right\} \times \left\{\begin{array}{l} [\textit{on the}]\textit{left}[\textit{ward}] \\ [\textit{on the}]\textit{right}[\textit{ward}] \end{array}\right\}$$

sentence

Stimuli



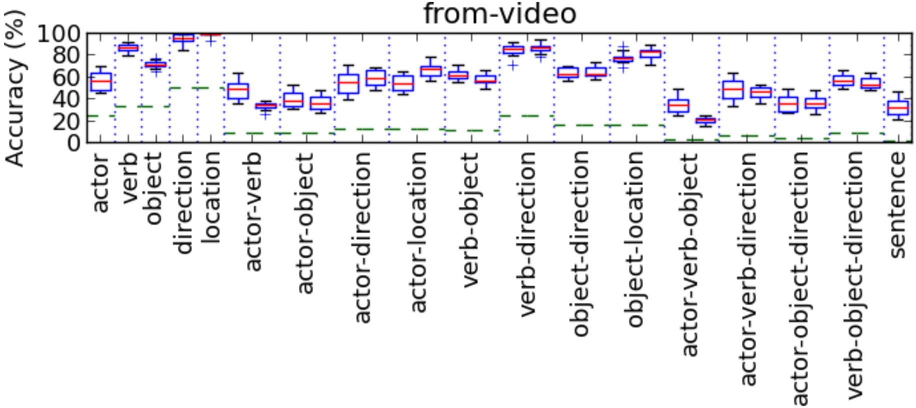
Stimuli



independent Train on individual constituents, test on constituent pairs and triples.

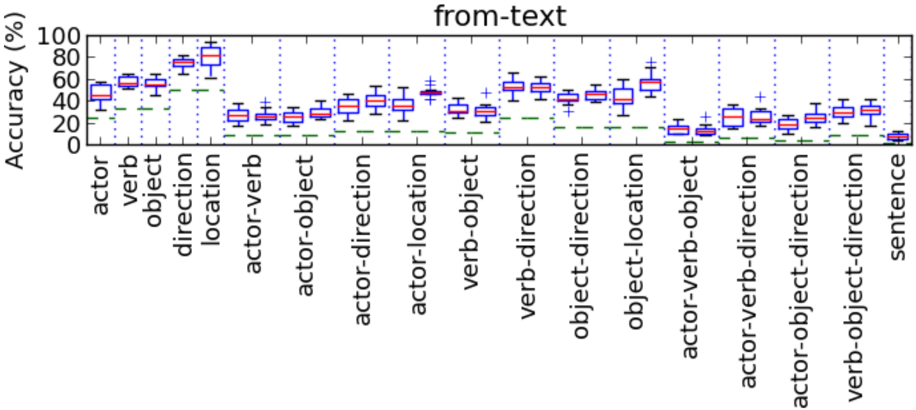
joint Train and test on constituent pairs and triples.

Classification Accuracies



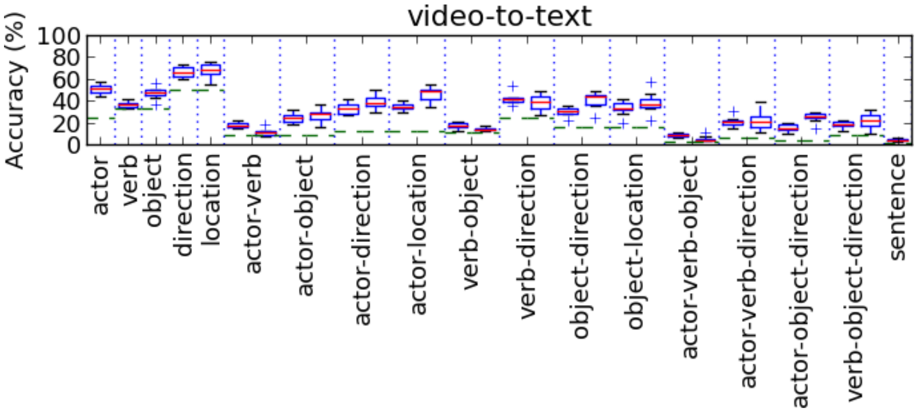
independent on left, joint on right of each pair and triple

Classification Accuracies



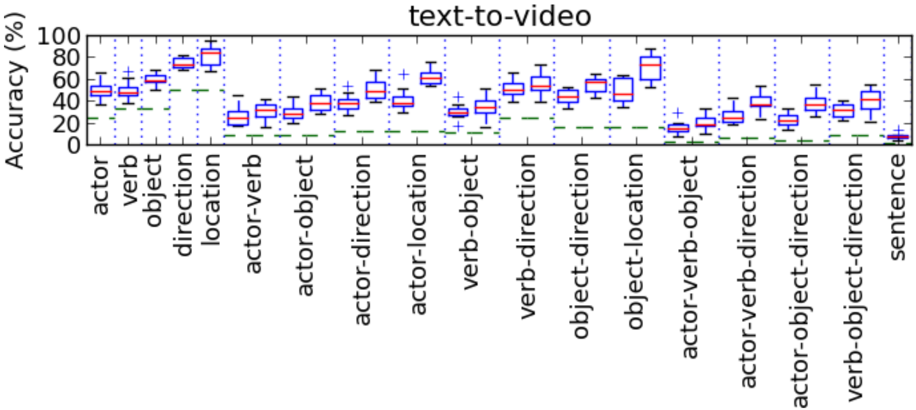
independent on left, joint on right of each pair and triple

Classification Accuracies



independent on left, joint on right of each pair and triple

Classification Accuracies



independent on left, joint on right of each pair and triple

Classification Accuracies

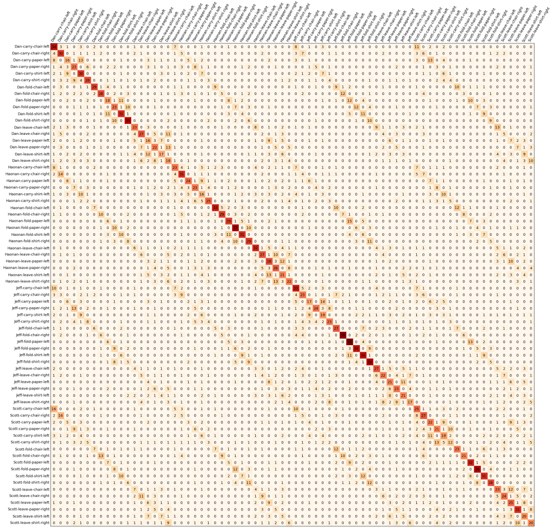
	from-video	from-text	video-to-text	text-to-video	chance
actor	56.1%	46.1%	50.5%	50.2%	25.0%
verb	85.8%	57.5%	36.7%	49.9%	33.3%
object	70.6%	55.9%	47.1%	59.4%	33.3%
direction	94.2%	74.0%	66.4%	74.6%	50.0%
location	98.4%	80.3%	67.2%	81.1%	50.0%
actor&verb	48.1%	27.2%	18.1%	25.9%	8.3%
actor-verb	33.3%	26.5%	11.6%	30.6%	8.3%
actor&object	39.7%	25.2%	24.3%	29.8%	8.3%
actor-object	36.7%	29.7%	26.5%	38.5%	8.3%
actor&direction	54.1%	34.6%	33.3%	38.2%	12.5%
actor-direction	58.8%	40.8%	38.8%	50.5%	12.5%
actor&location	54.1%	37.1%	34.6%	41.3%	12.5%
actor-location	65.9%	48.2%	46.0%	62.1%	12.5%
verb&object	61.3%	32.5%	17.3%	30.3%	11.1%
verb-object	57.3%	31.2%	14.5%	35.0%	11.1%
verb&direction	83.7%	53.5%	41.8%	51.7%	25.0%
verb-direction	85.8%	51.9%	38.6%	56.1%	25.0%
object&direction	63.0%	42.2%	30.2%	44.0%	16.6%
object-direction	63.9%	45.9%	40.5%	54.8%	16.6%
object&location	76.9%	43.7%	33.2%	49.2%	16.6%
object-location	81.3%	57.4%	38.1%	70.2%	16.6%
actor&verb&object	34.5%	14.8%	8.7%	15.8%	2.7%
actor-verb-object	20.4%	13.7%	5.3%	20.6%	2.7%
actor&verb&direction	48.0%	25.4%	20.9%	26.8%	6.2%
actor-verb-direction	45.3%	26.3%	22.0%	38.3%	6.2%
actor&object&direction	36.0%	19.0%	15.4%	22.6%	4.1%
actor-object-direction	36.6%	24.7%	25.2%	37.9%	4.1%
verb&object&direction	56.6%	30.6%	18.7%	31.3%	8.3%
verb-object-direction	54.2%	30.8%	21.7%	40.4%	8.3%

Classification Accuracies

decode whole sentence	
fMRI from-video	32.2%
fMRI from-text	7.6%
fMRI video-to-text	4.4%
fMRI text-to-video	7.6%
chance	1.4%

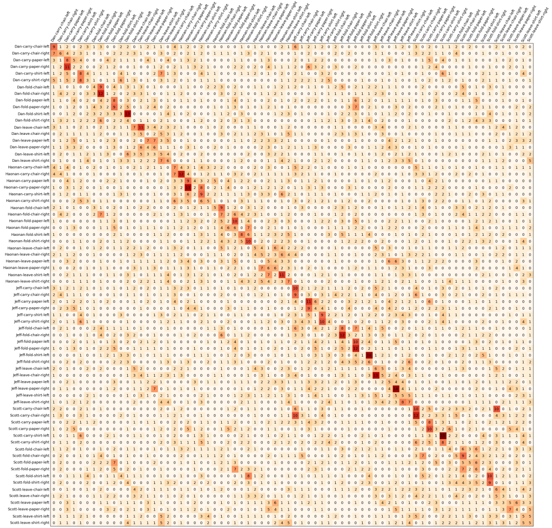
Confusion Matrices

sentence from-video



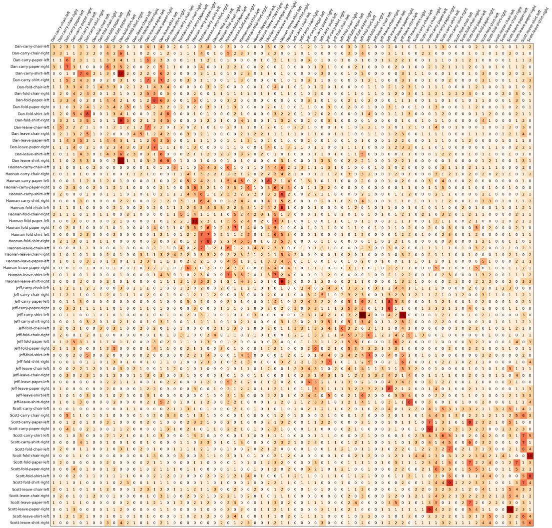
Confusion Matrices

sentence from-text



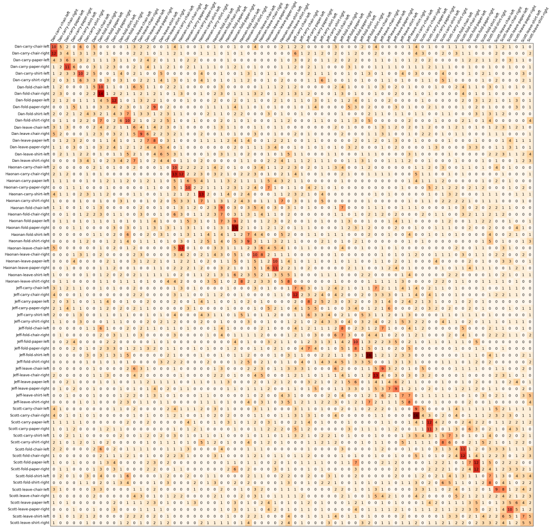
Confusion Matrices

sentence video-to-text



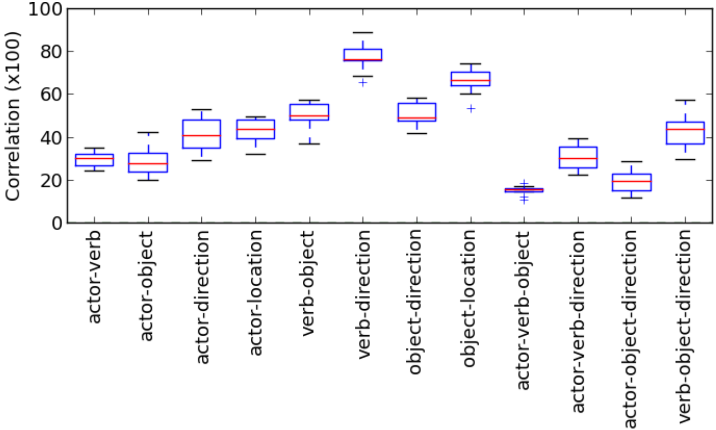
Confusion Matrices

sentence text-to-video



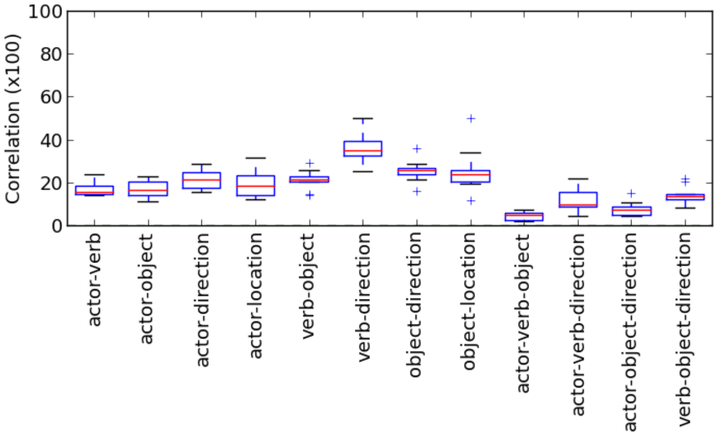
Correlation of Independent and Joint Classifier Judgments

from-video



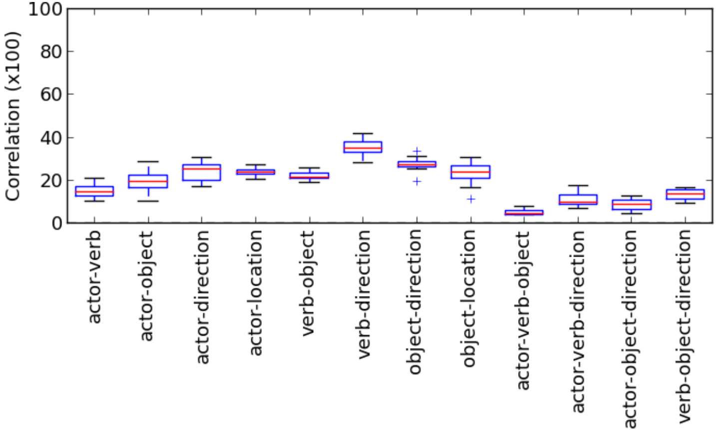
Correlation of Independent and Joint Classifier Judgments

from-text



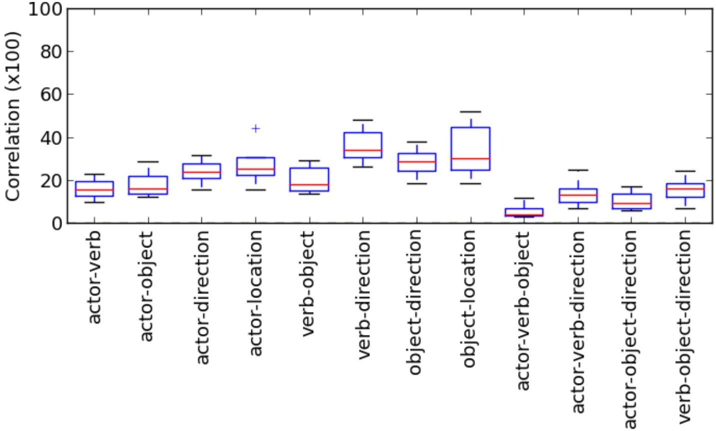
Correlation of Independent and Joint Classifier Judgments

video-to-text



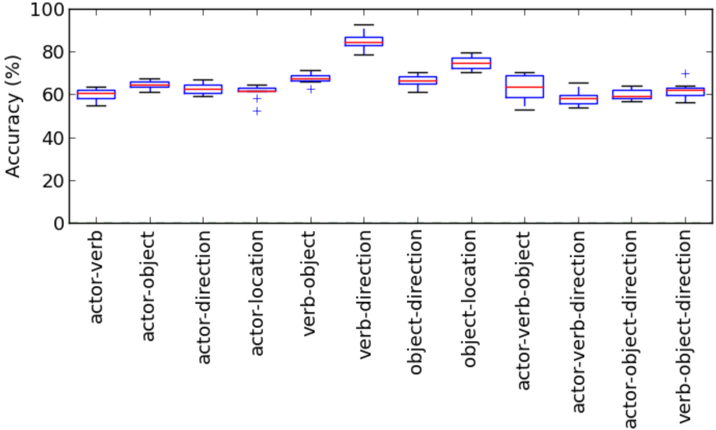
Correlation of Independent and Joint Classifier Judgments

text-to-video



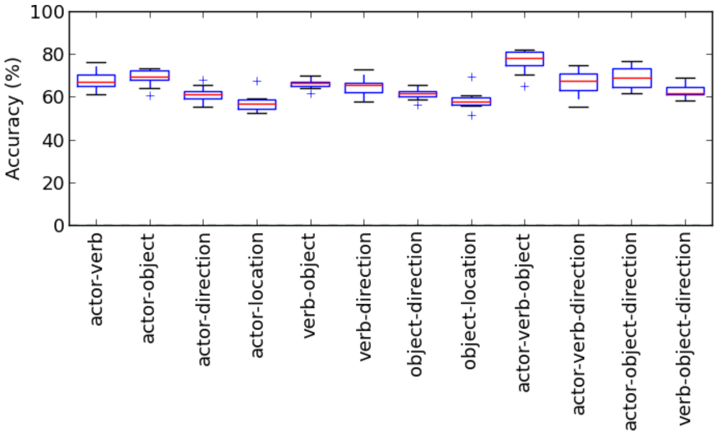
Correctness Agreement of Classifier Judgments

from-video



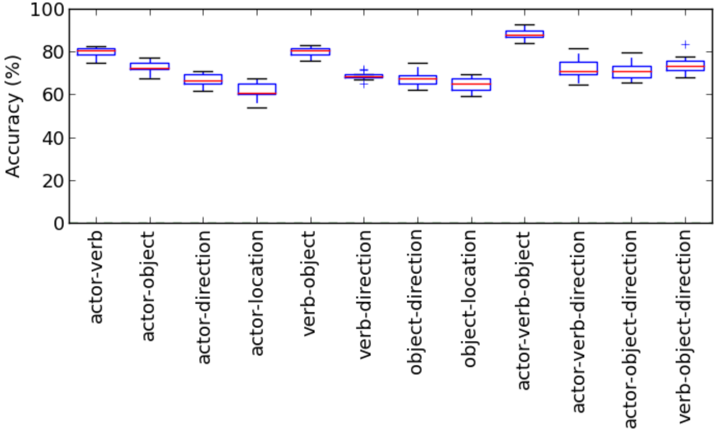
Correctness Agreement of Classifier Judgments

from-text



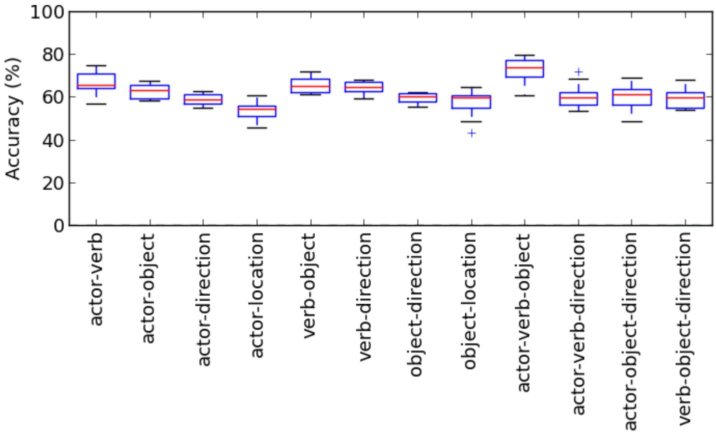
Correctness Agreement of Classifier Judgments

video-to-text

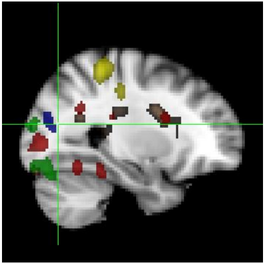


Correctness Agreement of Classifier Judgments

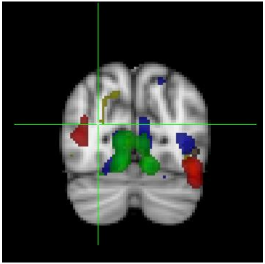
text-to-video



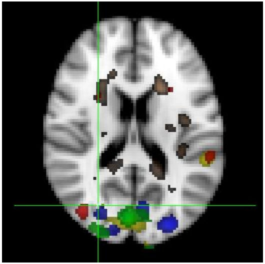
Brain Regions for Constituents



slice 31 (X = 28.00)



slice 27 (Y = -72.00)

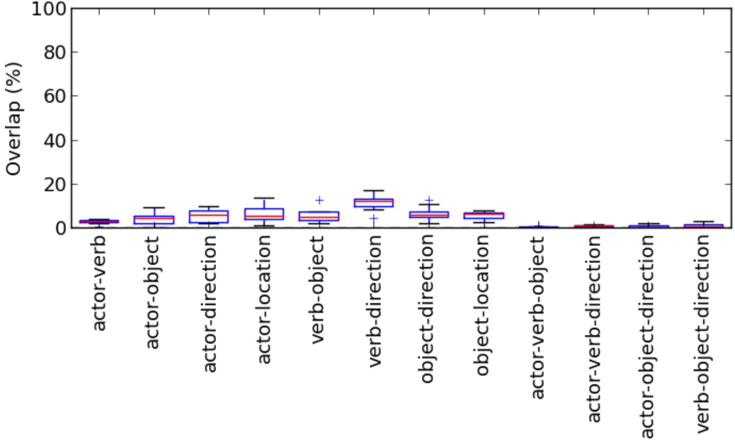


slice 45 (Z = 18.00)

- actor
- verb
- object
- direction
- location

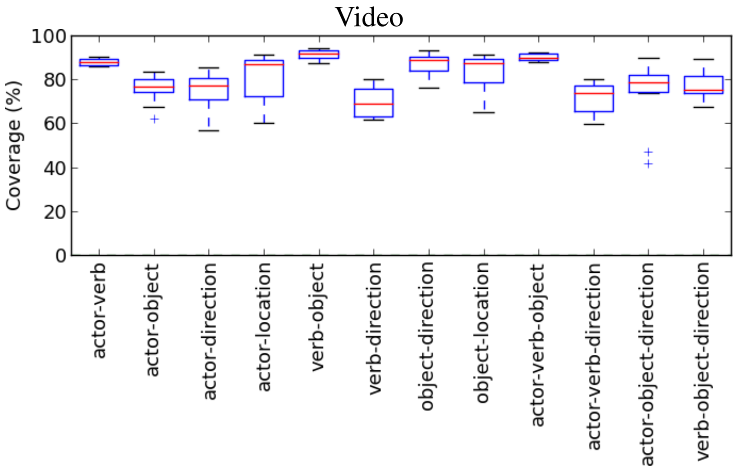
Subject 01, MNI_152, from-video

Video



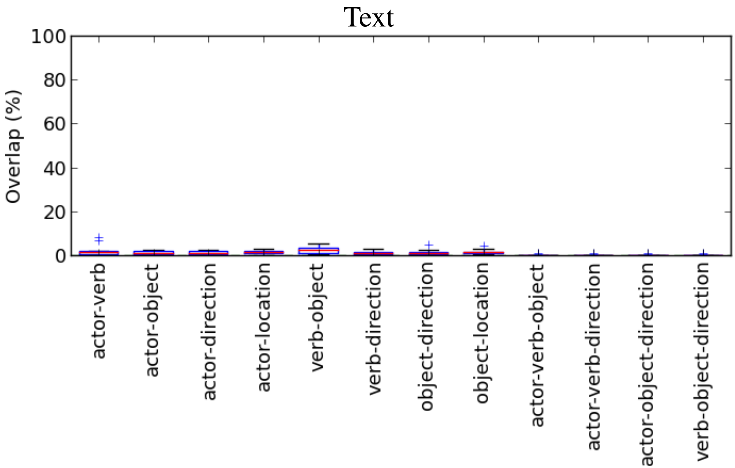
$$\frac{\left|\bigcap_i \text{independent}_i\right|}{\left|\bigcup_i \text{independent}_i\right|}$$

Fraction of overlap between all of the constituent classifier regions in the independent classifier



$$\frac{\left|\left(\bigcup_i \text{independent}_i\right) \cap \text{joint}\right|}{|\text{joint}|}$$

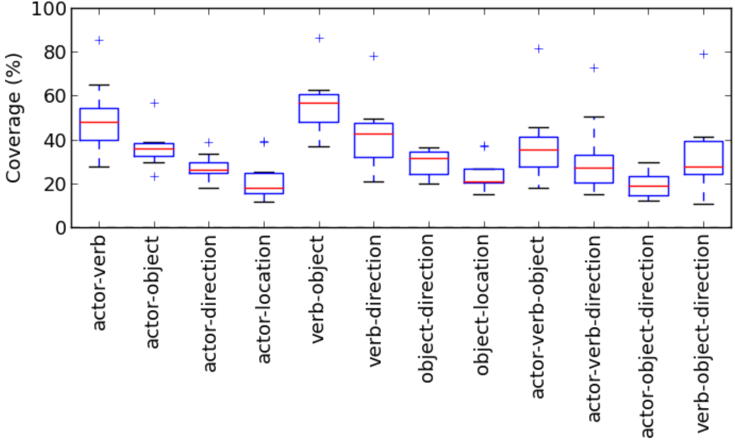
Fraction of the joint classifier region covered by the union of the independent classifier regions



$$\frac{\left|\bigcap_i \text{independent}_i\right|}{\left|\bigcup_i \text{independent}_i\right|}$$

Fraction of overlap between all of the constituent classifier regions in the independent classifier

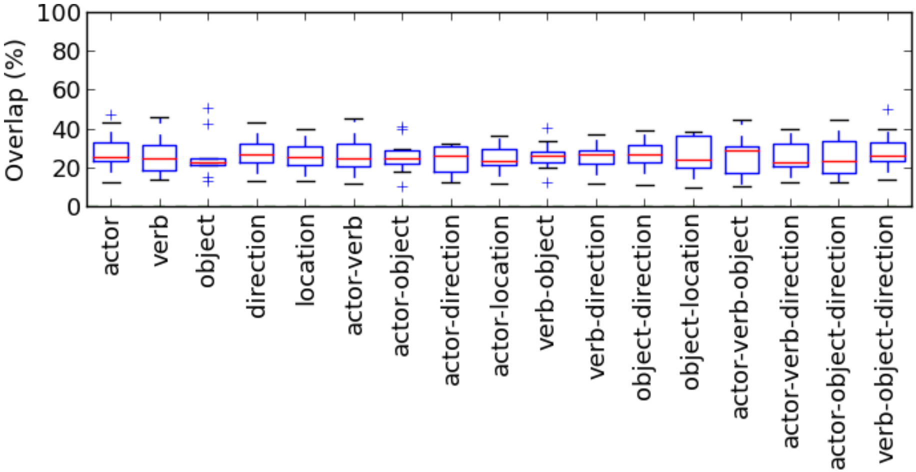
Text



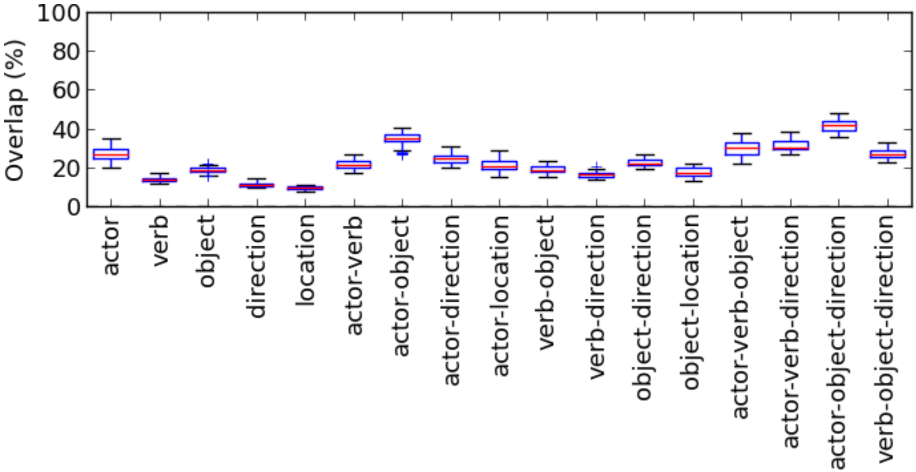
$$\frac{\left|\left(\bigcup_i \text{independent}_i\right) \cap \text{joint}\right|}{|\text{joint}|}$$

Fraction of the joint classifier region covered by the union of the independent classifier regions

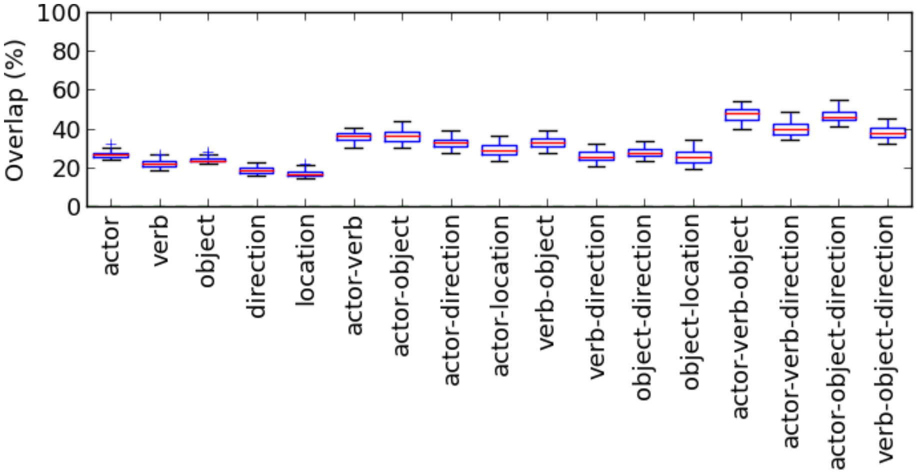
Cross-Modal Overlap



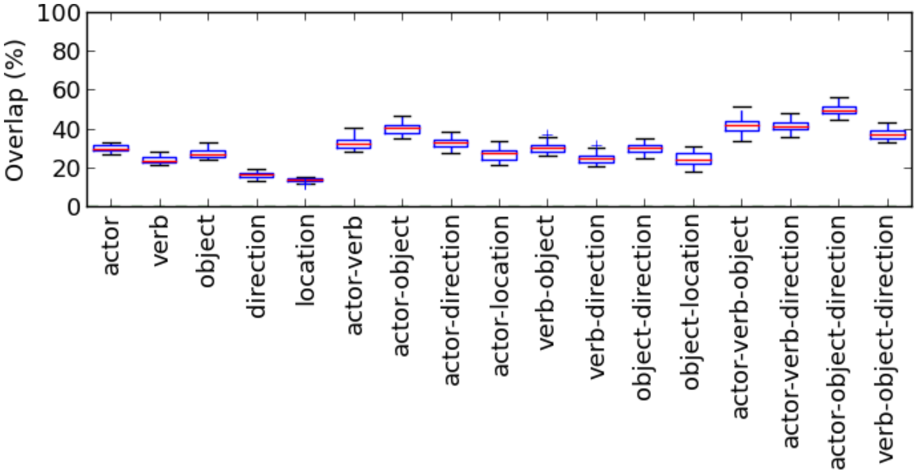
Cross-Subject Video Overlap



Cross-Subject Text Overlap



Cross-Subject Modality-Neutral Overlap



Outline

What underlies a symbolic representation?

- predication

$$walk(John) \wedge talk(Mary)$$
$$walk(Mary) \wedge talk(John)$$

not $\{John, Mary, walk, talk\}$

$$\begin{Bmatrix} \textit{Dan} \\ \textit{Scott} \\ \textit{nobody} \end{Bmatrix} \times \begin{Bmatrix} \textit{pick up} \\ \textit{put down} \\ \textit{does nothing} \end{Bmatrix} \times \begin{Bmatrix} \textit{briefcase} \\ \textit{chair} \\ \textit{nothing} \end{Bmatrix}$$

$$\left\{\begin{matrix} \textit{Dan} \\ \textit{Scott} \\ \textit{nobody} \end{matrix}\right\}$$

actor

$$\left\{\begin{array}{l}\textit{pick up}\\ \textit{put down}\\ \textit{does nothing}\end{array}\right\}$$

verb

$$\left\{\begin{array}{l} \textit{briefcase} \\ \textit{chair} \\ \textit{nothing} \end{array}\right\}$$

object

$$\left\{\begin{array}{l}\textit{Dan}\\ \textit{Scott}\\ \textit{nobody}\end{array}\right\} \times \left\{\begin{array}{l}\textit{pick up}\\ \textit{put down}\\ \textit{does nothing}\end{array}\right\}$$

actor-verb

$$\begin{Bmatrix} \textit{Dan} \\ \textit{Scott} \\ \textit{nobody} \end{Bmatrix} \times \qquad\qquad\qquad \begin{Bmatrix} \textit{briefcase} \\ \textit{chair} \\ \textit{nothing} \end{Bmatrix}$$

actor-object

$$\left\{\begin{array}{l} \textit{pick up} \\ \textit{put down} \\ \textit{does nothing} \end{array}\right\} \times \left\{\begin{array}{l} \textit{briefcase} \\ \textit{chair} \\ \textit{nothing} \end{array}\right\}$$

verb-object

$$\begin{Bmatrix} \textit{Dan} \\ \textit{Scott} \\ \textit{nobody} \end{Bmatrix} \times \begin{Bmatrix} \textit{pick up} \\ \textit{put down} \\ \textit{does nothing} \end{Bmatrix} \times \begin{Bmatrix} \textit{briefcase} \\ \textit{chair} \\ \textit{nothing} \end{Bmatrix}$$

actor-verb-object

$$\left(\begin{Bmatrix} \textit{Dan} \\ \textit{Scott} \\ \textit{nobody} \end{Bmatrix} \times \begin{Bmatrix} \textit{pick up} \\ \textit{put down} \\ \textit{does nothing} \end{Bmatrix} \times \begin{Bmatrix} \textit{briefcase} \\ \textit{chair} \\ \textit{nothing} \end{Bmatrix} \right)^2$$

actor-verb-object

$$\left(\begin{Bmatrix} \textit{Dan} \\ \textit{Scott} \\ \textit{nobody} \end{Bmatrix} \times \begin{Bmatrix} \textit{pick up} \\ \textit{put down} \\ \textit{does nothing} \end{Bmatrix} \times \begin{Bmatrix} \textit{briefcase} \\ \textit{chair} \\ \textit{nothing} \end{Bmatrix} \right)^2$$

actor-verb-object

on the left

$$\left(\begin{Bmatrix} \textit{Dan} \\ \textit{Scott} \\ \textit{nobody} \end{Bmatrix} \times \begin{Bmatrix} \textit{pick up} \\ \textit{put down} \\ \textit{does nothing} \end{Bmatrix} \times \begin{Bmatrix} \textit{briefcase} \\ \textit{chair} \\ \textit{nothing} \end{Bmatrix} \right)^2$$

actor-verb-object

on the left on the right

$$\left(\begin{Bmatrix} \textit{Dan} \\ \textit{Scott} \\ \textit{nobody} \end{Bmatrix} \times \begin{Bmatrix} \textit{pick up} \\ \textit{put down} \\ \textit{does nothing} \end{Bmatrix} \times \begin{Bmatrix} \textit{briefcase} \\ \textit{chair} \\ \textit{nothing} \end{Bmatrix} \right)^2$$

actor-verb-object

on the left on the right

- can be two *briefcase*s or two *chair*s, but only one *Dan* and one *Scott*

$$\left(\begin{Bmatrix} \textit{Dan} \\ \textit{Scott} \\ \textit{nobody} \end{Bmatrix} \times \begin{Bmatrix} \textit{pick up} \\ \textit{put down} \\ \textit{does nothing} \end{Bmatrix} \times \begin{Bmatrix} \textit{briefcase} \\ \textit{chair} \\ \textit{nothing} \end{Bmatrix} \right)^2$$

actor-verb-object

on the left on the right

- can be two *briefcase*s or two *chair*s, but only one *Dan* and one *Scott*
- **verb** requires **actor** and **object**

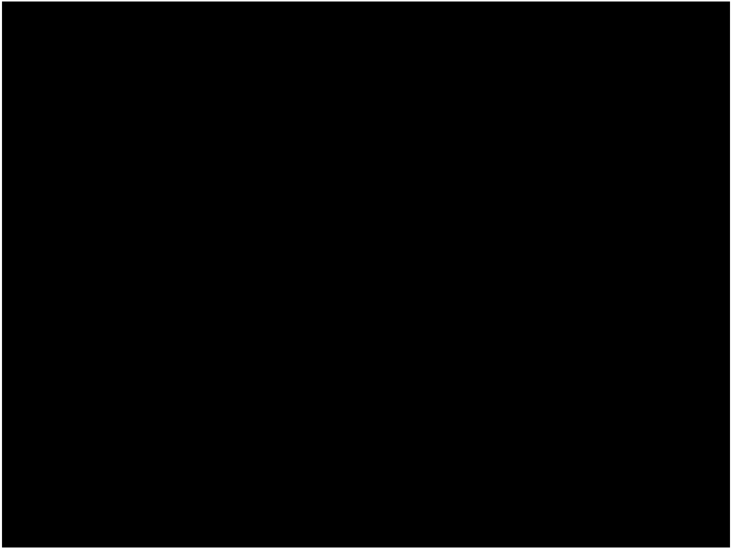
$$\left(\begin{Bmatrix} \textit{Dan} \\ \textit{Scott} \\ \textit{nobody} \end{Bmatrix} \times \begin{Bmatrix} \textit{pick up} \\ \textit{put down} \\ \textit{does nothing} \end{Bmatrix} \times \begin{Bmatrix} \textit{briefcase} \\ \textit{chair} \\ \textit{nothing} \end{Bmatrix} \right)^2$$

actor-verb-object

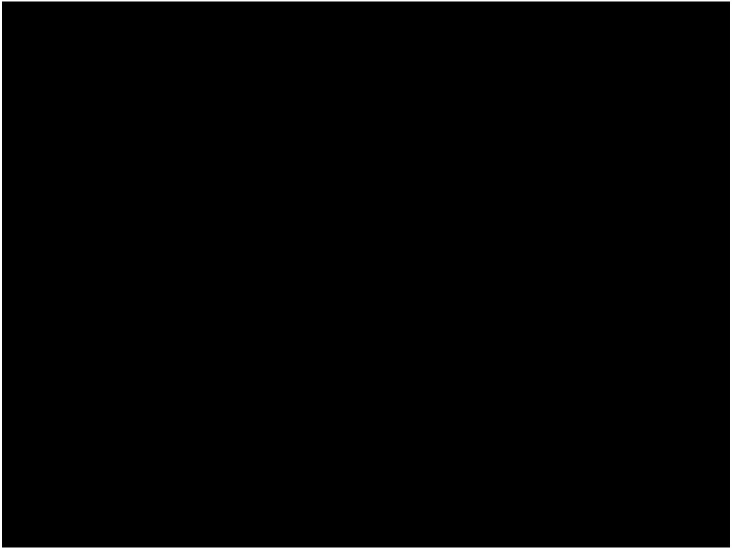
on the left on the right

- can be two *briefcase*s or two *chair*s, but only one *Dan* and one *Scott*
- **verb** requires **actor** and **object**
- something must happen either on the left or on the right

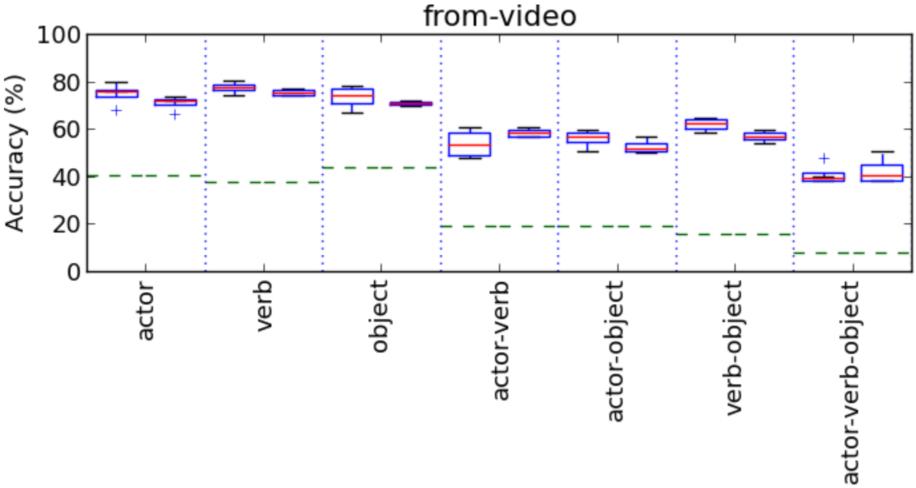
Stimuli



Stimuli

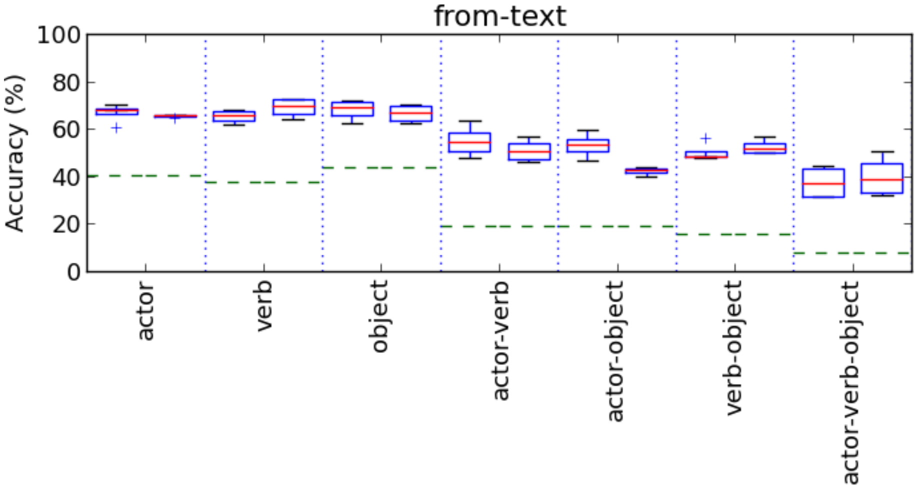


Classification Accuracies



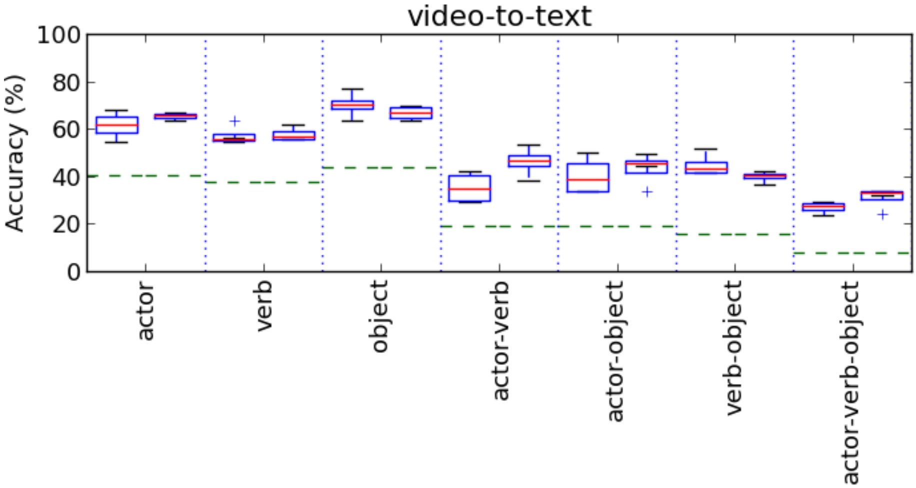
on-left on left, on-right on right of each constituent, pair, and triple

Classification Accuracies



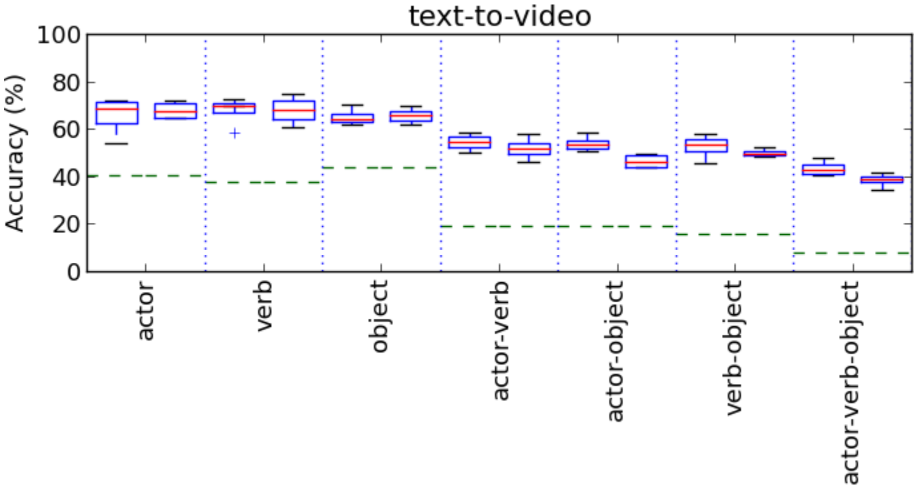
on-left on left, on-right on right of each constituent, pair, and triple

Classification Accuracies



on-left on left, on-right on right of each constituent, pair, and triple

Classification Accuracies



on-left on left, on-right on right of each constituent, pair, and triple

Classification Accuracies

	on-left				on-right				chance
	from-video	from-text	video-to-text	text-to-video	from-video	from-text	video-to-text	text-to-video	
actor	74.80%	66.79%	61.52%	65.62%	71.09%	65.42%	65.42%	67.96%	40.62%
verb	77.53%	65.23%	57.42%	67.77%	75.58%	69.14%	57.61%	67.96%	37.50%
object	73.43%	68.16%	70.31%	65.03%	70.70%	66.60%	66.60%	65.62%	43.75%
actor-verb	53.90%	54.88%	35.15%	54.49%	58.59%	50.97%	46.28%	51.75%	18.75%
actor-object	55.85%	53.12%	40.23%	53.90%	52.73%	42.18%	43.35%	46.28%	18.75%
verb-object	62.10%	50.19%	44.72%	52.53%	56.83%	52.53%	39.84%	50.00%	15.62%

Classification Accuracies

decode whole sentence

	on-left	on-right
fMRI from-video	41.0%	42.5%
fMRI from-text	37.5%	40.0%
fMRI video-to-text	26.7%	30.8%
fMRI text-to-video	43.3%	38.2%
chance	7.9%	7.9%

Confusion Matrices

actor-verb-object from-video

on-left

	Dan-pick-up-briefcase	Dan-pick-up-chair	Dan-put-down-briefcase	Dan-put-down-chair	Dan-does-nothing-briefcase	Dan-does-nothing-chair	Dan-does-nothing-nothing	Scott-pick-up-briefcase	Scott-pick-up-chair	Scott-put-down-briefcase	Scott-put-down-chair	Scott-does-nothing-briefcase	Scott-does-nothing-chair	Scott-does-nothing-nothing	nobody-does-nothing-briefcase	nobody-does-nothing-chair	nobody-does-nothing-nothing
Dan-pick-up-briefcase	23	3	1	2	0	0	0	3	1	3	3	0	0	0	0	1	0
Dan-pick-up-chair	5	21	2	2	0	0	0	3	5	0	2	0	0	0	0	0	0
Dan-put-down-briefcase	3	2	10	3	0	0	0	4	2	9	4	0	0	1	1	1	0
Dan-put-down-chair	2	1	4	14	0	1	0	3	1	3	9	0	0	0	1	1	0
Dan-does-nothing-briefcase	0	0	1	0	4	0	0	1	0	5	3	0	1	0	0	1	0
Dan-does-nothing-chair	0	0	0	0	3	4	0	3	0	2	3	0	0	0	0	0	1
Dan-does-nothing-nothing	0	0	2	1	0	1	6	0	0	0	2	0	0	1	1	1	1
Scott-pick-up-briefcase	5	3	1	5	0	3	0	15	2	1	5	0	0	0	0	0	0
Scott-pick-up-chair	1	10	5	2	1	0	0	0	12	3	4	0	0	0	0	0	2
Scott-put-down-briefcase	0	1	8	1	0	1	0	2	0	22	4	0	0	1	0	0	0
Scott-put-down-chair	3	0	3	4	0	0	0	4	3	6	12	1	0	0	0	3	1
Scott-does-nothing-briefcase	0	1	0	0	2	0	0	0	0	5	1	4	3	0	0	0	0
Scott-does-nothing-chair	0	0	0	0	1	0	1	0	0	6	2	2	1	2	0	1	0
Scott-does-nothing-nothing	0	4	0	1	0	0	1	1	0	1	1	0	0	5	1	1	0
nobody-does-nothing-briefcase	1	0	0	0	0	0	0	0	0	0	1	0	0	2	21	3	4
nobody-does-nothing-chair	1	2	2	0	0	0	0	1	1	2	1	0	0	0	4	17	1
nobody-does-nothing-nothing	0	1	1	0	1	0	0	0	1	1	1	0	0	0	4	3	19

on-right

	Dan-pick-up-briefcase	Dan-pick-up-chair	Dan-put-down-briefcase	Dan-put-down-chair	Dan-does-nothing-briefcase	Dan-does-nothing-chair	Dan-does-nothing-nothing	Scott-pick-up-briefcase	Scott-pick-up-chair	Scott-put-down-briefcase	Scott-put-down-chair	Scott-does-nothing-briefcase	Scott-does-nothing-chair	Scott-does-nothing-nothing	nobody-does-nothing-briefcase	nobody-does-nothing-chair	nobody-does-nothing-nothing
Dan-pick-up-briefcase	16	1	5	3	0	0	1	3	2	5	1	1	0	0	0	1	1
Dan-pick-up-chair	8	15	2	1	0	0	0	1	2	3	5	0	0	1	1	1	0
Dan-put-down-briefcase	1	2	19	2	0	0	0	2	3	6	4	0	0	0	0	0	1
Dan-put-down-chair	4	2	7	9	0	0	0	1	3	5	3	0	0	1	3	2	0
Dan-does-nothing-briefcase	0	0	0	0	3	0	0	1	1	5	6	0	0	0	0	0	0
Dan-does-nothing-chair	0	0	0	0	0	4	0	1	3	5	2	1	0	0	0	0	0
Dan-does-nothing-nothing	1	0	0	1	0	0	7	0	0	3	2	0	0	1	0	1	0
Scott-pick-up-briefcase	2	4	2	3	0	0	0	16	3	6	3	0	0	0	0	0	1
Scott-pick-up-chair	3	2	2	3	0	0	0	1	19	2	4	0	1	1	1	0	1
Scott-put-down-briefcase	6	1	5	1	1	0	0	0	0	19	3	1	0	0	2	0	1
Scott-put-down-chair	2	3	3	4	0	1	1	4	1	4	17	0	0	0	0	0	0
Scott-does-nothing-briefcase	2	1	1	0	0	0	0	1	0	3	0	7	0	0	0	1	0
Scott-does-nothing-chair	0	0	0	0	0	0	0	0	7	0	0	0	6	1	1	1	0
Scott-does-nothing-nothing	0	1	0	1	0	0	0	0	1	1	2	0	0	9	1	0	0
nobody-does-nothing-briefcase	1	0	1	1	0	0	0	1	0	3	1	0	0	1	17	3	3
nobody-does-nothing-chair	0	1	5	1	2	0	0	0	0	1	1	0	0	0	2	17	2
nobody-does-nothing-nothing	0	0	2	0	0	0	0	0	1	2	0	0	0	1	5	3	18

Confusion Matrices

actor-verb-object from-text

	Dan-pick-up-briefcase	Dan-pick-up-chair	Dan-put-down-briefcase	Dan-put-down-chair	Dan-does-nothing-briefcase	Dan-does-nothing-chair	Dan-does-nothing-nothing	Scott-pick-up-briefcase	Scott-pick-up-chair	Scott-put-down-briefcase	Scott-put-down-chair	Scott-does-nothing-briefcase	Scott-does-nothing-chair	Scott-does-nothing-nothing	nobody-does-nothing-briefcase	nobody-does-nothing-chair	nobody-does-nothing-nothing
Dan-pick-up-briefcase	20	3	3	2	0	0	0	2	3	3	1	1	0	0	0	1	1
Dan-pick-up-chair	3	16	2	4	0	0	0	4	0	4	3	0	0	0	0	1	3
Dan-put-down-briefcase	7	3	15	0	0	0	0	5	2	3	2	0	0	0	1	1	1
Dan-put-down-chair	2	7	1	19	0	0	0	6	1	3	0	0	0	0	0	0	1
Dan-does-nothing-briefcase	0	1	1	1	8	0	0	2	2	0	0	0	0	0	1	0	0
Dan-does-nothing-chair	1	4	2	1	0	2	0	2	1	0	2	0	0	0	1	0	0
Dan-does-nothing-nothing	0	1	0	0	0	0	5	1	1	3	1	0	0	0	3	0	1
Scott-pick-up-briefcase	2	4	1	3	0	0	0	20	2	2	0	0	0	0	2	1	3
Scott-pick-up-chair	2	4	3	0	1	0	0	5	14	7	1	0	0	0	0	3	0
Scott-put-down-briefcase	5	3	6	2	0	0	0	3	1	14	2	0	0	0	2	1	1
Scott-put-down-chair	0	5	2	0	1	1	0	5	2	6	13	0	0	0	1	2	2
Scott-does-nothing-briefcase	1	2	4	0	0	0	0	3	2	2	0	2	0	0	0	0	0
Scott-does-nothing-chair	1	2	1	0	1	0	0	1	0	4	1	0	3	0	2	0	0
Scott-does-nothing-nothing	1	1	2	1	0	0	0	3	0	1	2	0	0	2	0	0	3
nobody-does-nothing-briefcase	1	0	2	0	0	0	0	5	1	5	4	0	0	0	9	3	2
nobody-does-nothing-chair	2	3	0	1	0	0	0	3	3	4	1	0	0	0	3	10	2
nobody-does-nothing-nothing	0	5	0	0	0	0	0	2	0	2	0	0	0	1	2	0	20

on-left

	Dan-pick-up-briefcase	Dan-pick-up-chair	Dan-put-down-briefcase	Dan-put-down-chair	Dan-does-nothing-briefcase	Dan-does-nothing-chair	Dan-does-nothing-nothing	Scott-pick-up-briefcase	Scott-pick-up-chair	Scott-put-down-briefcase	Scott-put-down-chair	Scott-does-nothing-briefcase	Scott-does-nothing-chair	Scott-does-nothing-nothing	nobody-does-nothing-briefcase	nobody-does-nothing-chair	nobody-does-nothing-nothing
Dan-pick-up-briefcase	14	3	3	1	0	0	0	3	4	5	4	0	1	0	0	0	2
Dan-pick-up-chair	3	17	2	1	1	0	2	1	2	3	1	0	1	0	3	3	0
Dan-put-down-briefcase	3	5	12	3	0	0	0	1	3	5	2	0	0	0	3	1	2
Dan-put-down-chair	2	2	1	21	0	1	0	2	0	3	3	0	0	0	2	0	3
Dan-does-nothing-briefcase	2	1	0	2	4	0	0	2	1	0	0	0	0	0	2	0	2
Dan-does-nothing-chair	3	0	0	2	0	4	0	1	1	1	3	0	0	0	1	0	0
Dan-does-nothing-nothing	1	4	0	0	0	0	4	0	2	1	1	0	0	0	0	1	2
Scott-pick-up-briefcase	3	2	2	5	1	0	0	12	2	3	6	1	0	0	1	1	1
Scott-pick-up-chair	4	1	4	2	1	0	0	3	16	1	3	0	0	0	1	0	4
Scott-put-down-briefcase	2	3	2	2	0	0	0	2	1	22	1	0	0	0	1	3	1
Scott-put-down-chair	4	2	4	3	0	0	0	2	1	2	15	0	1	0	2	4	0
Scott-does-nothing-briefcase	0	1	2	1	0	0	0	1	2	1	2	4	0	0	1	0	1
Scott-does-nothing-chair	3	4	0	0	0	0	0	0	0	1	3	0	4	0	1	0	0
Scott-does-nothing-nothing	1	0	1	2	0	0	0	2	0	1	0	0	0	6	0	1	2
nobody-does-nothing-briefcase	3	5	2	3	0	0	0	2	0	0	1	0	0	0	15	1	0
nobody-does-nothing-chair	2	4	2	1	0	0	0	1	1	1	1	0	0	0	1	17	1
nobody-does-nothing-nothing	1	1	0	3	0	0	0	0	1	4	0	1	0	0	1	2	18

on-right

Confusion Matrices

actor-verb-object video-to-text

	Dan-pick-up-briefcase	Dan-pick-up-chair	Dan-put-down-briefcase	Dan-put-down-chair	Dan-does-nothing-briefcase	Dan-does-nothing-chair	Dan-does-nothing-nothing	Scott-pick-up-briefcase	Scott-pick-up-chair	Scott-put-down-briefcase	Scott-put-down-chair	Scott-does-nothing-briefcase	Scott-does-nothing-chair	Scott-does-nothing-nothing	nobody-does-nothing-briefcase	nobody-does-nothing-chair	nobody-does-nothing-nothing
Dan-pick-up-briefcase	3	1	7	1	0	1	0	6	2	6	8	0	0	2	1	2	0
Dan-pick-up-chair	0	2	1	1	1	1	0	11	1	4	7	1	0	0	1	5	4
Dan-put-down-briefcase	1	0	11	0	1	2	0	7	2	5	5	0	0	1	1	1	3
Dan-put-down-chair	0	1	4	8	0	1	1	4	0	4	5	0	1	3	1	4	3
Dan-does-nothing-briefcase	0	0	0	0	8	1	0	2	1	0	2	0	0	0	0	2	0
Dan-does-nothing-chair	0	0	0	0	0	5	0	2	1	1	3	1	0	0	1	2	0
Dan-does-nothing-nothing	0	0	1	0	1	0	3	5	0	1	2	0	0	1	0	1	1
Scott-pick-up-briefcase	0	1	2	0	1	4	0	19	2	1	5	0	0	0	3	1	1
Scott-pick-up-chair	1	0	1	0	1	1	2	9	11	4	7	0	0	0	1	1	1
Scott-put-down-briefcase	3	1	3	2	0	1	0	3	4	9	7	1	0	3	2	0	1
Scott-put-down-chair	1	0	0	0	1	2	0	2	2	4	17	0	2	1	4	3	1
Scott-does-nothing-briefcase	1	0	1	0	1	0	0	2	1	3	1	5	0	0	0	0	1
Scott-does-nothing-chair	0	0	2	1	0	1	0	3	1	1	1	1	4	0	0	0	1
Scott-does-nothing-nothing	0	0	0	0	0	1	0	1	2	1	3	0	0	7	1	0	0
nobody-does-nothing-briefcase	0	1	1	2	1	0	1	3	1	3	5	0	0	1	10	2	1
nobody-does-nothing-chair	0	0	1	2	1	0	0	5	3	6	2	0	0	1	5	6	0
nobody-does-nothing-nothing	0	0	4	0	1	0	1	4	1	1	2	0	0	2	5	2	9

on-left

	Dan-pick-up-briefcase	Dan-pick-up-chair	Dan-put-down-briefcase	Dan-put-down-chair	Dan-does-nothing-briefcase	Dan-does-nothing-chair	Dan-does-nothing-nothing	Scott-pick-up-briefcase	Scott-pick-up-chair	Scott-put-down-briefcase	Scott-put-down-chair	Scott-does-nothing-briefcase	Scott-does-nothing-chair	Scott-does-nothing-nothing	nobody-does-nothing-briefcase	nobody-does-nothing-chair	nobody-does-nothing-nothing
Dan-pick-up-briefcase	16	0	7	0	1	0	0	1	4	3	3	0	1	0	0	3	1
Dan-pick-up-chair	5	2	5	0	1	1	1	4	5	5	6	1	0	0	2	2	0
Dan-put-down-briefcase	2	0	14	1	0	1	0	2	6	5	5	0	0	1	1	1	1
Dan-put-down-chair	3	1	5	3	2	0	0	5	5	6	4	1	2	1	0	2	0
Dan-does-nothing-briefcase	1	0	3	0	7	0	0	2	2	0	1	0	0	0	0	0	0
Dan-does-nothing-chair	2	0	1	0	0	6	0	0	1	1	4	0	0	0	0	1	0
Dan-does-nothing-nothing	0	0	1	1	1	0	3	1	3	1	2	0	0	0	0	2	1
Scott-pick-up-briefcase	1	3	5	1	0	0	0	17	2	5	2	0	1	1	0	1	1
Scott-pick-up-chair	4	0	3	0	1	1	1	2	15	3	7	0	0	0	1	1	1
Scott-put-down-briefcase	2	1	3	0	2	0	0	3	7	14	5	0	0	0	0	2	1
Scott-put-down-chair	2	0	5	1	0	3	0	4	5	4	11	0	0	1	2	2	0
Scott-does-nothing-briefcase	2	0	2	0	0	0	0	2	1	0	2	6	0	0	0	1	0
Scott-does-nothing-chair	2	0	0	0	0	0	0	0	3	1	3	0	6	0	0	1	0
Scott-does-nothing-nothing	1	0	1	3	1	0	0	1	3	1	1	0	0	3	0	0	1
nobody-does-nothing-briefcase	1	0	4	1	0	1	0	2	7	2	3	0	0	2	7	1	1
nobody-does-nothing-chair	0	0	4	1	1	1	0	3	3	3	1	0	0	0	0	14	1
nobody-does-nothing-nothing	0	0	1	0	1	1	0	1	0	3	5	0	0	2	0	4	14

on-right

Confusion Matrices

actor-verb-object text-to-video

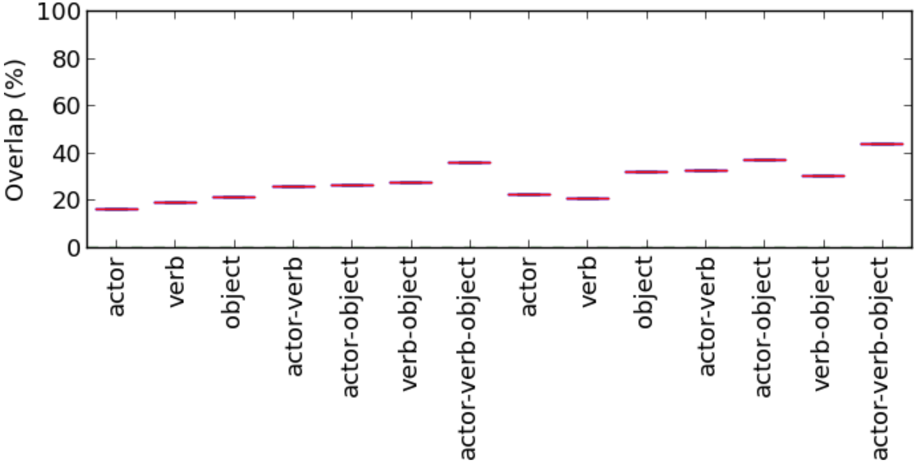
	Dan-pick-up-briefcase	Dan-pick-up-chair	Dan-put-down-briefcase	Dan-put-down-chair	Dan-does-nothing-briefcase	Dan-does-nothing-chair	Dan-does-nothing-nothing	Scott-pick-up-briefcase	Scott-pick-up-chair	Scott-put-down-briefcase	Scott-put-down-chair	Scott-does-nothing-briefcase	Scott-does-nothing-chair	Scott-does-nothing-nothing	nobody-does-nothing-briefcase	nobody-does-nothing-chair	nobody-does-nothing-nothing
Dan-pick-up-briefcase	13	2	1	1	0	1	0	3	4	2	0	0	1	0	4	6	2
Dan-pick-up-chair	4	10	4	5	0	0	1	1	2	2	4	1	0	0	1	3	2
Dan-put-down-briefcase	0	2	16	1	0	0	0	6	1	2	2	0	0	0	3	5	2
Dan-put-down-chair	2	2	1	22	0	0	0	2	2	1	5	0	0	0	1	1	1
Dan-does-nothing-briefcase	0	0	3	1	2	0	0	0	3	1	1	1	0	0	2	2	0
Dan-does-nothing-chair	1	1	1	1	0	5	0	0	0	1	2	0	0	0	3	1	0
Dan-does-nothing-nothing	1	0	0	2	0	0	9	1	0	0	0	0	0	0	0	0	3
Scott-pick-up-briefcase	0	3	1	3	0	0	0	20	4	3	0	0	0	0	2	0	4
Scott-pick-up-chair	2	1	0	3	0	0	1	3	22	1	1	0	0	0	0	3	3
Scott-put-down-briefcase	1	2	2	1	0	1	0	4	4	19	3	0	0	0	1	1	1
Scott-put-down-chair	3	3	1	1	0	1	0	2	3	3	19	0	0	0	0	1	3
Scott-does-nothing-briefcase	1	2	2	0	0	0	1	1	1	0	1	6	0	0	1	0	0
Scott-does-nothing-chair	0	1	0	1	0	0	0	2	4	0	2	1	2	0	1	1	1
Scott-does-nothing-nothing	0	0	0	0	0	1	0	1	3	2	1	0	0	6	1	0	1
nobody-does-nothing-briefcase	2	4	0	0	0	0	0	3	0	0	0	0	0	0	17	4	2
nobody-does-nothing-chair	0	2	0	0	0	0	0	5	3	0	2	0	0	0	1	18	1
nobody-does-nothing-nothing	0	1	2	1	0	0	0	2	1	2	2	0	0	0	4	1	16

on-left

	Dan-pick-up-briefcase	Dan-pick-up-chair	Dan-put-down-briefcase	Dan-put-down-chair	Dan-does-nothing-briefcase	Dan-does-nothing-chair	Dan-does-nothing-nothing	Scott-pick-up-briefcase	Scott-pick-up-chair	Scott-put-down-briefcase	Scott-put-down-chair	Scott-does-nothing-briefcase	Scott-does-nothing-chair	Scott-does-nothing-nothing	nobody-does-nothing-briefcase	nobody-does-nothing-chair	nobody-does-nothing-nothing
Dan-pick-up-briefcase	15	3	2	1	0	1	0	3	1	2	3	0	0	1	3	3	2
Dan-pick-up-chair	3	19	3	3	0	0	0	3	2	1	0	0	0	1	1	1	3
Dan-put-down-briefcase	3	3	23	2	0	0	0	2	0	2	3	0	0	0	0	2	0
Dan-put-down-chair	1	3	4	14	0	1	0	2	3	1	5	0	0	2	1	1	2
Dan-does-nothing-briefcase	0	4	4	1	1	0	0	0	1	2	3	0	0	0	0	0	0
Dan-does-nothing-chair	0	0	0	1	0	7	0	0	1	1	1	0	0	1	0	3	1
Dan-does-nothing-nothing	0	1	1	0	0	0	2	0	1	5	2	0	0	0	1	0	3
Scott-pick-up-briefcase	1	3	4	3	0	0	0	19	3	2	1	0	0	0	2	1	1
Scott-pick-up-chair	4	1	0	2	0	0	0	1	21	6	2	0	0	0	1	1	1
Scott-put-down-briefcase	1	3	8	3	0	0	0	1	3	9	4	0	1	0	0	3	4
Scott-put-down-chair	1	4	4	4	0	0	0	2	2	4	16	0	0	0	0	1	2
Scott-does-nothing-briefcase	1	1	3	3	0	0	0	1	1	0	1	4	0	0	0	1	0
Scott-does-nothing-chair	0	2	1	0	0	0	1	1	1	3	1	0	4	0	1	1	0
Scott-does-nothing-nothing	0	0	1	2	0	0	0	0	1	1	3	0	0	5	2	1	0
nobody-does-nothing-briefcase	0	5	3	0	0	0	0	1	2	3	3	0	0	1	9	3	2
nobody-does-nothing-chair	1	1	3	2	0	1	0	1	2	3	2	0	0	0	0	12	4
nobody-does-nothing-nothing	3	2	2	2	0	1	0	1	3	0	0	0	0	0	2	0	16

on-right

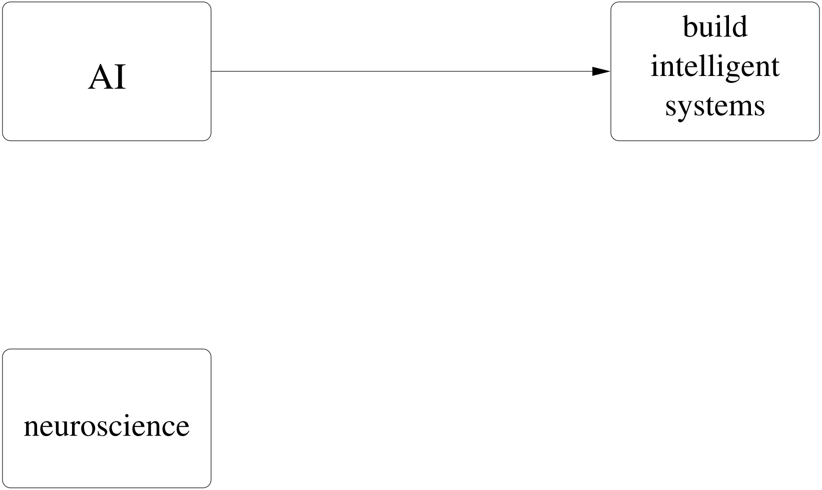
Cross-Modal Overlap

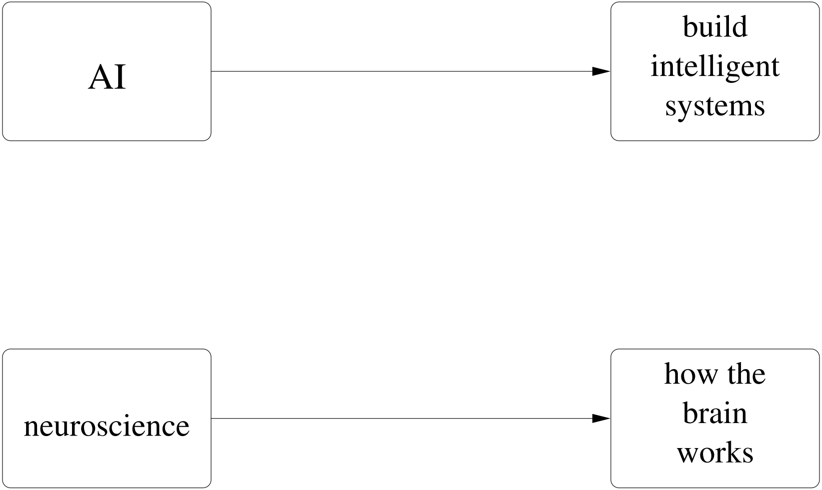


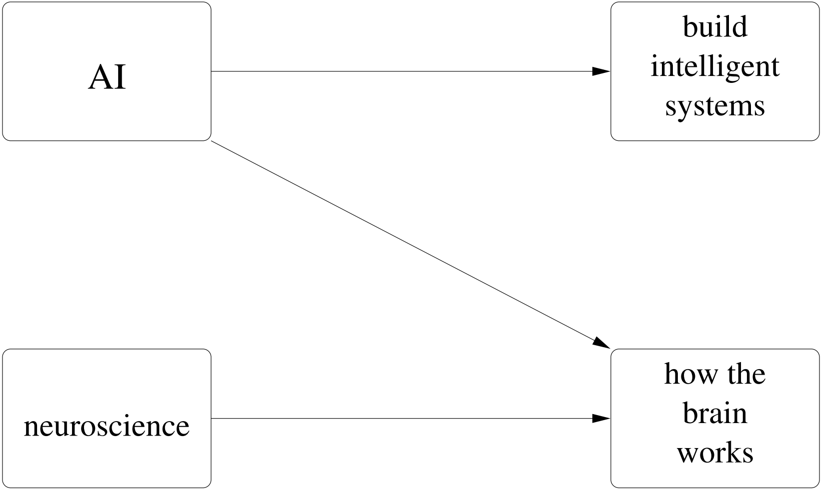
Summary

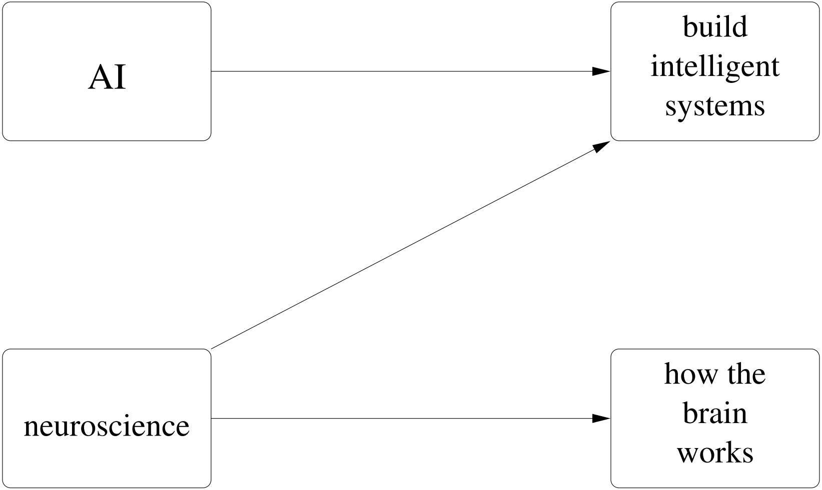
AI


AI
build
intelligent
systems


AI
build intelligent systems
neuroscience


AI
build intelligent systems
neuroscience
how the brain works


AI
build intelligent systems
neuroscience
how the brain works


AI
build intelligent systems
neuroscience
how the brain works

$\Longrightarrow$

$\Longrightarrow$

$\Longrightarrow$

Kiss

 $\Longrightarrow$  $\Longrightarrow$ *Kiss*

$\Longrightarrow$

$\Longrightarrow$

Kiss

$\Longrightarrow$

 $\Longrightarrow$  $\Longrightarrow$ *Kiss*

 $\Longrightarrow$  $\Longrightarrow$ *Kiss*

 $\Longrightarrow$  $\Longrightarrow$ *Kiss*

$=$

 $\Longrightarrow$  $\Longrightarrow$ *Kiss*

 $\Longrightarrow$  $\Longrightarrow$ *Kiss*

$=$ $=$

 $\Longrightarrow$  $\Longrightarrow$ *Kiss*

 $\Longrightarrow$  $\Longrightarrow$ Kiss

$=$ $=$ $=$

 $\Longrightarrow$  $\Longrightarrow$ Kiss

```
(define (AI)
 (let loop (mental-state '())
  (let* ((percept (perceive eyes ears skin tongue nose))
         (action (cognition percepts)))
   (act! arms legs mouth action)
   (loop mental-state))))
```

$\Downarrow$

 $\Longrightarrow$  $\Longrightarrow$ *Kiss*

$=$ $=$ $=$

 $\Longrightarrow$  $\Longrightarrow$ *Kiss*

```
(define (AI)
 (let loop (mental-state '())
  (let* ((percept (perceive eyes ears skin tongue nose))
         (action (cognition percepts)))
   (act! arms legs mouth action)
   (loop mental-state))))
```

$\Downarrow$

 $\Longrightarrow$  $\Longrightarrow$ *Kiss*

$=$ $=$ $=$

 $\Longrightarrow$  $\Longrightarrow$ *Kiss*

$\Uparrow$

```
(define (AI)
 (let loop (mental-state '())
  (let* ((percept (perceive eyes ears skin tongue nose))
         (action (cognition percepts)))
   (act! arms legs mouth action)
   (loop mental-state))))
```

$\Downarrow$

 $\Longrightarrow$  $\Longrightarrow$ *Kiss*

$=$ $=$ $=$

 $\Longrightarrow$  $\Longrightarrow$ *Kiss*

$\Uparrow$

```
(define (AI)
 (let loop (mental-state '())
  (let* ((percept (perceive eyes ears skin tongue nose))
         (action (cognition percepts)))
   (act! arms legs mouth action)
   (loop mental-state))))
```

$\Downarrow$

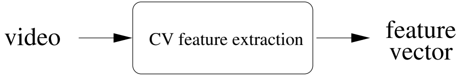
 $\Longrightarrow$  $\Longrightarrow$ *Kiss*

$=$ $=$ $=$

 $\Longrightarrow$  $\Longrightarrow$ *Kiss*

$\Uparrow$

video

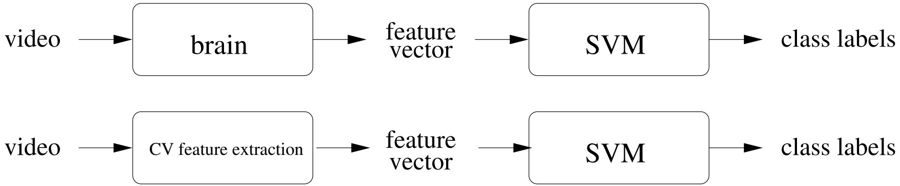

video
CV feature extraction
feature vector

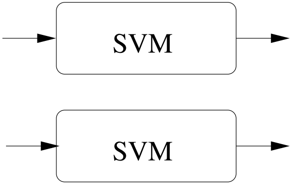

video
CV feature extraction
feature vector
SVM
class labels

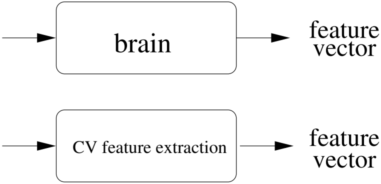
video

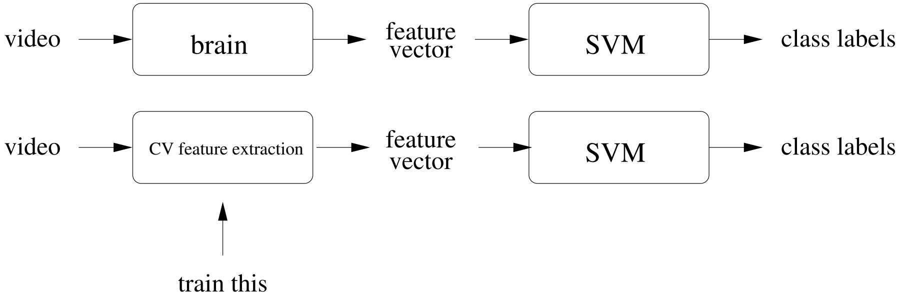

video
brain
feature
vector

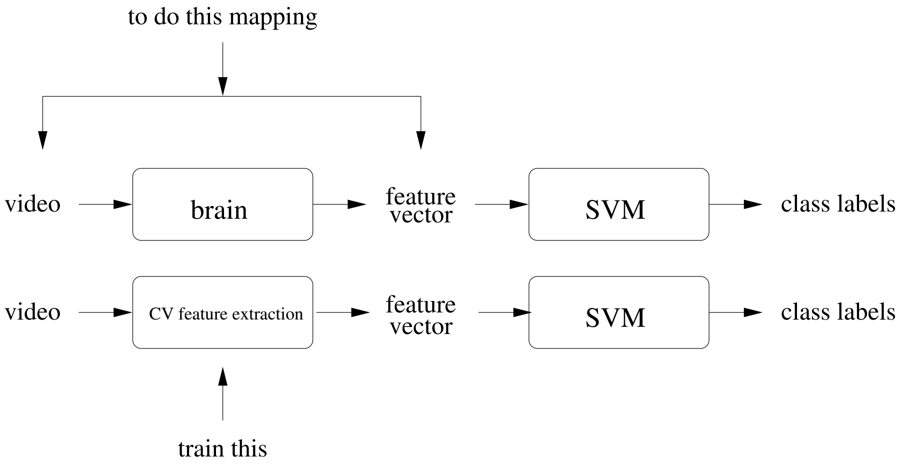

video
brain
feature vector
SVM
class labels

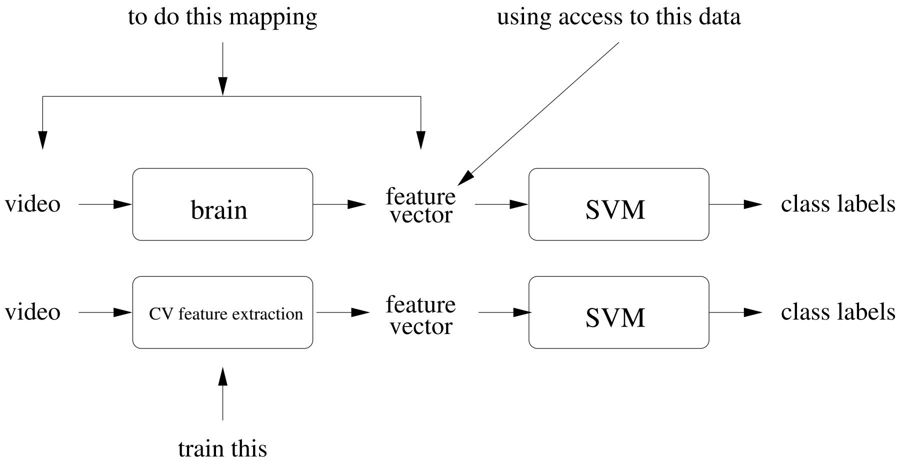

video
brain
feature vector
SVM
class labels
video
CV feature extraction
feature vector
SVM
class labels


SVM
SVM


brain
feature vector
CV feature extraction
feature vector


video
brain
feature vector
SVM
class labels
video
CV feature extraction
feature vector
SVM
class labels
train this

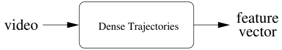

to do this mapping
video
brain
feature vector
SVM
class labels
video
CV feature extraction
feature vector
SVM
class labels
train this


to do this mapping
using access to this data
video
brain
feature vector
SVM
class labels
video
CV feature extraction
feature vector
SVM
class labels
train this

Hot Off The Press

video

Hot Off The Press



Hot Off The Press

Hot Off The Press

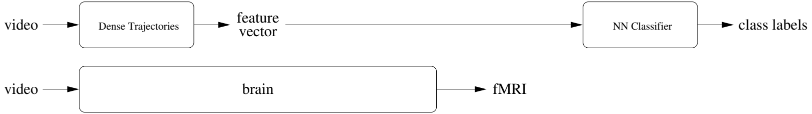
52.5%

Hot Off The Press

52.5%

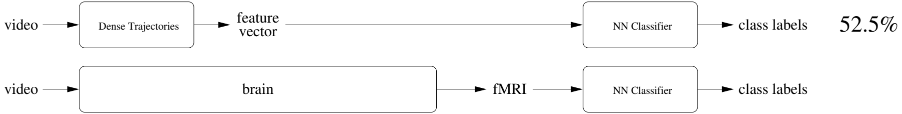
video

Hot Off The Press

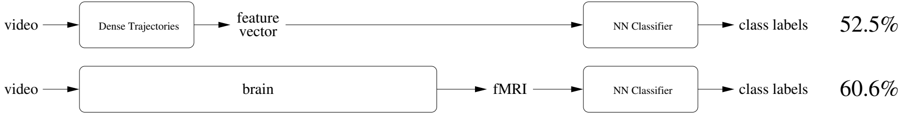


52.5%

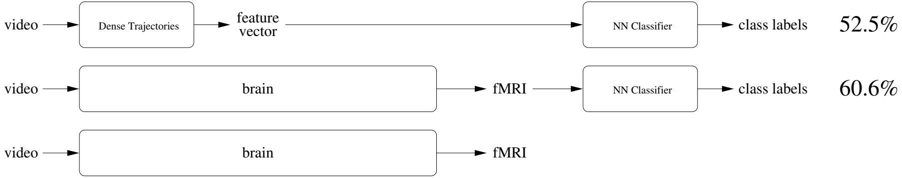
Hot Off The Press



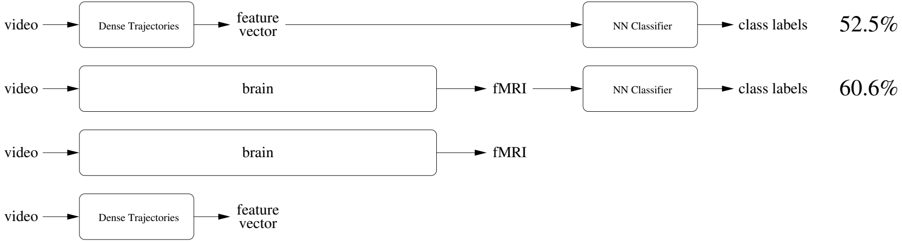
Hot Off The Press



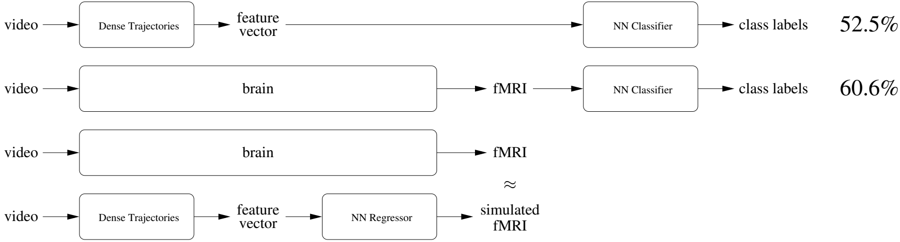
Hot Off The Press



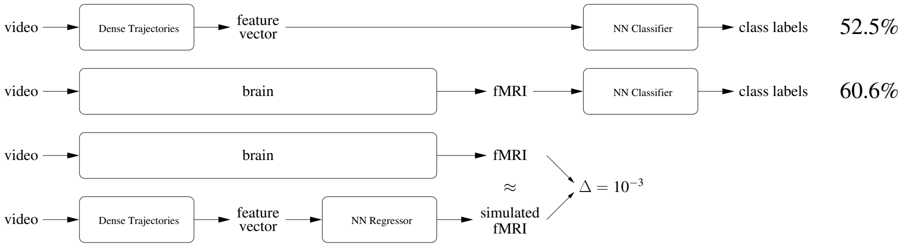
Hot Off The Press



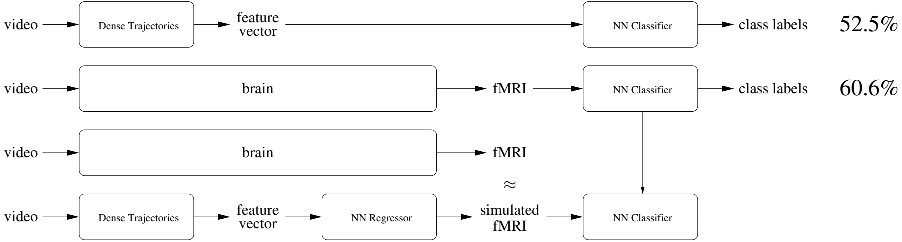
Hot Off The Press



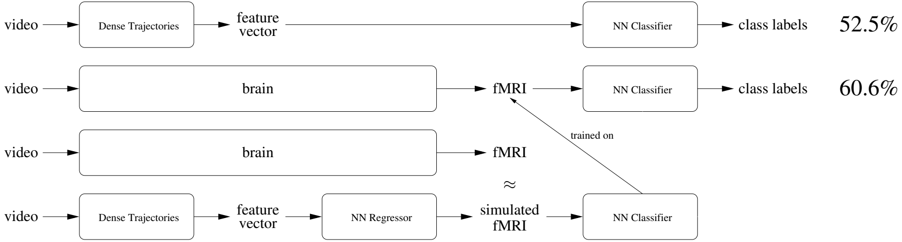
Hot Off The Press



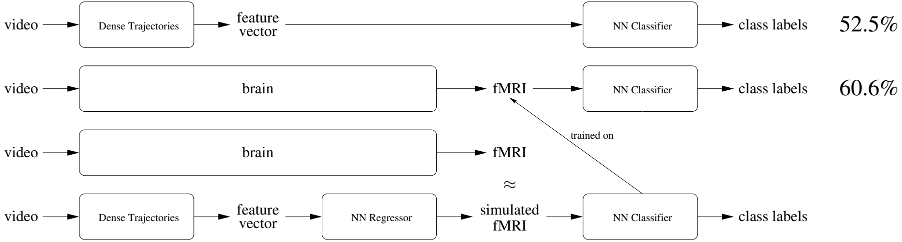
Hot Off The Press



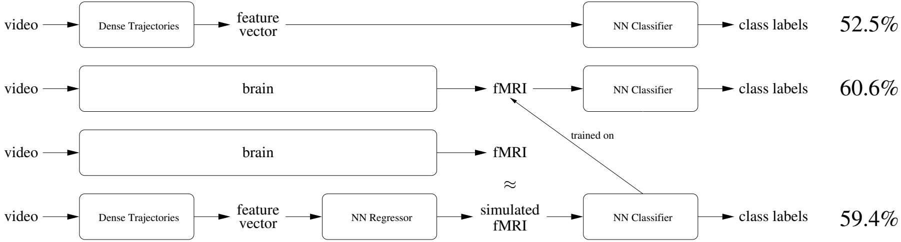
Hot Off The Press



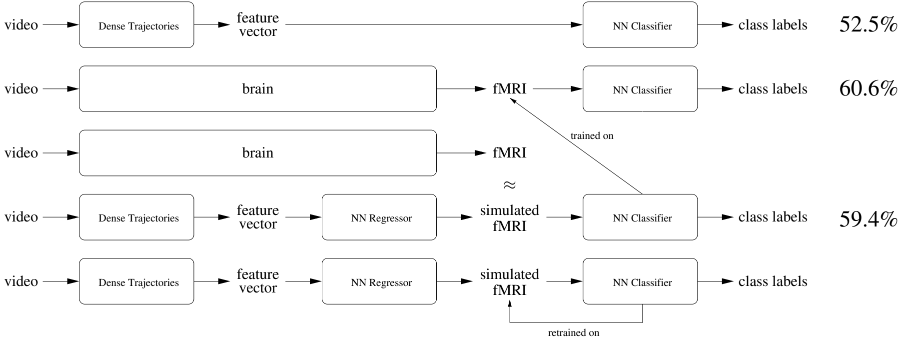
Hot Off The Press



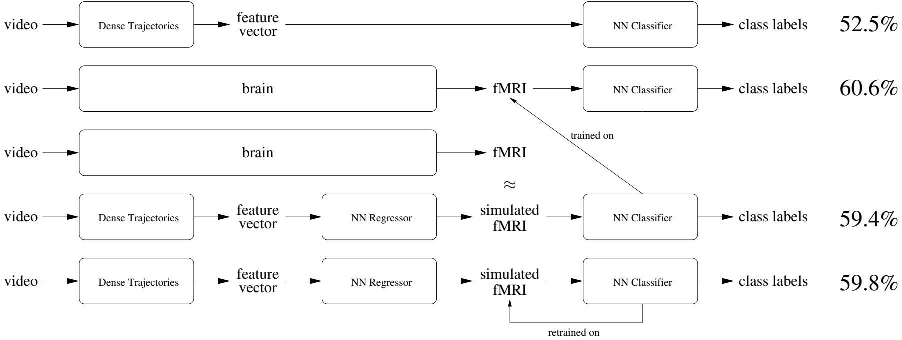
Hot Off The Press



Hot Off The Press



Hot Off The Press



Outline

1. Verbs, Arguments, and Predication in the Human Brain
   - Experiment 1: hollywood2-text-speech
   - Experiment 2: compositionality-noninterleaved
   - Experiment 3: predication
2. Sentence Directed Video Object Codetection
3. Driving Under the Influence (of Language)
   - Grounding Language Semantics in Robotics
   - Object Codetection from Mobile Robot Video
4. Playing Checkers from English

Joint work

Haonan Yu

Sentence-Directed Video Object Codetection

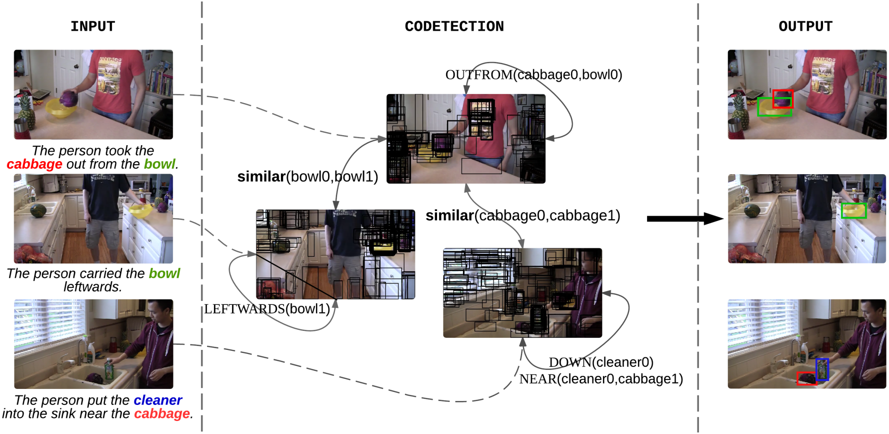
Sentence-Directed Video Object Codetection

video captioning: video+detections→sentences

Sentence-Directed Video Object Codetection

video captioning:	video+detections→sentences
inverse video captioning:	video+sentences→detections

Overview



INPUT
CODETECTION
OUTPUT
The person took the cabbage out from the bowl.
The person carried the bowl leftwards.
The person put the cleaner into the sink near the cabbage.
OUTFROM(cabbage0,bowl0)
similar(bowl0,bowl1)
similar(cabbage0,cabbage1)
LEFTWARDS(bowl1)
DOWN(cleaner0)
NEAR(cleaner0,cabbage1)

The 7 'No's

The 7 'No's

- no background subtraction

The 7 ‘No’s

- no background subtraction
- no object detector

The 7 ‘No’s

- no background subtraction
- no object detector
- no object models

The 7 ‘No’s

- no background subtraction
- no object detector
- no object models
- no per-object-class parameters

The 7 ‘No’s

- no background subtraction
- no object detector
- no object models
- no per-object-class parameters
- no learning

The 7 ‘No’s

- no background subtraction
- no object detector
- no object models
- no per-object-class parameters
- no learning
- no training data

The 7 ‘No’s

- no background subtraction
- no object detector
- no object models
- no per-object-class parameters
- no learning
- no training data
- no human-annotated bounding boxes

The person put the cleaner into the sink near the cabbage.

The person put the cleaner into the sink near the cabbage.

$$\textsc{down}(\mathit{cleaner}) \wedge \textsc{near}(\mathit{cleaner}, \mathit{cabbage})$$

The person put the cleaner into the sink near the cabbage.

$$\textsc{down}(\textit{cleaner}) \wedge \textsc{near}(\textit{cleaner}, \textit{cabbage})$$

Generated using Stanford parser (Socher et al. ACL 2013) and methods of Lin et al. (CVPR 2014).

The person put the cleaner into the sink near the cabbage.

$$\textsc{down}(\textit{cleaner}) \wedge \textsc{near}(\textit{cleaner}, \textit{cabbage})$$

Generated using Stanford parser (Socher et al. ACL 2013) and methods of Lin et al. (CVPR 2014).
Predicates are soft.

The person put the cleaner into the sink near the cabbage.

$$\textsc{down}(\mathit{cleaner}) \wedge \textsc{near}(\mathit{cleaner}, \mathit{cabbage})$$

Generated using Stanford parser (Socher et al. ACL 2013) and methods of Lin et al. (CVPR 2014).
Predicates are soft.
Some are unary, some are binary.

Our Predicates

Predicates	Constants
$\textsc{move}(p) \triangleq \mathsf{medFlMg}(p)$	$\Delta\textsc{distLarge} \triangleq 0.25$
$\textsc{moveUp}(p) \triangleq \textsc{move}(p) + \mathsf{distLessThan}\left(\mathsf{y}(p^{(T)}) - \mathsf{y}(p^{(1)}), -\Delta\textsc{distLarge}\right)$	$\Delta\textsc{distSmall} \triangleq 0.05$
$\textsc{moveDown}(p) \triangleq \textsc{move}(p) + \mathsf{distGreaterThan}\left(\mathsf{y}(p^{(T)}) - \mathsf{y}(p^{(1)}), \Delta\textsc{distLarge}\right)$	$\Delta\textsc{angle} \triangleq \pi/2$
$\textsc{moveHorizontal}(p) \triangleq \textsc{move}(p) + \mathsf{distGreaterThan}\left(\left\lvert\mathsf{x}(p^{(T)}) - \mathsf{x}(p^{(1)})\right\rvert, \Delta\textsc{distLarge}\right)$	
$\textsc{moveLeftwards}(p) \triangleq \textsc{move}(p) + \mathsf{distLessThan}\left(\mathsf{x}(p^{(T)}) - \mathsf{x}(p^{(1)}), -\Delta\textsc{distLarge}\right)$	
$\textsc{moveRightwards}(p) \triangleq \textsc{move}(p) + \mathsf{distGreaterThan}\left(\mathsf{x}(p^{(T)}) - \mathsf{x}(p^{(1)}), \Delta\textsc{distLarge}\right)$	
$\textsc{rotate}(p) \triangleq \textsc{move}(p) + \max_t \mathsf{hasRotation}\left(\mathsf{rotAngle}(p^{(t)}), \Delta\textsc{angle}\right)$	
$\textsc{towards}(p_1,p_2) \triangleq \textsc{move}(p_1) + \mathsf{distLessThan}\left(\mathsf{dist}(p_1^{(T)},p_2^{(T)}) - \mathsf{dist}(p_1^{(1)},p_2^{(1)}), -\Delta\textsc{distLarge}\right)$	
$\textsc{awayFrom}(p_1,p_2) \triangleq \textsc{move}(p_1) + \mathsf{distGreaterThan}\left(\mathsf{dist}(p_1^{(T)},p_2^{(T)}) - \mathsf{dist}(p_1^{(1)},p_2^{(1)}), \Delta\textsc{distLarge}\right)$	
$\textsc{leftOfStart}(p_1,p_2) \triangleq \mathsf{tempCoher}(p_2) + \mathsf{distLessThan}\left(\mathsf{x}(p_1^{(1)}) - \mathsf{x}(p_2^{(1)}), -\Delta\textsc{distSmall}\right)$	
$\textsc{leftOfEnd}(p_1,p_2) \triangleq \mathsf{tempCoher}(p_2) + \mathsf{distLessThan}\left(\mathsf{x}(p_1^{(T)}) - \mathsf{x}(p_2^{(T)}), -\Delta\textsc{distSmall}\right)$	
$\textsc{rightOfStart}(p_1,p_2) \triangleq \mathsf{tempCoher}(p_2) + \mathsf{distGreaterThan}\left(\mathsf{x}(p_1^{(1)}) - \mathsf{x}(p_2^{(1)}), \Delta\textsc{distSmall}\right)$	
$\textsc{rightOfEnd}(p_1,p_2) \triangleq \mathsf{tempCoher}(p_2) + \mathsf{distGreaterThan}\left(\mathsf{x}(p_1^{(T)}) - \mathsf{x}(p_2^{(T)}), \Delta\textsc{distSmall}\right)$	
$\textsc{onTopOfStart}(p_1,p_2) \triangleq \mathsf{tempCoher}(p_2) + \mathsf{distGreaterThan}\left(\mathsf{y}(p_1^{(1)}) - \mathsf{y}(p_2^{(1)}), -2\Delta\textsc{distLarge}\right) + \mathsf{distLessThan}\left(\mathsf{y}(p_1^{(1)}) - \mathsf{y}(p_2^{(1)}), 0\right) + \mathsf{distLessThan}\left(\left\lvert\mathsf{x}(p_1^{(1)}) - \mathsf{x}(p_2^{(1)})\right\rvert, 2\Delta\textsc{distSmall}\right)$	
$\textsc{onTopOfEnd}(p_1,p_2) \triangleq \mathsf{tempCoher}(p_2) + \mathsf{distGreaterThan}\left(\mathsf{y}(p_1^{(T)}) - \mathsf{y}(p_2^{(T)}), -2\Delta\textsc{distLarge}\right) + \mathsf{distLessThan}\left(\mathsf{y}(p_1^{(T)}) - \mathsf{y}(p_2^{(T)}), 0\right) + \mathsf{distLessThan}\left(\left\lvert\mathsf{x}(p_1^{(T)}) - \mathsf{x}(p_2^{(T)})\right\rvert, 2\Delta\textsc{distSmall}\right)$	
$\textsc{nearStart}(p_1,p_2) \triangleq \mathsf{tempCoher}(p_2) + \mathsf{distLessThan}\left(\mathsf{dist}(p_1^{(1)},p_2^{(1)}), 2\Delta\textsc{distSmall}\right)$	
$\textsc{nearEnd}(p_1,p_2) \triangleq \mathsf{tempCoher}(p_2) + \mathsf{distLessThan}\left(\mathsf{dist}(p_1^{(T)},p_2^{(T)}), 2\Delta\textsc{distSmall}\right)$	
$\textsc{inStart}(p_1,p_2) \triangleq \mathsf{tempCoher}(p_2) + \textsc{nearStart}(p_1,p_2) + \mathsf{smaller}(p_1^{(1)},p_2^{(1)})$	
$\textsc{inEnd}(p_1,p_2) \triangleq \mathsf{tempCoher}(p_2) + \textsc{nearEnd}(p_1,p_2) + \mathsf{smaller}(p_1^{(T)},p_2^{(T)})$	
$\textsc{belowStart}(p_1,p_2) \triangleq \mathsf{tempCoher}(p_2) + \mathsf{distGreaterThan}\left(\mathsf{y}(p_1^{(1)}) - \mathsf{y}(p_2^{(1)}), \Delta\textsc{distSmall}\right)$	
$\textsc{belowEnd}(p_1,p_2) \triangleq \mathsf{tempCoher}(p_2) + \mathsf{distGreaterThan}\left(\mathsf{y}(p_1^{(T)}) - \mathsf{y}(p_2^{(T)}), \Delta\textsc{distSmall}\right)$	
$\textsc{aboveStart}(p_1,p_2) \triangleq \mathsf{tempCoher}(p_2) + \mathsf{distLessThan}\left(\mathsf{y}(p_1^{(1)}) - \mathsf{y}(p_2^{(1)}), -\Delta\textsc{distSmall}\right)$	
$\textsc{aboveEnd}(p_1,p_2) \triangleq \mathsf{tempCoher}(p_2) + \mathsf{distLessThan}\left(\mathsf{y}(p_1^{(T)}) - \mathsf{y}(p_2^{(T)}), -\Delta\textsc{distSmall}\right)$	
$\textsc{over}(p_1,p_2) \triangleq \mathsf{tempCoher}(p_2) + \max_t \begin{pmatrix} \mathsf{distLessThan}\left(\mathsf{y}(p_1^{(t)}) - \mathsf{y}(p_2^{(t)}), -\Delta\textsc{distSmall}\right) \\ \mathsf{distLessThan}\left(\left\lvert\mathsf{x}(p_1^{(t)}) - \mathsf{x}(p_2^{(t)})\right\rvert, \Delta\textsc{distLarge}\right) \end{pmatrix}$	

Method

1. generate proposals with EdgeBoxes (Zitnick et al. ECCV 2014) and MCG (Arbelaez et al. CVPR 2014)

Method

1. generate proposals with EdgeBoxes (Zitnick et al. ECCV 2014) and MCG (Arbelaez et al. CVPR 2014)
2. sample MOVING and STATIONARY proposals from sampled frames

Method

1. generate proposals with EdgeBoxes (Zitnick et al. ECCV 2014) and MCG (Arbelaez et al. CVPR 2014)
2. sample MOVING and STATIONARY proposals from sampled frames
3. track sampled MOVING proposal with CamShift (Bradski 1998) in HSV and STATIONARY proposals with MeanShift (Comaniciu et al. 2000) in RGB forward and backward over whole clip

Method

1. generate proposals with EdgeBoxes (Zitnick et al. ECCV 2014) and MCG (Arbelaez et al. CVPR 2014)
2. sample MOVING and STATIONARY proposals from sampled frames
3. track sampled MOVING proposal with CamShift (Bradski 1998) in HSV and STATIONARY proposals with MeanShift (Comaniciu et al. 2000) in RGB forward and backward over whole clip
4. rotate proposal multiples of 90°

Method

1. generate proposals with EdgeBoxes (Zitnick et al. ECCV 2014) and MCG (Arbelaez et al. CVPR 2014)
2. sample MOVING and STATIONARY proposals from sampled frames
3. track sampled MOVING proposal with CamShift (Bradski 1998) in HSV and STATIONARY proposals with MeanShift (Comaniciu et al. 2000) in RGB forward and backward over whole clip
4. rotate proposal multiples of 90°
5. graphical model

Method

1. generate proposals with EdgeBoxes (Zitnick et al. ECCV 2014) and MCG (Arbelaez et al. CVPR 2014)
2. sample MOVING and STATIONARY proposals from sampled frames
3. track sampled MOVING proposal with CamShift (Bradski 1998) in HSV and STATIONARY proposals with MeanShift (Comaniciu et al. 2000) in RGB forward and backward over whole clip
4. rotate proposal multiples of 90°
5. graphical model
    - object instances appearing in the sentences as vertices

Method

1. generate proposals with EdgeBoxes (Zitnick et al. ECCV 2014) and MCG (Arbelaez et al. CVPR 2014)
2. sample MOVING and STATIONARY proposals from sampled frames
3. track sampled MOVING proposal with CamShift (Bradski 1998) in HSV and STATIONARY proposals with MeanShift (Comaniciu et al. 2000) in RGB forward and backward over whole clip
4. rotate proposal multiples of 90°
5. graphical model
   - object instances appearing in the sentences as vertices
   - tracks in the video associated with the sentence as vertex labels

Method

1. generate proposals with EdgeBoxes (Zitnick et al. ECCV 2014) and MCG (Arbelaez et al. CVPR 2014)
2. sample MOVING and STATIONARY proposals from sampled frames
3. track sampled MOVING proposal with CamShift (Bradski 1998) in HSV and STATIONARY proposals with MeanShift (Comaniciu et al. 2000) in RGB forward and backward over whole clip
4. rotate proposal multiples of 90°
5. graphical model
   - object instances appearing in the sentences as vertices
   - tracks in the video associated with the sentence as vertex labels
   - edge between two object instances in the same sentence
     clique for all object instances for the same noun

Method

1. generate proposals with EdgeBoxes (Zitnick et al. ECCV 2014) and MCG (Arbelaez et al. CVPR 2014)
2. sample MOVING and STATIONARY proposals from sampled frames
3. track sampled MOVING proposal with CamShift (Bradski 1998) in HSV and STATIONARY proposals with MeanShift (Comaniciu et al. 2000) in RGB forward and backward over whole clip
4. rotate proposal multiples of 90°
5. graphical model
   - object instances appearing in the sentences as vertices
   - tracks in the video associated with the sentence as vertex labels
   - edge between two object instances in the same sentence
     clique for all object instances for the same noun
   - unary predicate score from sentences as vertex score

Method

1. generate proposals with EdgeBoxes (Zitnick et al. ECCV 2014) and MCG (Arbelaez et al. CVPR 2014)
2. sample MOVING and STATIONARY proposals from sampled frames
3. track sampled MOVING proposal with CamShift (Bradski 1998) in HSV and STATIONARY proposals with MeanShift (Comaniciu et al. 2000) in RGB forward and backward over whole clip
4. rotate proposal multiples of 90°
5. graphical model
    - object instances appearing in the sentences as vertices
    - tracks in the video associated with the sentence as vertex labels
    - edge between two object instances in the same sentence
      clique for all object instances for the same noun
    - unary predicate score from sentences as vertex score
    - binary predicate score from sentences as edge score
      similarity score as edge score

Method

1. generate proposals with EdgeBoxes (Zitnick et al. ECCV 2014) and MCG (Arbelaez et al. CVPR 2014)
2. sample MOVING and STATIONARY proposals from sampled frames
3. track sampled MOVING proposal with CamShift (Bradski 1998) in HSV and STATIONARY proposals with MeanShift (Comaniciu et al. 2000) in RGB forward and backward over whole clip
4. rotate proposal multiples of 90°
5. graphical model
   - object instances appearing in the sentences as vertices
   - tracks in the video associated with the sentence as vertex labels
   - edge between two object instances in the same sentence
     clique for all object instances for the same noun
   - unary predicate score from sentences as vertex score
   - binary predicate score from sentences as edge score
     similarity score as edge score
     - χ^2 of PHOW (Bosch et al. ICCV 2007) and L_2 HOG (Dalal & Triggs CVPR 2005) to measure similarity

Method


OUTFROM(cabbage0,bowl0)
bowl0 <-> bowl1
cabbage0 <-> cabbage1
AWAYFROM(squash0,bowl1)
DOWN(cleaner0)
NEAR(cleaner0,cabbage1)

Four Variants

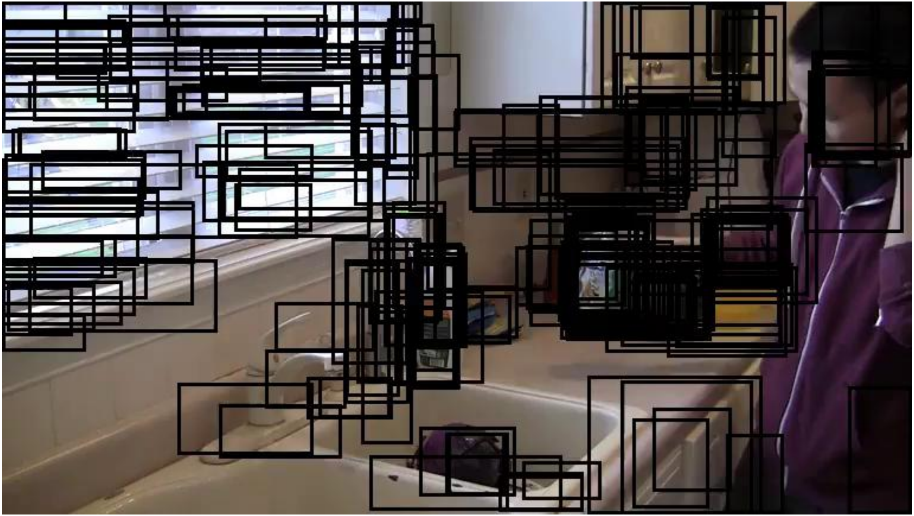
	SIM (variant 1)	**FLOW** (variant 2)	**SIM+FLOW** (variant 3)	**SENT** (variant 4)	**SIM+SENT** (our full method)
Similarity score?	yes	no	yes	no	yes
Sentence score?	no	partial	partial	yes	yes

partial: motion and temporal coherence but no other components of sentence semantics

Our Predicates

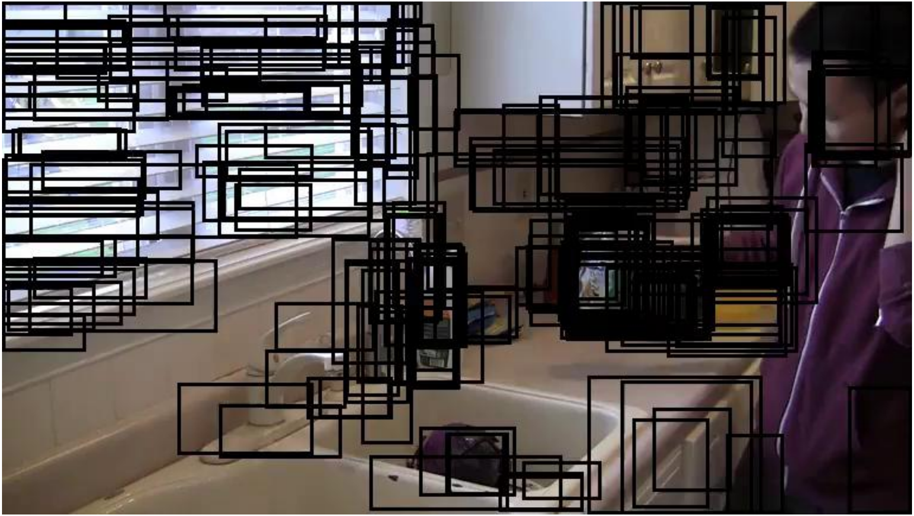
Predicates	Constants
$\textsc{move}(p) \triangleq \textsf{medFlMg}(p)$	$\Delta\textsc{distLarge} \triangleq 0.25$
$\textsc{moveUp}(p) \triangleq \textsc{move}(p) + \textsf{distLessThan}\left(\textsf{y}(p^{(T)}) - \textsf{y}(p^{(1)}), -\Delta\textsc{distLarge}\right)$	$\Delta\textsc{distSmall} \triangleq 0.05$
$\textsc{moveDown}(p) \triangleq \textsc{move}(p) + \textsf{distGreaterThan}\left(\textsf{y}(p^{(T)}) - \textsf{y}(p^{(1)}), \Delta\textsc{distLarge}\right)$	$\Delta\textsc{angle} \triangleq \pi/2$
$\textsc{moveHorizontal}(p) \triangleq \textsc{move}(p) + \textsf{distGreaterThan}\left(\left\|\textsf{x}(p^{(T)}) - \textsf{x}(p^{(1)})\right\|, \Delta\textsc{distLarge}\right)$	
$\textsc{moveLeftwards}(p) \triangleq \textsc{move}(p) + \textsf{distLessThan}\left(\textsf{x}(p^{(T)}) - \textsf{x}(p^{(1)}), -\Delta\textsc{distLarge}\right)$	
$\textsc{moveRightwards}(p) \triangleq \textsc{move}(p) + \textsf{distGreaterThan}\left(\textsf{x}(p^{(T)}) - \textsf{x}(p^{(1)}), \Delta\textsc{distLarge}\right)$	
$\textsc{rotate}(p) \triangleq \textsc{move}(p) + \max_t \textsf{hasRotation}\left(\textsf{rotAngle}(p^{(t)}), \Delta\textsc{angle}\right)$	
$\textsc{towards}(p_1,p_2) \triangleq \textsc{move}(p_1) + \textsf{distLessThan}\left(\textsf{dist}(p_1^{(T)},p_2^{(T)}) - \textsf{dist}(p_1^{(1)},p_2^{(1)}), -\Delta\textsc{distLarge}\right)$	
$\textsc{awayFrom}(p_1,p_2) \triangleq \textsc{move}(p_1) + \textsf{distGreaterThan}\left(\textsf{dist}(p_1^{(T)},p_2^{(T)}) - \textsf{dist}(p_1^{(1)},p_2^{(1)}), \Delta\textsc{distLarge}\right)$	
$\textsc{leftOfStart}(p_1,p_2) \triangleq \textsf{tempCoher}(p_2) + \textsf{distLessThan}\left(\textsf{x}(p_1^{(1)}) - \textsf{x}(p_2^{(1)}), -\Delta\textsc{distSmall}\right)$	
$\textsc{leftOfEnd}(p_1,p_2) \triangleq \textsf{tempCoher}(p_2) + \textsf{distLessThan}\left(\textsf{x}(p_1^{(T)}) - \textsf{x}(p_2^{(T)}), -\Delta\textsc{distSmall}\right)$	
$\textsc{rightOfStart}(p_1,p_2) \triangleq \textsf{tempCoher}(p_2) + \textsf{distGreaterThan}\left(\textsf{x}(p_1^{(1)}) - \textsf{x}(p_2^{(1)}), \Delta\textsc{distSmall}\right)$	
$\textsc{rightOfEnd}(p_1,p_2) \triangleq \textsf{tempCoher}(p_2) + \textsf{distGreaterThan}\left(\textsf{x}(p_1^{(T)}) - \textsf{x}(p_2^{(T)}), \Delta\textsc{distSmall}\right)$	
$\textsc{onTopOfStart}(p_1,p_2) \triangleq \textsf{tempCoher}(p_2) + \textsf{distGreaterThan}\left(\textsf{y}(p_1^{(1)}) - \textsf{y}(p_2^{(1)}), -2\Delta\textsc{distLarge}\right) + \textsf{distLessThan}\left(\textsf{y}(p_1^{(1)}) - \textsf{y}(p_2^{(1)}), 0\right) + \textsf{distLessThan}\left(\left\|\textsf{x}(p_1^{(1)}) - \textsf{x}(p_2^{(1)})\right\|, 2\Delta\textsc{distSmall}\right)$	
$\textsc{onTopOfEnd}(p_1,p_2) \triangleq \textsf{tempCoher}(p_2) + \textsf{distGreaterThan}\left(\textsf{y}(p_1^{(T)}) - \textsf{y}(p_2^{(T)}), -2\Delta\textsc{distLarge}\right) + \textsf{distLessThan}\left(\textsf{y}(p_1^{(T)}) - \textsf{y}(p_2^{(T)}), 0\right) + \textsf{distLessThan}\left(\left\|\textsf{x}(p_1^{(T)}) - \textsf{x}(p_2^{(T)})\right\|, 2\Delta\textsc{distSmall}\right)$	
$\textsc{nearStart}(p_1,p_2) \triangleq \textsf{tempCoher}(p_2) + \textsf{distLessThan}\left(\textsf{dist}(p_1^{(1)},p_2^{(1)}), 2\Delta\textsc{distSmall}\right)$	
$\textsc{nearEnd}(p_1,p_2) \triangleq \textsf{tempCoher}(p_2) + \textsf{distLessThan}\left(\textsf{dist}(p_1^{(T)},p_2^{(T)}), 2\Delta\textsc{distSmall}\right)$	
$\textsc{inStart}(p_1,p_2) \triangleq \textsf{tempCoher}(p_2) + \textsc{nearStart}(p_1,p_2) + \textsf{smaller}(p_1^{(1)},p_2^{(1)})$	
$\textsc{inEnd}(p_1,p_2) \triangleq \textsf{tempCoher}(p_2) + \textsc{nearEnd}(p_1,p_2) + \textsf{smaller}(p_1^{(T)},p_2^{(T)})$	
$\textsc{belowStart}(p_1,p_2) \triangleq \textsf{tempCoher}(p_2) + \textsf{distGreaterThan}\left(\textsf{y}(p_1^{(1)}) - \textsf{y}(p_2^{(1)}), \Delta\textsc{distSmall}\right)$	
$\textsc{belowEnd}(p_1,p_2) \triangleq \textsf{tempCoher}(p_2) + \textsf{distGreaterThan}\left(\textsf{y}(p_1^{(T)}) - \textsf{y}(p_2^{(T)}), \Delta\textsc{distSmall}\right)$	
$\textsc{aboveStart}(p_1,p_2) \triangleq \textsf{tempCoher}(p_2) + \textsf{distLessThan}\left(\textsf{y}(p_1^{(1)}) - \textsf{y}(p_2^{(1)}), -\Delta\textsc{distSmall}\right)$	
$\textsc{aboveEnd}(p_1,p_2) \triangleq \textsf{tempCoher}(p_2) + \textsf{distLessThan}\left(\textsf{y}(p_1^{(T)}) - \textsf{y}(p_2^{(T)}), -\Delta\textsc{distSmall}\right)$	
$\textsc{over}(p_1,p_2) \triangleq \textsf{tempCoher}(p_2) + \max_t \begin{pmatrix} \textsf{distLessThan}\left(\textsf{y}(p_1^{(t)}) - \textsf{y}(p_2^{(t)}), -\Delta\textsc{distSmall}\right) \\ \textsf{distLessThan}\left(\left\|\textsf{x}(p_1^{(t)}) - \textsf{x}(p_2^{(t)})\right\|, \Delta\textsc{distLarge}\right) \end{pmatrix}$	

Proposals



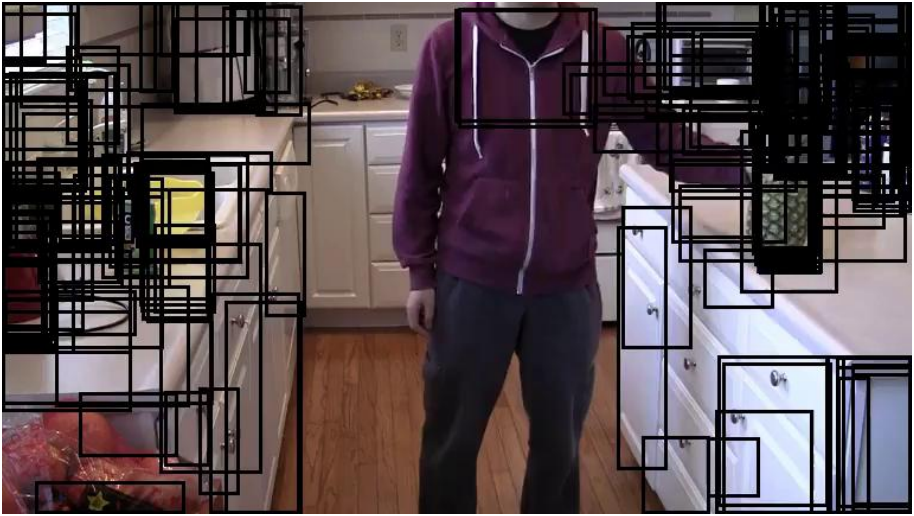
The person put the cleaner into the sink near the cabbage.

Proposals



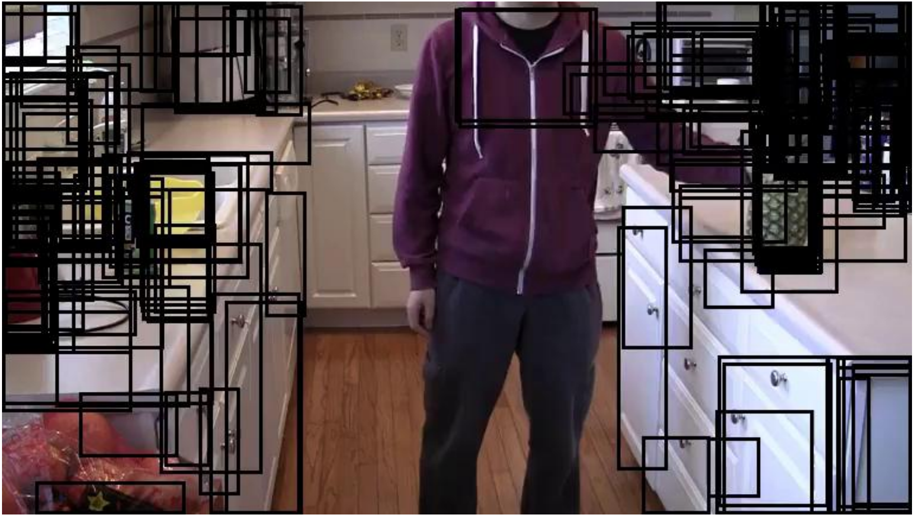
The person put the cleaner into the sink near the cabbage.

Proposals



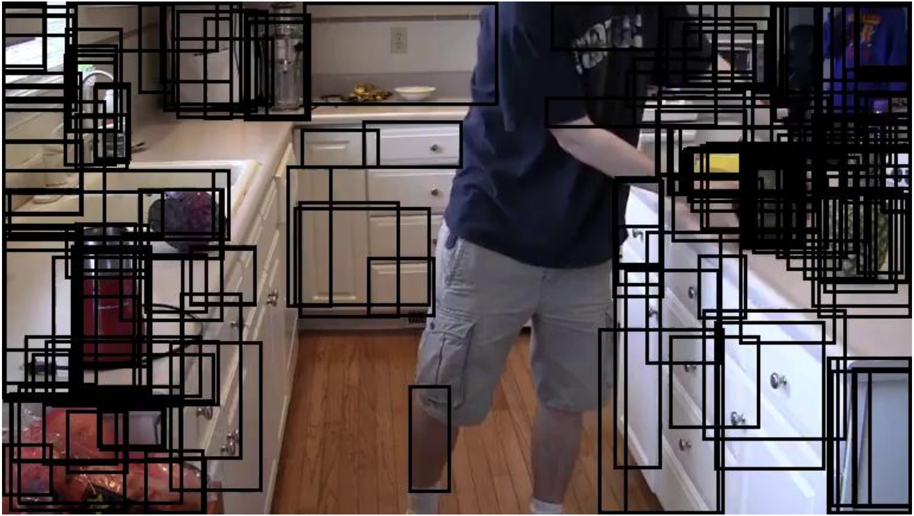
The person carried the pineapple towards the cleaner.

Proposals



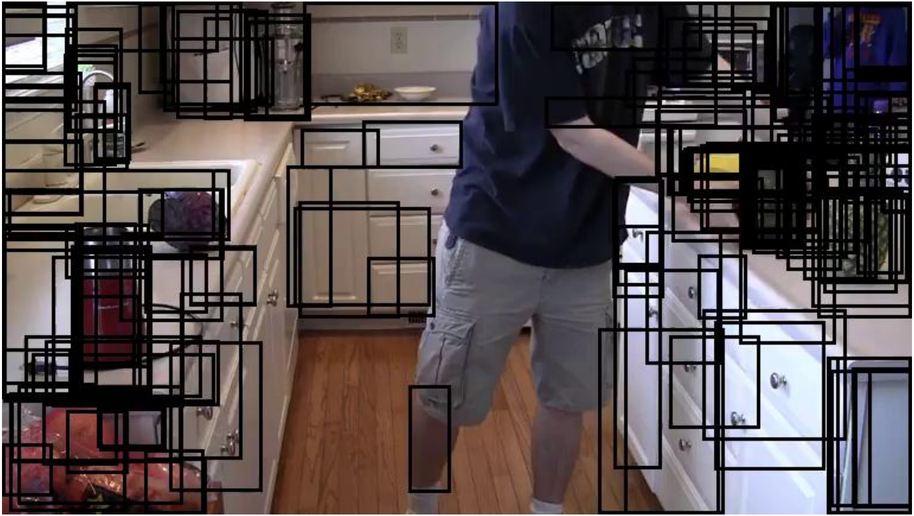
The person carried the pineapple towards the cleaner.

Proposals



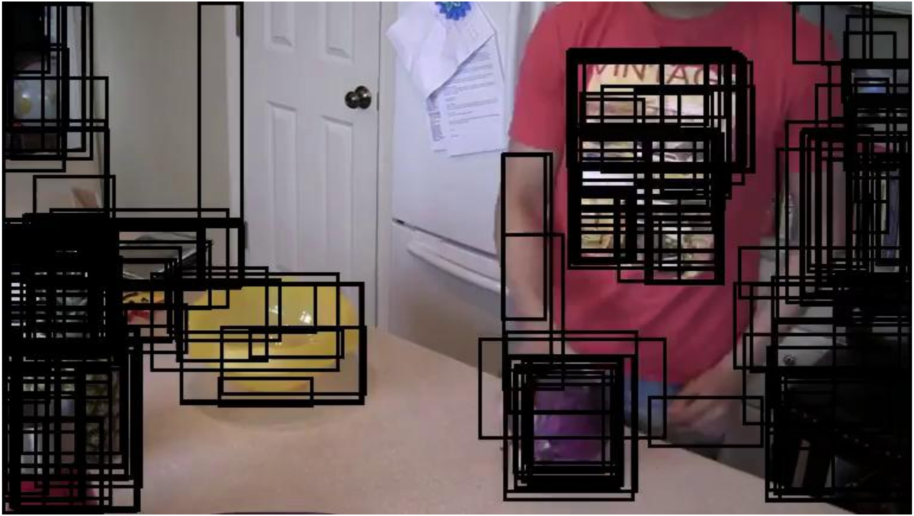
The person took the squash away from the pineapple and put it near the coffee.

Proposals



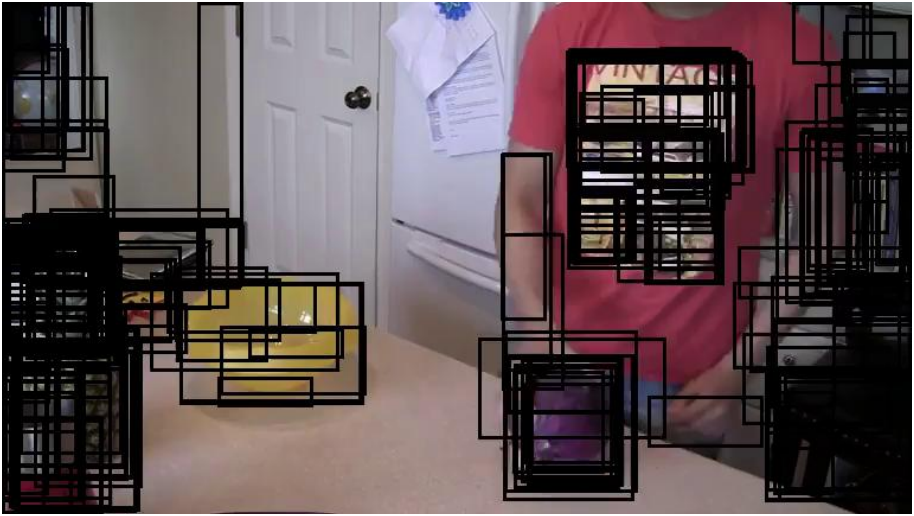
The person took the squash away from the pineapple and put it near the coffee.

Proposals



The person put the cabbage into the bowl.

Proposals



The person put the cabbage into the bowl.

The person put the cleaner into the sink near the cabbage.

SIM

The person put the cleaner into the sink near the cabbage.

The person carried the pineapple towards the cleaner.

SIM

The person carried the pineapple towards the cleaner.

SIM

The person took the squash away from the pineapple and put it near the coffee.

The person took the squash away from the pineapple and put it near the coffee.

SIM

The person put the cabbage into the bowl.

The person put the cabbage into the bowl.

The person put the cleaner into the sink near the cabbage.

FLOW

The person put the cleaner into the sink near the cabbage.

FLOW

The person carried the pineapple towards the cleaner.

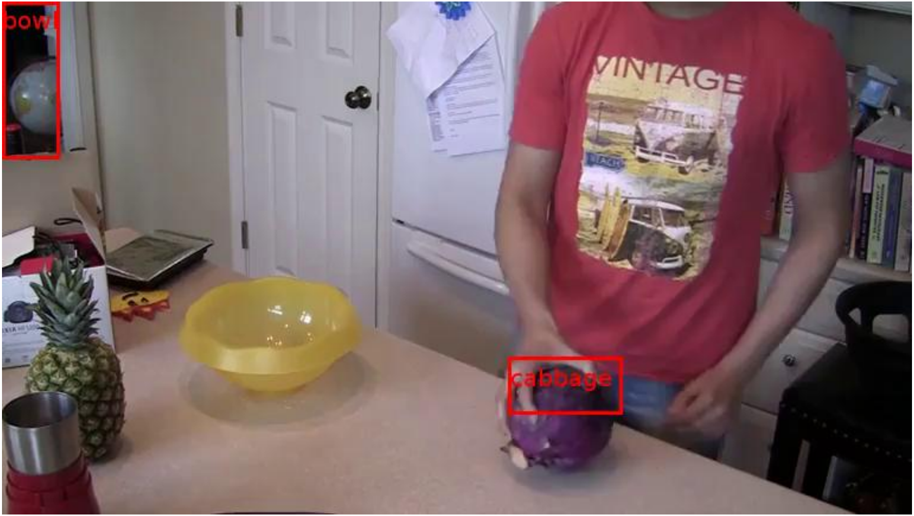
FLOW

The person carried the pineapple towards the cleaner.

The person took the squash away from the pineapple and put it near the coffee.

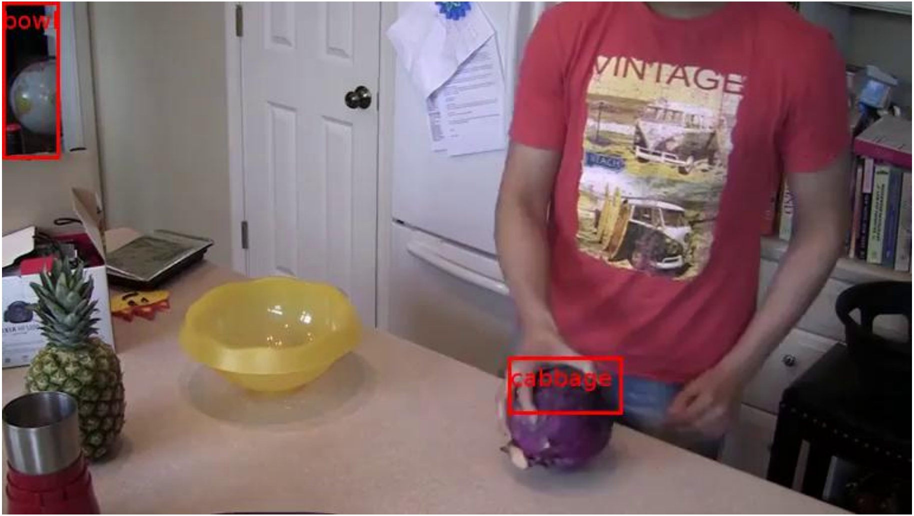
FLOW

The person took the squash away from the pineapple and put it near the coffee.



The person put the cabbage into the bowl.

FLOW



The person put the cabbage into the bowl.

SIM+FLOW

The person put the cleaner into the sink near the cabbage.

The person put the cleaner into the sink near the cabbage.

SIM+FLOW

The person carried the pineapple towards the cleaner.

SIM+FLOW

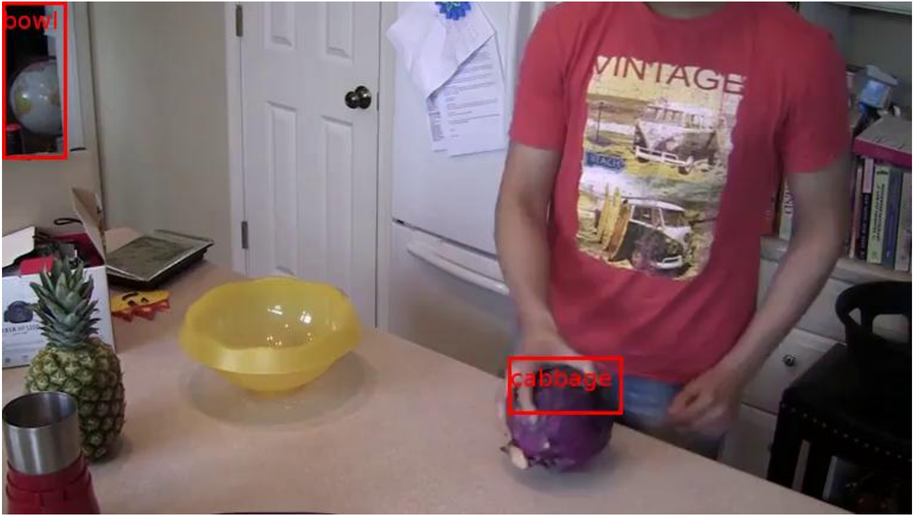
The person carried the pineapple towards the cleaner.

SIM+FLOW

The person took the squash away from the pineapple and put it near the coffee.

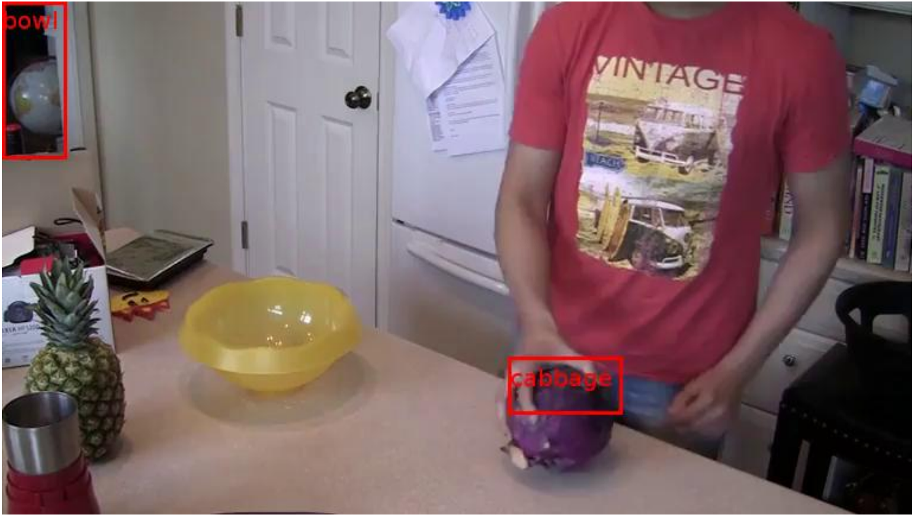
SIM+FLOW

The person took the squash away from the pineapple and put it near the coffee.



The person put the cabbage into the bowl.

SIM+FLOW



The person put the cabbage into the bowl.

SENT

The person put the cleaner into the sink near the cabbage.

SENT

The person put the cleaner into the sink near the cabbage.

SENT

The person carried the pineapple towards the cleaner.

SENT

The person carried the pineapple towards the cleaner.

SENT

The person took the squash away from the pineapple and put it near the coffee.

SENT

The person took the squash away from the pineapple and put it near the coffee.

SENT

The person put the cabbage into the bowl.

SENT

The person put the cabbage into the bowl.

The person put the cleaner into the sink near the cabbage.

SIM+SENT

The person put the cleaner into the sink near the cabbage.

SIM+SENT

The person carried the pineapple towards the cleaner.

SIM+SENT

The person carried the pineapple towards the cleaner.

SIM+SENT

The person took the squash away from the pineapple and put it near the coffee.

SIM+SENT

The person took the squash away from the pineapple and put it near the coffee.

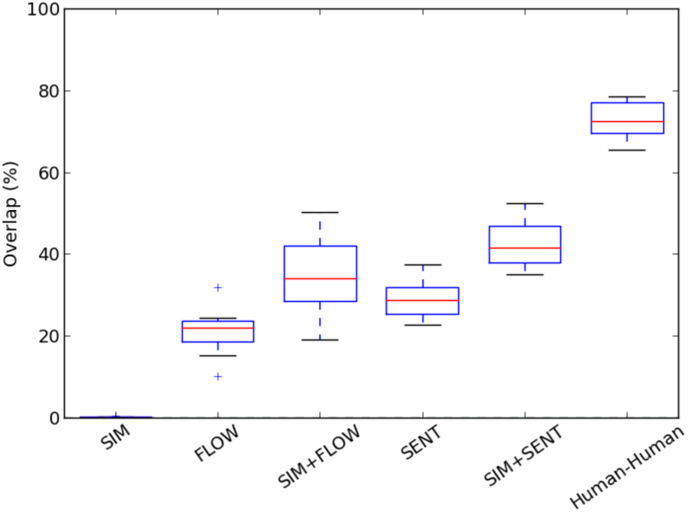
SIM+SENT

The person put the cabbage into the bowl.

SIM+SENT

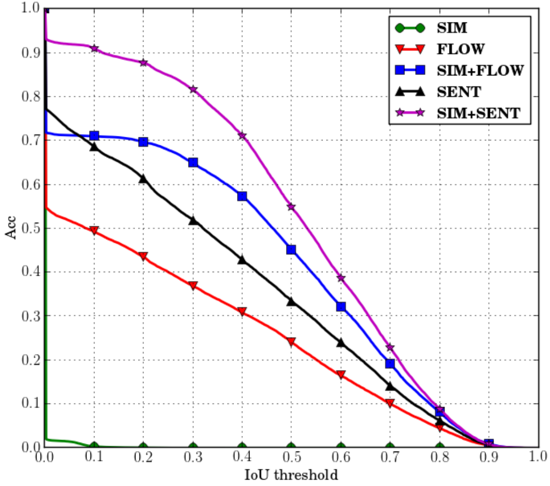
The person put the cabbage into the bowl.

IoU Scores



100
80
60
40
20
0
Overlap (%)
SIM
FLOW
SIM+FLOW
SENT
SIM+SENT
Human-Human

Codetection Accuracy



More Examples

More Examples

Outline

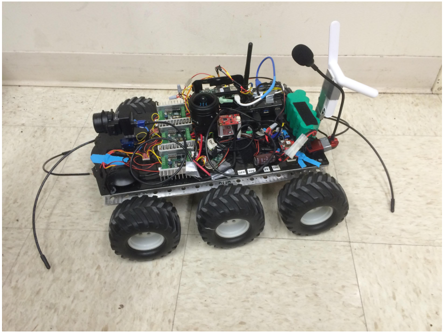
1. Verbs, Arguments, and Predication in the Human Brain
   - Experiment 1: hollywood2-text-speech
   - Experiment 2: compositionality-noninterleaved
   - Experiment 3: predication
2. Sentence Directed Video Object Codetection
3. Driving Under the Influence (of Language)
   - Grounding Language Semantics in Robotics
   - Object Codetection from Mobile Robot Video
4. Playing Checkers from English

Joint work

Daniel Paul Barrett   Scott Alan Bronikowski   Haonan Yu

Our Custom Mobile Robot

- IMU (3-axis accelerometers, gyros, and magnetometers)
- GPS
- 6 independently controllable wheel motors
- 2 shaft encoders with Teensy controller
- Gumstix Overo FireSTORM + Summit running Linux
- Bluetooth, WiFi, and 4G LTE
- front and rear bump sensors
- ultrasonic rangefinder
- pan-tilt front-facing camera (Point Grey)
- omnidirectional camera (Point Grey)
- audio input and output
- touchscreen
- Logitech Wireless Gamepad
- custom firmware on IMU and Teensy
- synchronized timestamped logging of sensor and control data



Outline

1. Verbs, Arguments, and Predication in the Human Brain
   - Experiment 1: hollywood2-text-speech
   - Experiment 2: compositionality-noninterleaved
   - Experiment 3: predication
2. Sentence Directed Video Object Codetection
3. Driving Under the Influence (of Language)
   - Grounding Language Semantics in Robotics
   - Object Codetection from Mobile Robot Video
4. Playing Checkers from English

Grounding Language Semantics in Robotics

$$\mathcal{R} : (\mathbf{s}, \mathbf{p}, \mathbf{f}, \Lambda) \mapsto \tau$$

- $\mathbf{s}$: sentence
- $\mathbf{p}$: path
- $\mathbf{f}$: floorplan
- Λ: lexicon
- τ: score

Three Uses of the Unified Scoring Function

Three Uses of the Unified Scoring Function

- **Language Acquisition**: sentence $\times$ path $\rightarrow$ lexicon

$$\arg\max_{\Lambda} \sum_{i=1} \mathcal{R}(\mathbf{s}_i, \mathbf{p}_i, \mathbf{f}_i, \Lambda)$$

Three Uses of the Unified Scoring Function

- **Language Acquisition**: sentence $\times$ path $\rightarrow$ lexicon

$$\arg\max_{\Lambda} \sum_{i=1} \mathcal{R}(\mathbf{s}_i, \mathbf{p}_i, \mathbf{f}_i, \Lambda)$$

- **Language Generation**: path $\times$ lexicon $\rightarrow$ sentence

$$\arg\max_{\mathbf{s}} \mathcal{R}(\mathbf{s}, \mathbf{p}, \mathbf{f}, \Lambda)$$

Three Uses of the Unified Scoring Function

- **Language Acquisition**: sentence $\times$ path $\rightarrow$ lexicon

$$\arg\max_{\Lambda} \sum_{i=1} \mathcal{R}(\mathbf{s}_i, \mathbf{p}_i, \mathbf{f}_i, \Lambda)$$

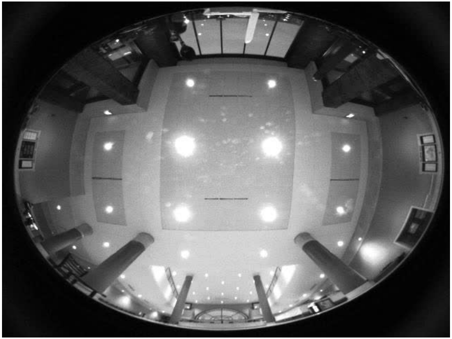
- **Language Generation**: path $\times$ lexicon $\rightarrow$ sentence

$$\arg\max_{\mathbf{s}} \mathcal{R}(\mathbf{s}, \mathbf{p}, \mathbf{f}, \Lambda)$$

- **Language Comprehension**: sentence $\times$ lexicon $\rightarrow$ path

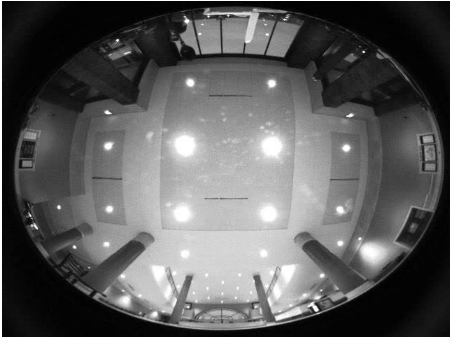
$$\arg\max_{\mathbf{p}} \mathcal{R}(\mathbf{s}, \mathbf{p}, \mathbf{f}, \Lambda)$$

Language Acquisition



The robot went behind the cone and then turned around and went further behind the cone to the right of the chair.

Language Acquisition



The robot went behind the cone and then turned around and went further behind the cone to the right of the chair.

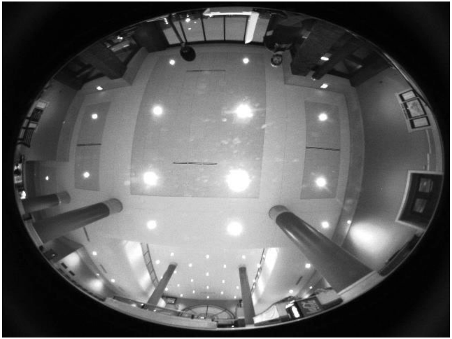
Language Acquisition

The robot went behind the cone and then turned around and went further behind the cone to the right of the chair.

Language Acquisition

The robot went behind the cone and then turned around and went further behind the cone to the right of the chair.

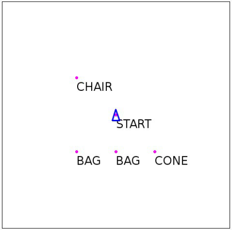
Language Acquisition



The robot went behind the cone and then turned around and went further behind the cone to the right of the chair.

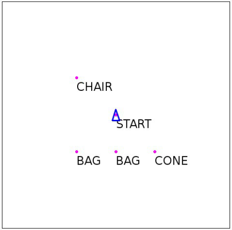
Language Acquisition

input: *The robot went behind the cone and then turned around and went further behind the cone to the right of the chair.*



Language Acquisition

input: *The robot went behind the cone and then turned around and went further behind the cone to the right of the chair.*



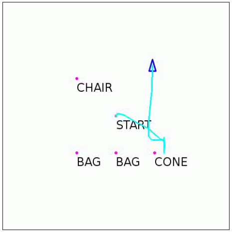
Language Acquisition

input: *The robot went behind the cone and then turned around and went further behind the cone to the right of the chair.*

Language Acquisition

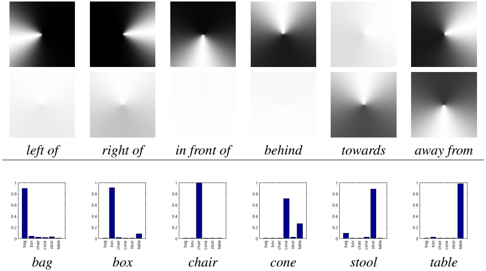
input: *The robot went behind the cone and then turned around and went further behind the cone to the right of the chair.*

input: *The robot went behind the cone and then turned around and went further behind the cone to the right of the chair.*

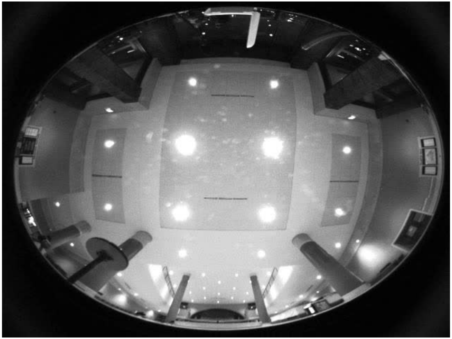


. . . **plus 599 more**

output:

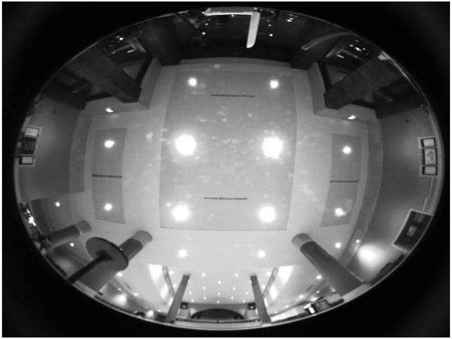


Language Generation



The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.

Language Generation



The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.

Language Generation

The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.

Language Generation

The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.

Language Generation

The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.

Language Generation

The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.

Language Generation

The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.

Language Generation

The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.

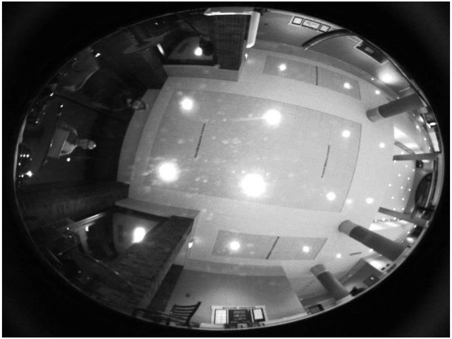
Language Generation

The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.

Language Generation

The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.

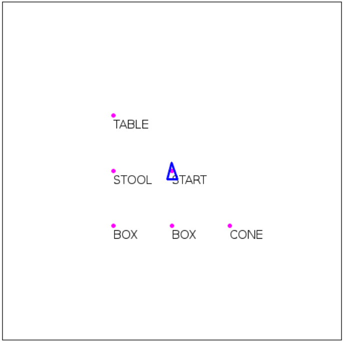
Language Generation

The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.

Language Generation

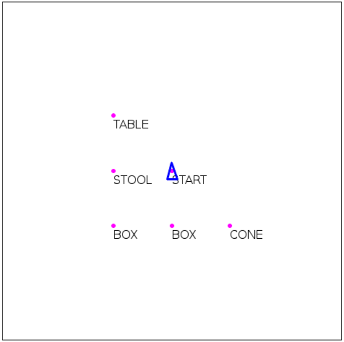
input:



output: *The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.*

Language Generation

input:



output: *The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.*

Language Generation

input:

output: *The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.*

Language Generation

input:

output: *The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.*

input:

output: *The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.*

Language Generation

input:

output: *The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.*

Language Generation

input:

output: *The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.*

Language Generation

input:

output: *The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.*

Language Generation

input:

output: *The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.*

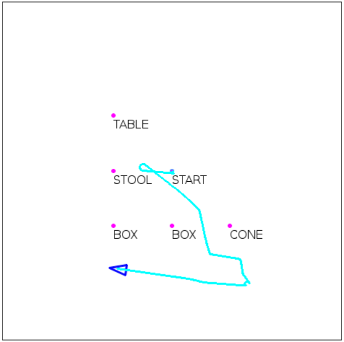
Language Generation

input:

output: *The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.*

Language Generation

input:

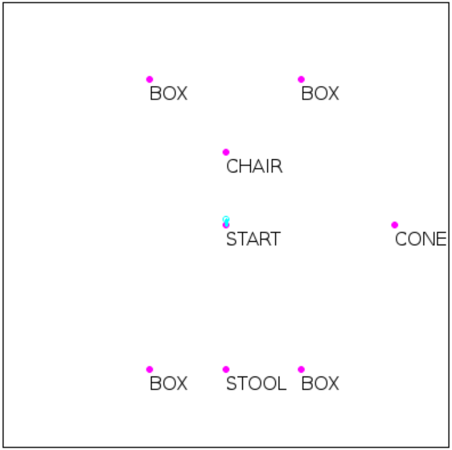


output: *The robot went behind the box which is right of the box then went right of the stool then went right of the box which is right of the box then went left of the cone then went in front of the cone then went away from the cone then went in front of the cone then went in front of the box which is right of the box then went in front of the box which is left of the box.*

Language Comprehension

input: *The robot went away from the cone then went behind the box which is right of the chair and which is behind the cone then went towards the stool.*

output:

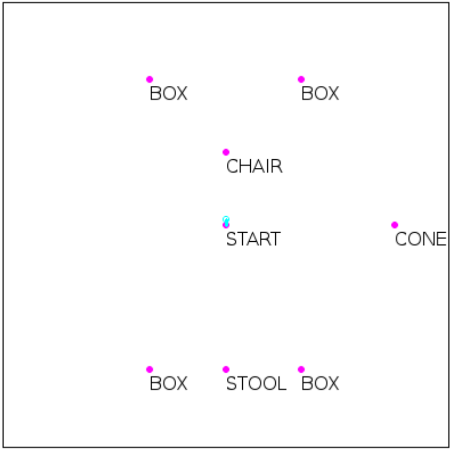


- Determine path waypoints that satisfy sentence
- Add intermediate points to avoid obstacles

Language Comprehension

input: *The robot went away from the cone then went behind the box which is right of the chair and which is behind the cone then went towards the stool.*

output:

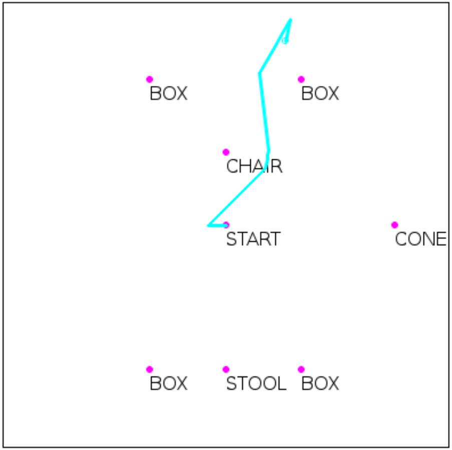


- Determine path waypoints that satisfy sentence
- Add intermediate points to avoid obstacles

Language Comprehension

input: *The robot went away from the cone then went behind the box which is right of the chair and which is behind the cone then went towards the stool.*

output:

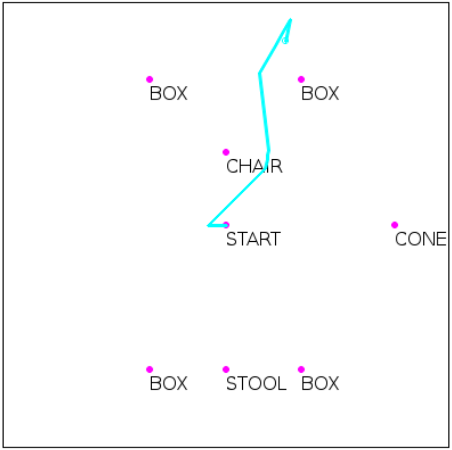


- Determine path waypoints that satisfy sentence
- Add intermediate points to avoid obstacles

Language Comprehension

input: *The robot went away from the cone then went behind the box which is right of the chair and which is behind the cone then went towards the stool.*

output:

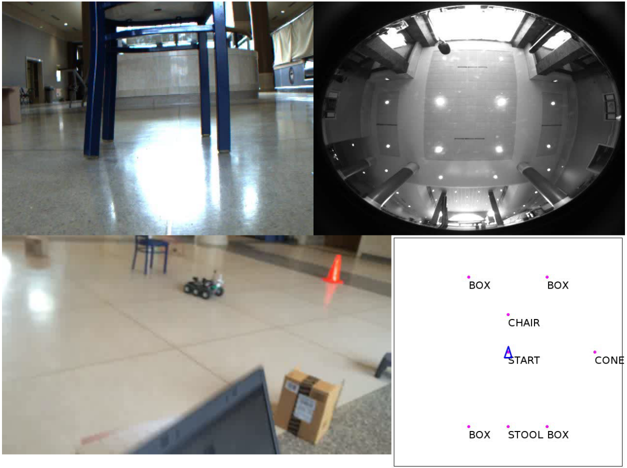


- Determine path waypoints that satisfy sentence
- Add intermediate points to avoid obstacles

Language Comprehension

The Effect of Different Prepositions (1)

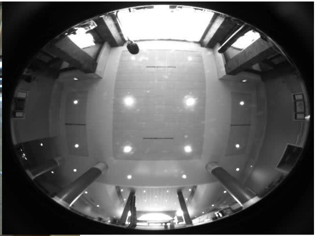
The robot went away from the cone then went behind the box which is **right of** *the chair and which is* **behind** *the cone then went towards the stool.*

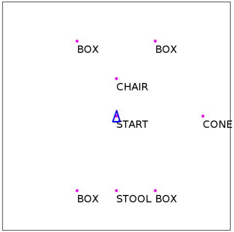


Language Comprehension

The Effect of Different Prepositions (1)

The robot went away from the cone then went behind the box which is **<u>right of</u>** *the chair and which is* **<u>behind</u>** *the cone then went towards the stool.*





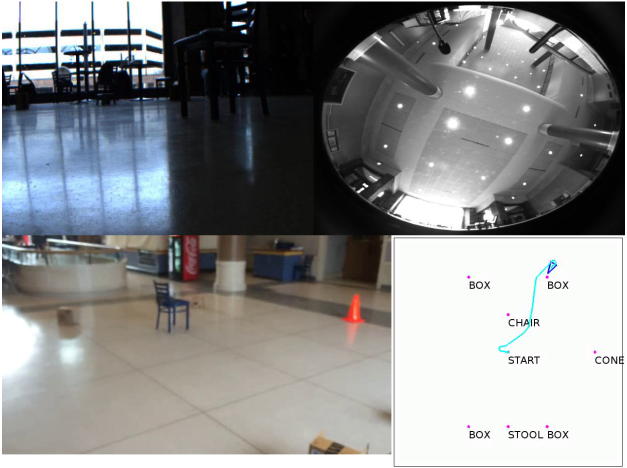
The robot went away from the cone then went behind the box which is **right of** *the chair and which is* **behind** *the cone then went towards the stool.*

The robot went away from the cone then went behind the box which is **right of** *the chair and which is* **behind** *the cone then went towards the stool.*

Language Comprehension

The Effect of Different Prepositions (1)

The robot went away from the cone then went behind the box which is **right of** *the chair and which is* **behind** *the cone then went towards the stool.*



Language Comprehension

The Effect of Different Prepositions (2)

The robot went away from the cone then went behind the box which is **right of** *the chair and which is* **in front of** *the cone then went towards the stool.*

The robot went away from the cone then went behind the box which is ***right of*** *the chair and which is* ***in front of*** *the cone then went towards the stool.*

The robot went away from the cone then went behind the box which is **right of** *the chair and which is* **in front of** *the cone then went towards the stool.*

Language Comprehension

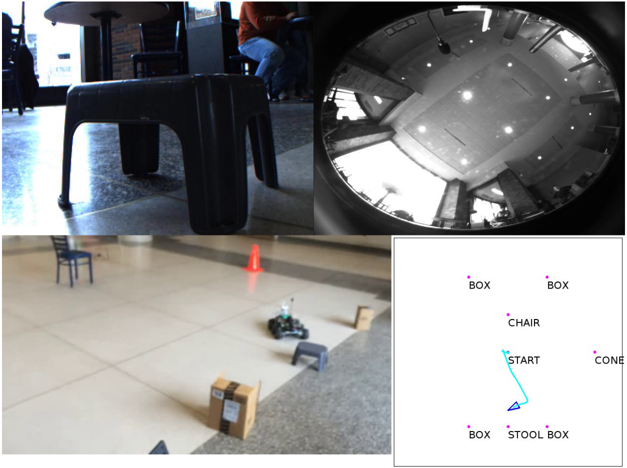
The Effect of Different Prepositions (2)

The robot went away from the cone then went behind the box which is **right of** *the chair and which is* **in front of** *the cone then went towards the stool.*

Language Comprehension

The Effect of Different Prepositions (2)

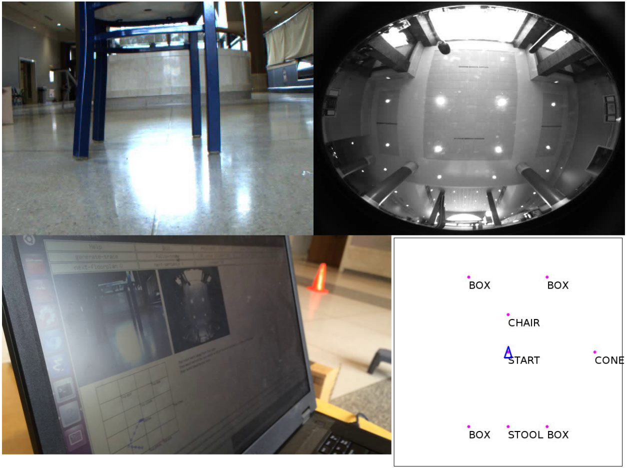
The robot went away from the cone then went behind the box which is **right of** *the chair and which is* **in front of** *the cone then went towards the stool.*



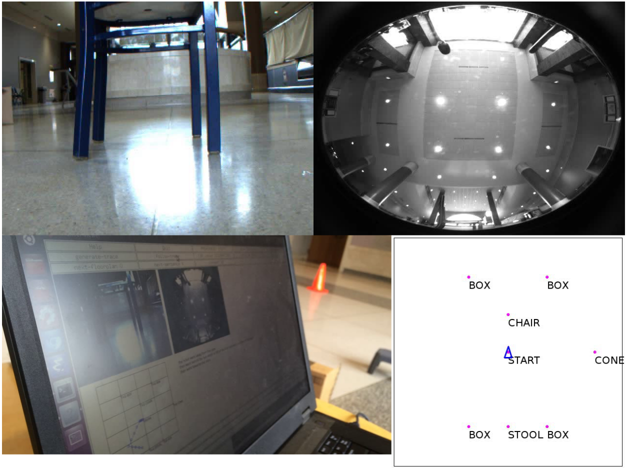
Language Comprehension

The Effect of Different Prepositions (3)

The robot went away from the cone then went behind the box which is **<u>left of</u>** *the chair and which is* **<u>in front of</u>** *the cone then went towards the stool.*



The robot went away from the cone then went behind the box which is **<u>left of</u>** *the chair and which is* **<u>in front of</u>** *the cone then went towards the stool.*



Language Comprehension

The Effect of Different Prepositions (3)

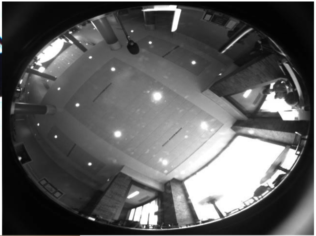
The robot went away from the cone then went behind the box which is **left of** *the chair and which is* **in front of** *the cone then went towards the stool.*

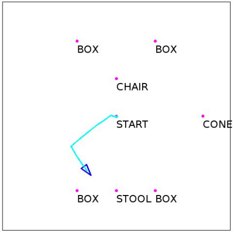
The robot went away from the cone then went behind the box which is **left of** *the chair and which is* **in front of** *the cone then went towards the stool.*

Language Comprehension

The Effect of Different Prepositions (3)

The robot went away from the cone then went behind the box which is **left of** *the chair and which is* **in front of** *the cone then went towards the stool.*





Logical Form

The robot went toward the left side of the stool, then toward the cone which is behind the stool, then toward the table which is left of the cone, then went back toward the stool and to the left of the stool.

Logical Form

The robot went toward the left side of the stool, then toward the cone which is behind the stool, then toward the table which is left of the cone, then went back toward the stool and to the left of the stool.

$$[\alpha,\beta,\gamma,\delta]\{t,u,v,w,x,y,z\}\left(\begin{array}{l}\textsc{left}(\alpha,t)\wedge\textsc{stool}(t)\wedge\\ \textsc{toward}(\beta,u)\wedge\textsc{cone}(u)\wedge\textsc{behind}(u,v)\wedge\textsc{stool}(v)\wedge\\ \textsc{toward}(\gamma,w)\wedge\textsc{table}(w)\wedge\textsc{left}(w,x)\wedge\textsc{cone}(x)\wedge\\ \textsc{toward}(\delta,y)\wedge\textsc{left}(\delta,z)\wedge\textsc{stool}(y)\wedge\textsc{stool}(z)\end{array}\right)$$

Logical Form

The robot went toward the left side of the stool, then toward the cone which is behind the stool, then toward the table which is left of the cone, then went back toward the stool and to the left of the stool.

$$[\alpha,\beta,\gamma,\delta]\{t,u,v,w,x,y,z\}\left(\begin{array}{l}\text{LEFT}(\alpha,t)\wedge\text{STOOL}(t)\wedge\\ \text{TOWARD}(\beta,u)\wedge\text{CONE}(u)\wedge\text{BEHIND}(u,v)\wedge\text{STOOL}(v)\wedge\\ \text{TOWARD}(\gamma,w)\wedge\text{TABLE}(w)\wedge\text{LEFT}(w,x)\wedge\text{CONE}(x)\wedge\\ \text{TOWARD}(\delta,y)\wedge\text{LEFT}(\delta,z)\wedge\text{STOOL}(y)\wedge\text{STOOL}(z)\end{array}\right)$$

Logical Form

The robot went toward the left side of the stool, then toward the cone which is behind the stool, then toward the table which is left of the cone, then went back toward the stool and to the left of the stool.

$$[\alpha,\beta,\gamma,\delta]\{t,u,v,w,x,y,z\}\left(\begin{array}{l}\text{LEFT}(\alpha,t)\wedge\text{STOOL}(t)\wedge\\ \text{TOWARD}(\beta,u)\wedge\text{CONE}(u)\wedge\text{BEHIND}(u,v)\wedge\text{STOOL}(v)\wedge\\ \text{TOWARD}(\gamma,w)\wedge\text{TABLE}(w)\wedge\text{LEFT}(w,x)\wedge\text{CONE}(x)\wedge\\ \text{TOWARD}(\delta,y)\wedge\text{LEFT}(\delta,z)\wedge\text{STOOL}(y)\wedge\text{STOOL}(z)\end{array}\right)$$

Logical Form

The robot went toward the left side of the stool, then toward the cone which is behind the stool, then toward the table which is left of the cone, then went back toward the stool and to the left of the stool.

$$[\alpha, \beta, \gamma, \delta]\{t, u, v, w, x, y, z\} \left(\begin{array}{l} \text{LEFT}(\alpha, t) \wedge \text{STOOL}(t) \wedge \\ \text{TOWARD}(\beta, u) \wedge \text{CONE}(u) \wedge \text{BEHIND}(u, v) \wedge \text{STOOL}(v) \wedge \\ \text{TOWARD}(\gamma, w) \wedge \text{TABLE}(w) \wedge \text{LEFT}(w, x) \wedge \text{CONE}(x) \wedge \\ \text{TOWARD}(\delta, y) \wedge \text{LEFT}(\delta, z) \wedge \text{STOOL}(y) \wedge \text{STOOL}(z) \end{array} \right)$$

Logical Form

The robot went toward the left side of the stool, then toward the cone which is behind the stool, then toward the table which is left of the cone, then went back toward the stool and to the left of the stool.

$$[\alpha, \beta, \gamma, \delta]\{t, u, v, w, x, y, z\} \left(\begin{array}{l} \text{LEFT}(\alpha, t) \wedge \text{STOOL}(t) \wedge \\ \text{TOWARD}(\beta, u) \wedge \text{CONE}(u) \wedge \text{BEHIND}(u, v) \wedge \text{STOOL}(v) \wedge \\ \text{TOWARD}(\gamma, w) \wedge \text{TABLE}(w) \wedge \text{LEFT}(w, x) \wedge \text{CONE}(x) \wedge \\ \text{TOWARD}(\delta, y) \wedge \text{LEFT}(\delta, z) \wedge \text{STOOL}(y) \wedge \text{STOOL}(z) \end{array} \right)$$

Logical Form

The robot went toward the left side of the stool, then toward the cone which is behind the stool, then toward the table which is left of the cone, then went back toward the stool and to the left of the stool.

$$[\alpha, \beta, \gamma, \delta]\{t, u, v, w, x, y, z\} \left(\begin{array}{l} \text{LEFT}(\alpha, t) \wedge \text{STOOL}(t) \wedge \\ \text{TOWARD}(\beta, u) \wedge \text{CONE}(u) \wedge \text{BEHIND}(u, v) \wedge \text{STOOL}(v) \wedge \\ \text{TOWARD}(\gamma, w) \wedge \text{TABLE}(w) \wedge \text{LEFT}(w, x) \wedge \text{CONE}(x) \wedge \\ \text{TOWARD}(\delta, y) \wedge \text{LEFT}(\delta, z) \wedge \text{STOOL}(y) \wedge \text{STOOL}(z) \end{array} \right)$$

- all sentences naturally elicited from humans through AMT

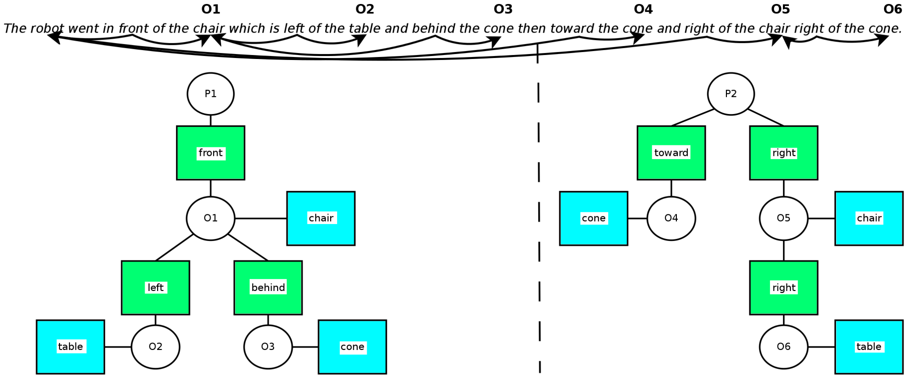
Logical Form

The robot went toward the left side of the stool, then toward the cone which is behind the stool, then toward the table which is left of the cone, then went back toward the stool and to the left of the stool.

$$[\alpha,\beta,\gamma,\delta]\{t,u,v,w,x,y,z\}\left(\begin{array}{l}\text{LEFT}(\alpha,t)\wedge\text{STOOL}(t)\wedge\\ \text{TOWARD}(\beta,u)\wedge\text{CONE}(u)\wedge\text{BEHIND}(u,v)\wedge\text{STOOL}(v)\wedge\\ \text{TOWARD}(\gamma,w)\wedge\text{TABLE}(w)\wedge\text{LEFT}(w,x)\wedge\text{CONE}(x)\wedge\\ \text{TOWARD}(\delta,y)\wedge\text{LEFT}(\delta,z)\wedge\text{STOOL}(y)\wedge\text{STOOL}(z)\end{array}\right)$$

- all sentences naturally elicited from humans through AMT
- no grammar or parse trees at all

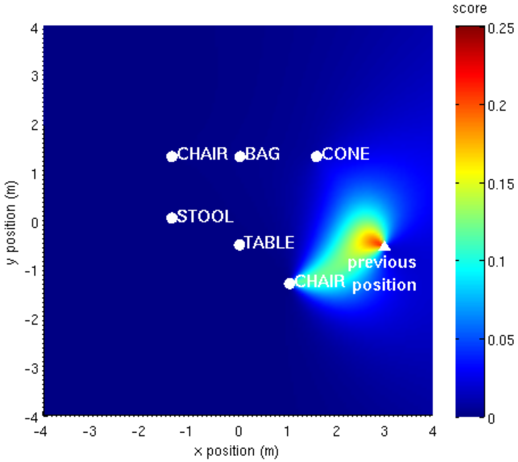
Parsing


O1
O2
O3
O4
O5
O6
The robot went in front of the chair which is left of the table and behind the cone then toward the cone and right of the chair right of the cone.
P1
front
O1
chair
left
behind
table
O2
O3
cone
P2
toward
right
cone
O4
O5
chair
right
O6
table

Semantics as a Soft Context-Sensitive Scoring Function

O4 **O5** **O6**

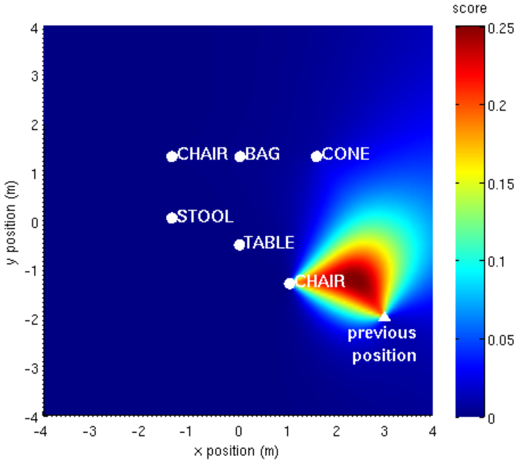
toward the cone and right of the chair right of the table.



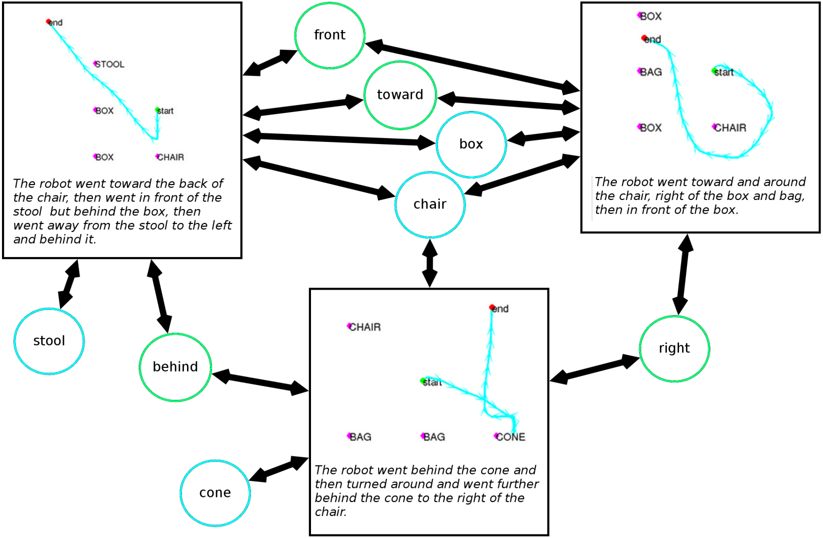
Semantics as a Soft Context-Sensitive Scoring Function

O4 **O5** **O6**

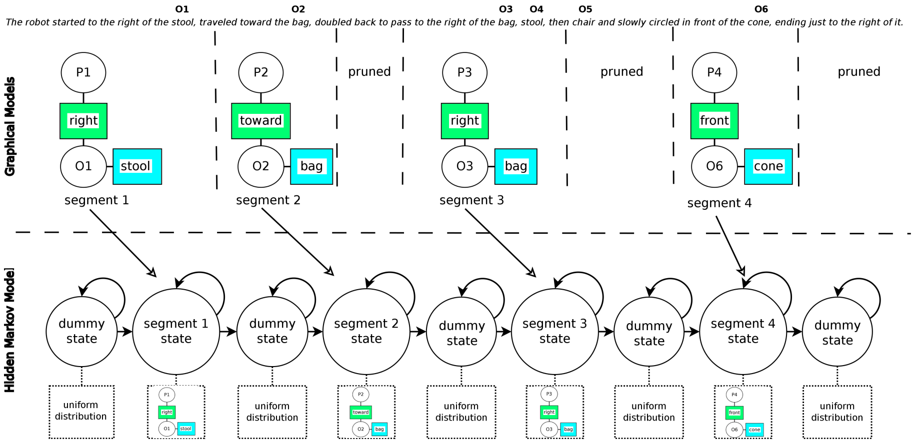
toward the cone and right of the chair right of the table.



Acquisition Method

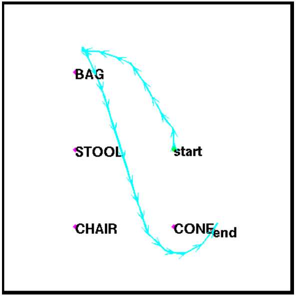


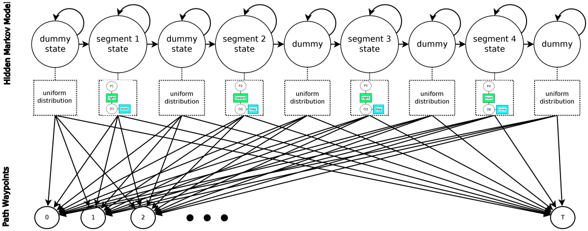
Acquisition Method



Acquisition Method

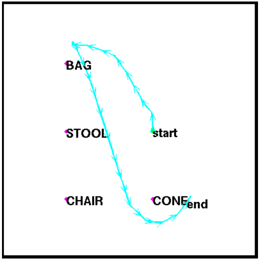
The robot started to the right of the stool, traveled toward the bag, doubled back to pass to the right of the bag, stool, then chair and slowly circled in front of the cone, ending just to the right of it.

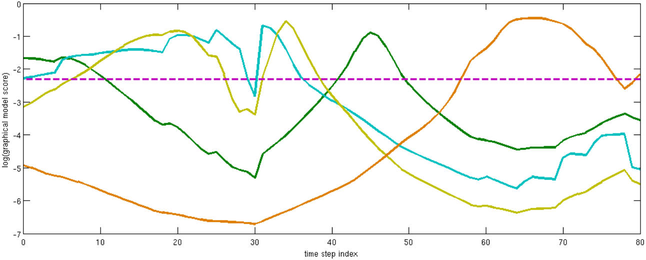




Acquisition Method

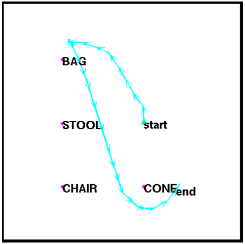
The robot started to the right of the stool, traveled toward the bag, doubled back to pass to the right of the bag, stool, then chair and slowly circled in front of the cone, ending just to the right of it.

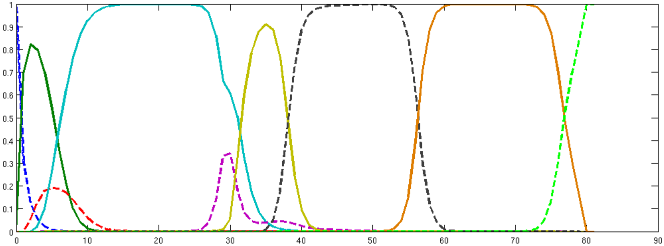




Acquisition Method

The robot started to the right of the stool, traveled toward the bag, doubled back to pass to the right of the bag, stool, then chair and slowly circled in front of the cone, ending just to the right of it.

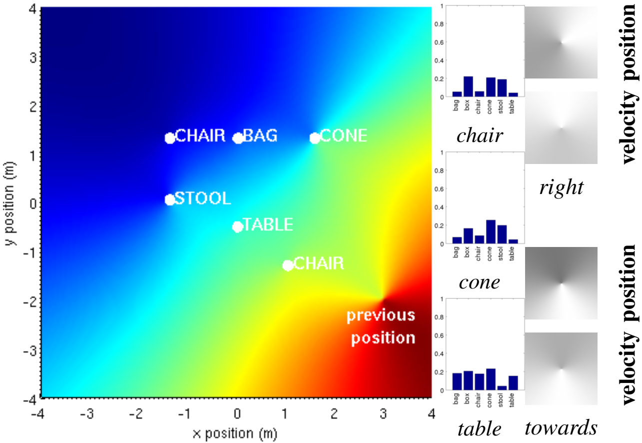




Acquisition Method

Iteration 0

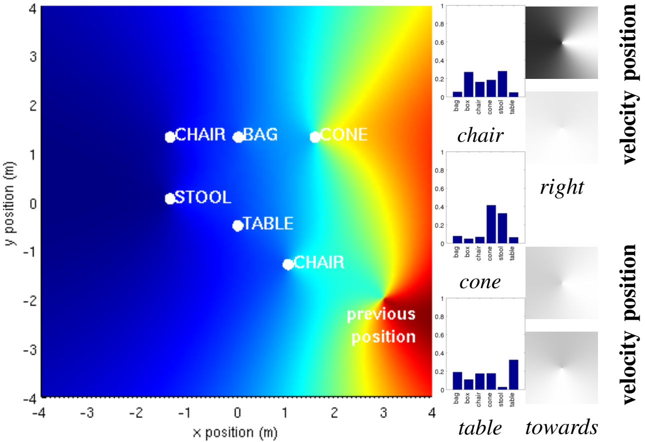
toward the cone and right of the chair right of the table



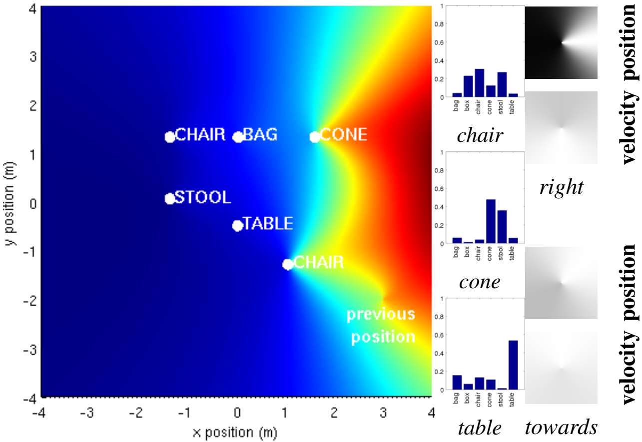
Acquisition Method

Iteration 1

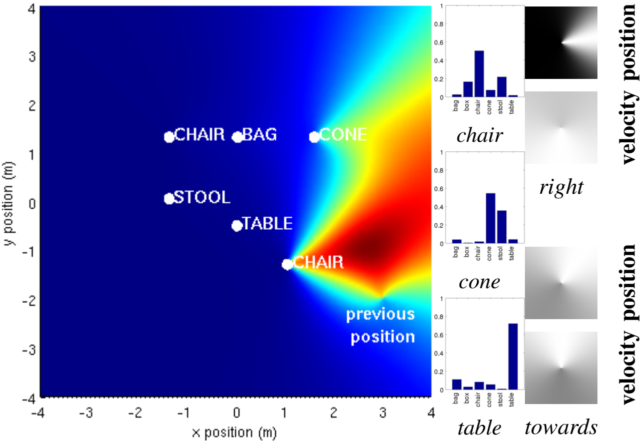
toward the cone and right of the chair right of the table



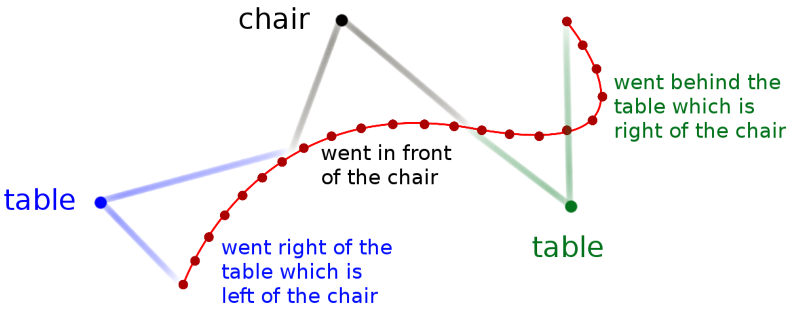
toward the cone and right of the chair right of the table



toward the cone and right of the chair right of the table



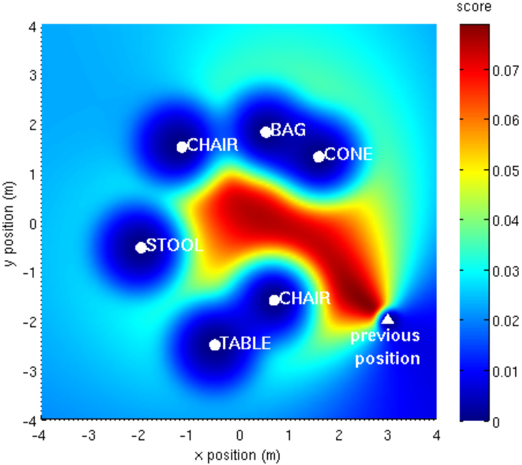
Generation Method



The robot went right of the table which is left of the chair, then went in front of the chair, then went behind the table which is right of the chair.

Comprehension Method

toward the chair left of the bag



Dataset

1. 10 random floorplans
   - 3 or 4 objects, at most one duplicate
   - tile corners (not perimeter)
2. 25 random sentences per floorplan
3. manually drive 250 paths; recover paths from odometry
4. get 3 AMT sentences for each path, 750 total
5. get AMT judgments for each sentence-path pair

acquisition (human sentences vs. human-driven paths)
tests human performance because testing human sentences produced after paths

Dataset

1. 10 random floorplans
   - 4 or 5 objects, at most one duplicate
   - tile corners, centers, or edge centers (not perimeter)
2. 10 random sentences per floorplan
3. automatically drive 100 paths; recover paths from odometry
4. get 3 AMT sentences for each path, 300 total
5. get AMT judgments for each sentence-path pair

comprehension (human sentences vs. machine-driven paths)
tests human performance because testing human sentences produced after paths

Dataset

1. 10 random floorplans
   - 4 or 5 objects, at most one duplicate
   - tile corners, centers, or edge centers (not perimeter)
2. 10 random sentences per floorplan
3. automatically drive 100 paths; recover paths from odometry
4. get 3 AMT sentences for each path, 300 total
5. automatically drive 300 paths; recover paths from odometry
6. get AMT judgments for each sentence-path pair

comprehension (human sentences vs. machine-driven paths) tests machine performance because testing machine production of paths after human production of sentences

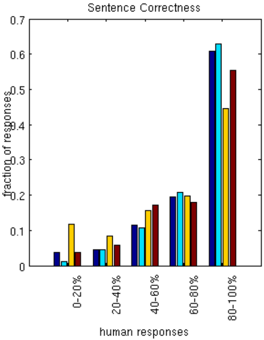
Dataset

1. 10 random floorplans
   - 4 or 5 objects, at most one duplicate
   - tile corners, centers, or edge centers (not perimeter)
2. 10 random sentences per floorplan
3. manually drive 100 paths; recover paths from odometry
4. generate 100 sentences
5. get AMT judgments for each sentence-path pair

generation (machine sentences vs. human-driven paths) tests machine performance because testing machine production of sentences from paths

Sentence Correctness

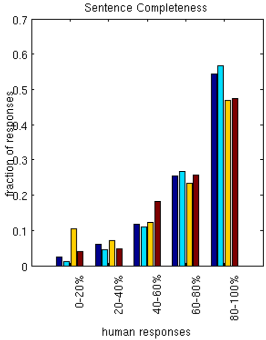
Approximately how much of the sentence is true of the path?



- acquisition (human sentences vs. human-driven paths)
- comprehension (human sentences vs. machine-driven paths)
- comprehension (human sentences vs. machine-driven paths)
- generation (machine sentences vs. human-driven paths)

Sentence Completeness

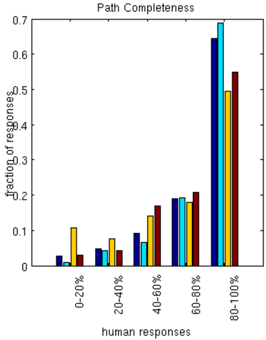
Approximately how much of the path is described by the sentence?



- acquisition (human sentences vs. human-driven paths)
- comprehension (human sentences vs. machine-driven paths)
- comprehension (human sentences vs. machine-driven paths)
- generation (machine sentences vs. human-driven paths)

Path Completeness

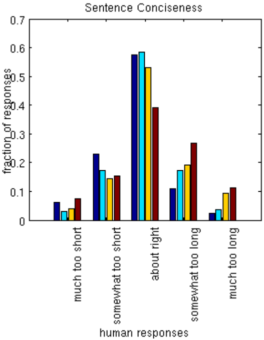
Approximately how much of the sentence is depicted by the path?



- acquisition (human sentences vs. human-driven paths)
- comprehension (human sentences vs. machine-driven paths)
- comprehension (human sentences vs. machine-driven paths)
- generation (machine sentences vs. human-driven paths)

Sentence Conciseness

Rate the length of the sentence.



- acquisition (human sentences vs. human-driven paths)
- comprehension (human sentences vs. machine-driven paths)
- comprehension (human sentences vs. machine-driven paths)
- generation (machine sentences vs. human-driven paths)

Outline

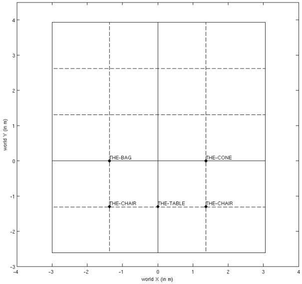
1. Verbs, Arguments, and Predication in the Human Brain
   - Experiment 1: hollywood2-text-speech
   - Experiment 2: compositionality-noninterleaved
   - Experiment 3: predication

2. Sentence Directed Video Object Codetection

3. Driving Under the Influence (of Language)
   - Grounding Language Semantics in Robotics
   - Object Codetection from Mobile Robot Video

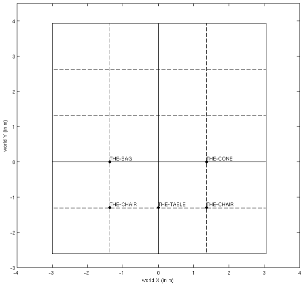
4. Playing Checkers from English

Floorplans



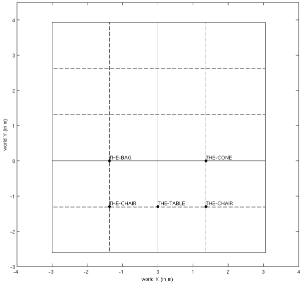
- So far, acquisition, generation, and comprehension all required floorplan as input.

Floorplans



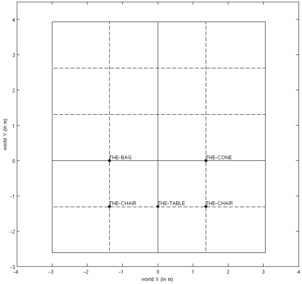
- So far, acquisition, generation, and comprehension all required floorplan as input.
- The floorplan took the form of a set of 2D points labeled with abstract classes.
  $\{(5, -3) : \mathbf{foo}, (-7, 3) : \mathbf{bar}\}$

Floorplans



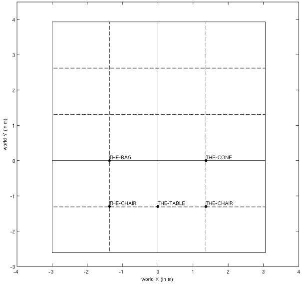
- The class labeling across different floorplans must be consistent; two instances of the same object class (in the same or different floorplans) should have the same label. This is what allows acquisition to work.

Floorplans



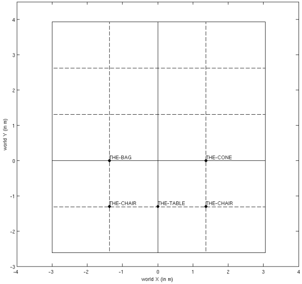
- The class labeling across different floorplans must be consistent; two instances of the same object class (in the same or different floorplans) should have the same label. This is what allows acquisition to work.
- The mapping from nouns to abstract class labels is ***learned*** (by the acquisition process).

Floorplans



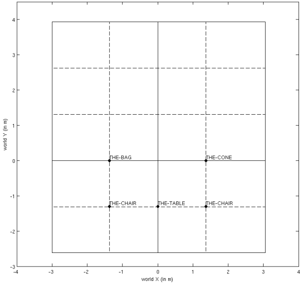
- It need not be a bijection.

Floorplans



- It need not be a bijection.
  - A noun can correspond to more than one abstract class label (homonymy).

Floorplans



- It need not be a bijection.
  - A noun can correspond to more than one abstract class label (homonymy).
  - An abstract class label can correspond to more than one noun (synonymy).

Using Codetection to Recover Floorplans

- Compute floorplan automatically from video stream and odometry using codetection

Using Codetection to Recover Floorplans

- Compute floorplan automatically from video stream and odometry using codetection
- Different from prior work on codetection

Using Codetection to Recover Floorplans

- Compute floorplan automatically from video stream and odometry using codetection
- Different from prior work on codetection
  1. egocentric video from a moving camera (changing position and orientation)

Using Codetection to Recover Floorplans

- Compute floorplan automatically from video stream and odometry using codetection
- Different from prior work on codetection
  1. egocentric video from a moving camera (changing position and orientation)
  2. integrates video stream with odometry and inertial guidance

Using Codetection to Recover Floorplans

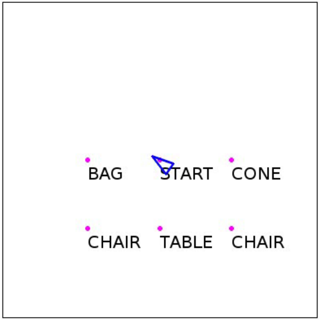
- Compute floorplan automatically from video stream and odometry using codetection
- Different from prior work on codetection
  1. egocentric video from a moving camera (changing position and orientation)
  2. integrates video stream with odometry and inertial guidance
  3. localizes in 3D, not just 2D

Data Collection

The robot went towards the chair which is left of the table then went away from the cone then went away from the bag then went behind the chair which is right of the table then went towards the table.

Video

Trace

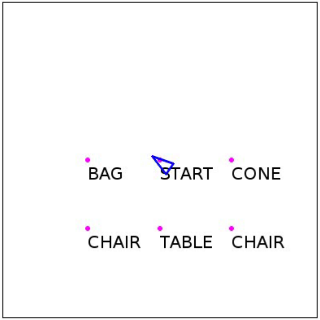


Data Collection

The robot went towards the chair which is left of the table then went away from the cone then went away from the bag then went behind the chair which is right of the table then went towards the table.

Video

Trace



Data Collection

The robot went towards the chair which is left of the table then went away from the cone then went away from the bag then went behind the chair which is right of the table then went towards the table.

Video | Trace

Data Collection

The robot went towards the chair which is left of the table then went away from the cone then went away from the bag then went behind the chair which is right of the table then went towards the table.

Video

Trace

Data Collection

The robot went towards the chair which is left of the table then went away from the cone then went away from the bag then went behind the chair which is right of the table then went towards the table.

Video

Trace

Data Collection

The robot went towards the chair which is left of the table then went away from the cone then went away from the bag then went behind the chair which is right of the table then went towards the table.

Video

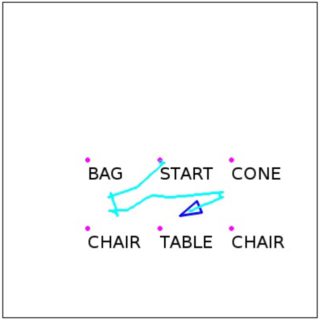
Trace

Data Collection

The robot went towards the chair which is left of the table then went away from the cone then went away from the bag then went behind the chair which is right of the table then went towards the table.

Video

Trace



. . . **plus 59 more**

Method

Codetection approach is a five-step process:

Method

Codetection approach is a five-step process:

1. **Proposal Generation** Generate a set of 2D proposal boxes for each frame.

Method

Codetection approach is a five-step process:

1. **Proposal Generation** Generate a set of 2D proposal boxes for each frame.
2. **Proposal Localization** Find 3D world location for each proposal box.

Method

Codetection approach is a five-step process:

1. **Proposal Generation** Generate a set of 2D proposal boxes for each frame.
2. **Proposal Localization** Find 3D world location for each proposal box.
3. **Proposal Selection** Select at most a single proposal for each frame that denotes the prominent object in the field of view. Some frames will not have a selected proposal because there may not be a prominent object in the field of view.

Method

Codetection approach is a five-step process:

1. **Proposal Generation** Generate a set of 2D proposal boxes for each frame.
2. **Proposal Localization** Find 3D world location for each proposal box.
3. **Proposal Selection** Select at most a single proposal for each frame that denotes the prominent object in the field of view. Some frames will not have a selected proposal because there may not be a prominent object in the field of view.
4. **Clustering** Cluster locations of selected proposals to find object locations on the floor plan.

Method

Codetection approach is a five-step process:

1. **Proposal Generation** Generate a set of 2D proposal boxes for each frame.
2. **Proposal Localization** Find 3D world location for each proposal box.
3. **Proposal Selection** Select at most a single proposal for each frame that denotes the prominent object in the field of view. Some frames will not have a selected proposal because there may not be a prominent object in the field of view.
4. **Clustering** Cluster locations of selected proposals to find object locations on the floor plan.
5. **Labeling** Assign an abstract class label to each localized object instance.

Method

Codetection approach is a five-step process:

1. **Proposal Generation** Generate a set of 2D proposal boxes for each frame.
2. **Proposal Localization** Find 3D world location for each proposal box.
3. **Proposal Selection** Select at most a single proposal for each frame that denotes the prominent object in the field of view. Some frames will not have a selected proposal because there may not be a prominent object in the field of view.
4. **Clustering** Cluster locations of selected proposals to find object locations on the floor plan.
5. **Labeling** Assign an abstract class label to each localized object instance.

Steps 1–4 are done independently for each floorplan, but jointly across all paths driven in that floorplan.

Method

Codetection approach is a five-step process:

1. **Proposal Generation** Generate a set of 2D proposal boxes for each frame.
2. **Proposal Localization** Find 3D world location for each proposal box.
3. **Proposal Selection** Select at most a single proposal for each frame that denotes the prominent object in the field of view. Some frames will not have a selected proposal because there may not be a prominent object in the field of view.
4. **Clustering** Cluster locations of selected proposals to find object locations on the floor plan.
5. **Labeling** Assign an abstract class label to each localized object instance.

Steps 1–4 are done independently for each floorplan, but jointly across all paths driven in that floorplan.
Step 5 is done jointly across all floorplans.

Proposal Generation and Selection

1. Generate proposals with MCG (Arbelaez et al. 2014)

Proposal Generation and Selection

1. Generate proposals with MCG (Arbelaez et al. 2014)
2. Graphical model

Proposal Generation and Selection

1. Generate proposals with MCG (Arbelaez et al. 2014)
2. Graphical model
   - vertex for each frame to denote the most prominent object in that frame

Proposal Generation and Selection

1. Generate proposals with MCG (Arbelaez et al. 2014)
2. Graphical model
   - vertex for each frame to denote the most prominent object in that frame
   - vertex labels range over proposals plus dummy

Proposal Generation and Selection

1. Generate proposals with MCG (Arbelaez et al. 2014)
2. Graphical model
   - vertex for each frame to denote the most prominent object in that frame
   - vertex labels range over proposals plus dummy
   - complete graph except no self edges

Proposal Generation and Selection

1. Generate proposals with MCG (Arbelaez et al. 2014)
2. Graphical model
   - vertex for each frame to denote the most prominent object in that frame
   - vertex labels range over proposals plus dummy
   - complete graph except no self edges
   - proposal score as vertex score, penalized by implausibility of world size and position recovered with projective geometry

Proposal Generation and Selection

1. Generate proposals with MCG (Arbelaez et al. 2014)
2. Graphical model
   - vertex for each frame to denote the most prominent object in that frame
   - vertex labels range over proposals plus dummy
   - complete graph except no self edges
   - proposal score as vertex score, penalized by implausibility of world size and position recovered with projective geometry
     - behind camera (bottom edge above horizon)

Proposal Generation and Selection

1. Generate proposals with MCG (Arbelaez et al. 2014)
2. Graphical model
   - vertex for each frame to denote the most prominent object in that frame
   - vertex labels range over proposals plus dummy
   - complete graph except no self edges
   - proposal score as vertex score, penalized by implausibility of world size and position recovered with projective geometry
     - behind camera (bottom edge above horizon)
     - close to any two image boundaries

Proposal Generation and Selection

1. Generate proposals with MCG (Arbelaez et al. 2014)
2. Graphical model
   - vertex for each frame to denote the most prominent object in that frame
   - vertex labels range over proposals plus dummy
   - complete graph except no self edges
   - proposal score as vertex score, penalized by implausibility of world size and position recovered with projective geometry
     - behind camera (bottom edge above horizon)
     - close to any two image boundaries
     - close to any single image boundary and exceed specified height or width

Proposal Generation and Selection

1. Generate proposals with MCG (Arbelaez et al. 2014)
2. Graphical model
   - vertex for each frame to denote the most prominent object in that frame
   - vertex labels range over proposals plus dummy
   - complete graph except no self edges
   - proposal score as vertex score, penalized by implausibility of world size and position recovered with projective geometry
     - behind camera (bottom edge above horizon)
     - close to any two image boundaries
     - close to any single image boundary and exceed specified height or width
     - exceed both specified height and width

Proposal Generation and Selection

1. Generate proposals with MCG (Arbelaez et al. 2014)
2. Graphical model
   - vertex for each frame to denote the most prominent object in that frame
   - vertex labels range over proposals plus dummy
   - complete graph except no self edges
   - proposal score as vertex score, penalized by implausibility of world size and position recovered with projective geometry
     - behind camera (bottom edge above horizon)
     - close to any two image boundaries
     - close to any single image boundary and exceed specified height or width
     - exceed both specified height and width
     - outside floorplan

Proposal Generation and Selection

1. Generate proposals with MCG (Arbelaez et al. 2014)
2. Graphical model
    - vertex for each frame to denote the most prominent object in that frame
    - vertex labels range over proposals plus dummy
    - complete graph except no self edges
    - proposal score as vertex score, penalized by implausibility of world size and position recovered with projective geometry
        - behind camera (bottom edge above horizon)
        - close to any two image boundaries
        - close to any single image boundary and exceed specified height or width
        - exceed both specified height and width
        - outside floorplan
    - edge score is weighted sum of

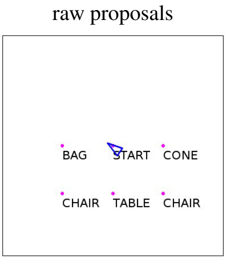
Proposal Generation and Selection

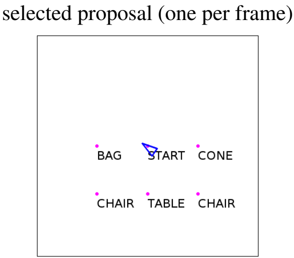
1. Generate proposals with MCG (Arbelaez et al. 2014)
2. Graphical model
   - vertex for each frame to denote the most prominent object in that frame
   - vertex labels range over proposals plus dummy
   - complete graph except no self edges
   - proposal score as vertex score, penalized by implausibility of world size and position recovered with projective geometry
     - behind camera (bottom edge above horizon)
     - close to any two image boundaries
     - close to any single image boundary and exceed specified height or width
     - exceed both specified height and width
     - outside floorplan
   - edge score is weighted sum of
     - similarity of SIFT descriptors

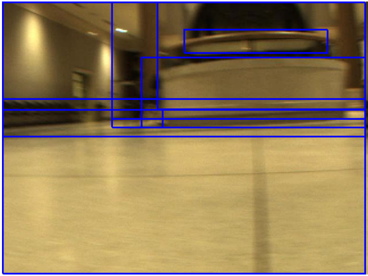
Proposal Generation and Selection

1. Generate proposals with MCG (Arbelaez et al. 2014)
2. Graphical model
   - vertex for each frame to denote the most prominent object in that frame
   - vertex labels range over proposals plus dummy
   - complete graph except no self edges
   - proposal score as vertex score, penalized by implausibility of world size and position recovered with projective geometry
     - behind camera (bottom edge above horizon)
     - close to any two image boundaries
     - close to any single image boundary and exceed specified height or width
     - exceed both specified height and width
     - outside floorplan
   - edge score is weighted sum of
     - similarity of SIFT descriptors
     - similarity of world size and position as determined by projective geometry

Proposal Generation, Selection, and Localization

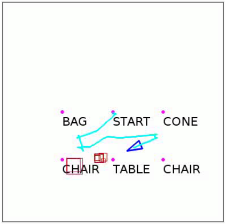




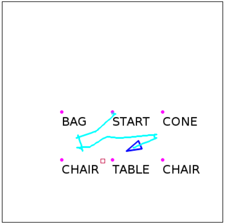


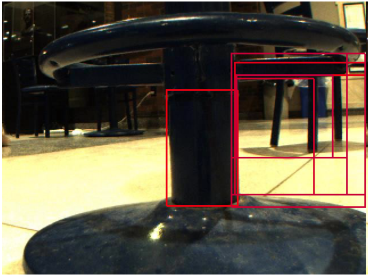
Proposal Generation, Selection, and Localization

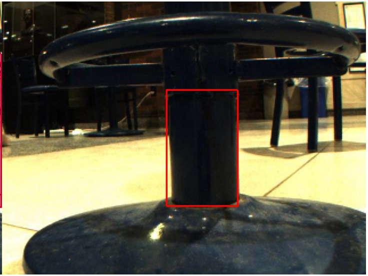
raw proposals



selected proposal (one per frame)

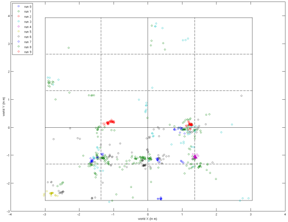






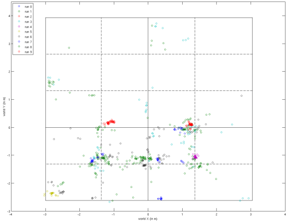
Clustering

Take selected proposal locations for all navigational paths on a floor plan



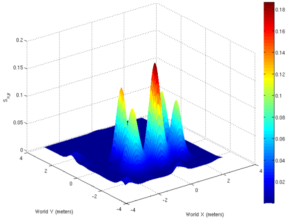
Clustering

Take selected proposal locations for all navigational paths on a floor plan



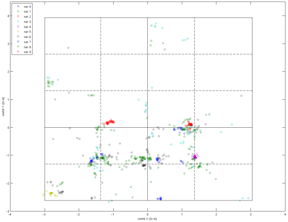
Compute density function

$$S_{x,y} = \sum_{n=1}^{N} f_n \frac{\|(x,y) - (x_n, y_n)\|}{v_n}$$



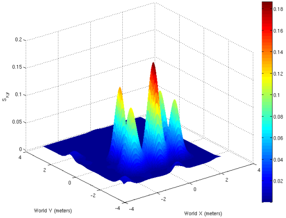
Clustering

Take selected proposal locations for all navigational paths on a floor plan

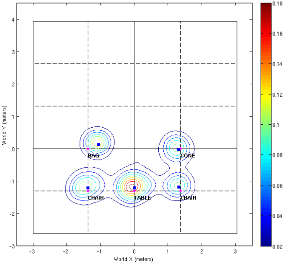


Compute density function

$$S_{x,y} = \sum_{n=1}^{N} f_n \frac{\|(x,y) - (x_n, y_n)\|}{v_n}$$



Find peaks to locate objects



Labeling

1. Assign proposals to closest peak, rejecting outliers.

Labeling

1. Assign proposals to closest peak, rejecting outliers.
2. Let C_i denote the proposals assigned to peak i.

Labeling

1. Assign proposals to closest peak, rejecting outliers.
2. Let C_i denote the proposals assigned to peak i.
3. Let U_{ab} denote the similarity between proposals a and b.

Labeling

1. Assign proposals to closest peak, rejecting outliers.
2. Let C_i denote the proposals assigned to peak i.
3. Let U_{ab} denote the similarity between proposals a and b.
4. Compute similarity Q_{ij} between peaks i and j.

$$Q_{ij} = \frac{\sum_{a \in C_i} \max_{b \in C_j} U_{ab} + \sum_{b \in C_j} \max_{a \in C_i} U_{ab}}{|C_i| + |C_j|}$$

Labeling

1. Assign proposals to closest peak, rejecting outliers.
2. Let C_i denote the proposals assigned to peak i.
3. Let U_{ab} denote the similarity between proposals a and b.
4. Compute similarity Q_{ij} between peaks i and j.

$$Q_{ij} = \frac{\sum_{a \in C_i} \max_{b \in C_j} U_{ab} + \sum_{b \in C_j} \max_{a \in C_i} U_{ab}}{|C_i| + |C_j|}$$

5. Form a graphical model

Labeling

1. Assign proposals to closest peak, rejecting outliers.
2. Let C_i denote the proposals assigned to peak i.
3. Let U_{ab} denote the similarity between proposals a and b.
4. Compute similarity Q_{ij} between peaks i and j.

$$Q_{ij} = \frac{\sum_{a \in C_i} \max_{b \in C_j} U_{ab} + \sum_{b \in C_j} \max_{a \in C_i} U_{ab}}{|C_i| + |C_j|}$$

5. Form a graphical model
   - vertex for each peak

Labeling

1. Assign proposals to closest peak, rejecting outliers.
2. Let C_i denote the proposals assigned to peak i.
3. Let U_{ab} denote the similarity between proposals a and b.
4. Compute similarity Q_{ij} between peaks i and j.

$$Q_{ij} = \frac{\sum_{a \in C_i} \max_{b \in C_j} U_{ab} + \sum_{b \in C_j} \max_{a \in C_i} U_{ab}}{|C_i| + |C_j|}$$

5. Form a graphical model
   - vertex for each peak
   - vertex labels range over abstract object classes

Labeling

1. Assign proposals to closest peak, rejecting outliers.
2. Let C_i denote the proposals assigned to peak i.
3. Let U_{ab} denote the similarity between proposals a and b.
4. Compute similarity Q_{ij} between peaks i and j.

$$Q_{ij} = \frac{\displaystyle\sum_{a \in C_i} \max_{b \in C_j} U_{ab} + \sum_{b \in C_j} \max_{a \in C_i} U_{ab}}{|C_i| + |C_j|}$$

5. Form a graphical model
   - vertex for each peak
   - vertex labels range over abstract object classes
   - complete graph except no self edges

Labeling

1. Assign proposals to closest peak, rejecting outliers.
2. Let C_i denote the proposals assigned to peak i.
3. Let U_{ab} denote the similarity between proposals a and b.
4. Compute similarity Q_{ij} between peaks i and j.

$$Q_{ij} = \frac{\sum_{a \in C_i} \max_{b \in C_j} U_{ab} + \sum_{b \in C_j} \max_{a \in C_i} U_{ab}}{|C_i| + |C_j|}$$

5. Form a graphical model
   - vertex for each peak
   - vertex labels range over abstract object classes
   - complete graph except no self edges
   - no vertex score

Labeling

1. Assign proposals to closest peak, rejecting outliers.
2. Let C_i denote the proposals assigned to peak i.
3. Let U_{ab} denote the similarity between proposals a and b.
4. Compute similarity Q_{ij} between peaks i and j.

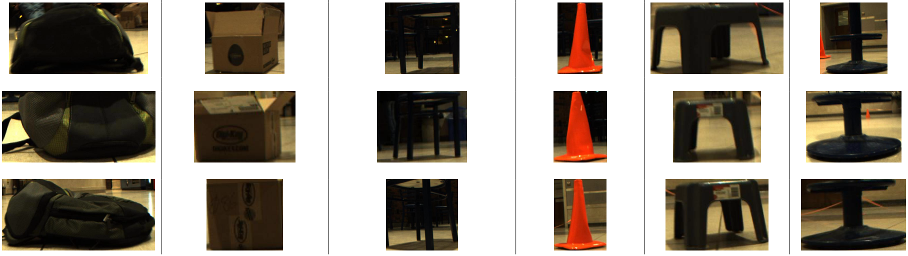
$$Q_{ij} = \frac{\sum_{a \in C_i} \max_{b \in C_j} U_{ab} + \sum_{b \in C_j} \max_{a \in C_i} U_{ab}}{|C_i| + |C_j|}$$

5. Form a graphical model
   - vertex for each peak
   - vertex labels range over abstract object classes
   - complete graph except no self edges
   - no vertex score
   - edge score high if same label and high similarity or different label and low similarity

Results

Labeling output:

BAG	BOX	CHAIR	CONE	STOOL	TABLE
class labels: 1, 2	class labels: 3, 4, 5	class labels: 6, 7, 8, 9	class label: 10	class label: 11	class label: 12

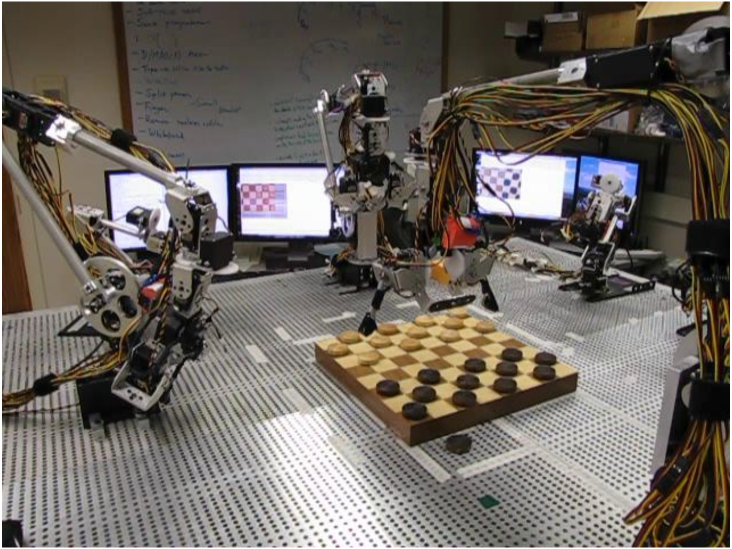


Outline

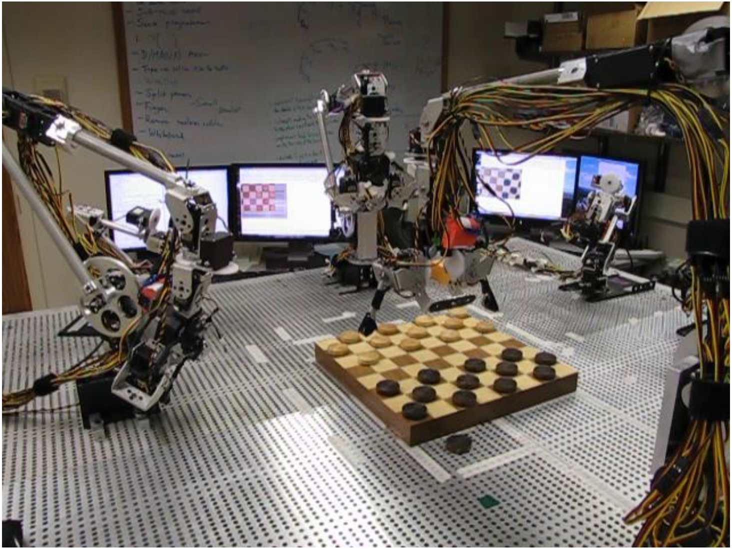
Joint work

Daniel Paul Barrett   Seth Benjamin   Zachary Burchill

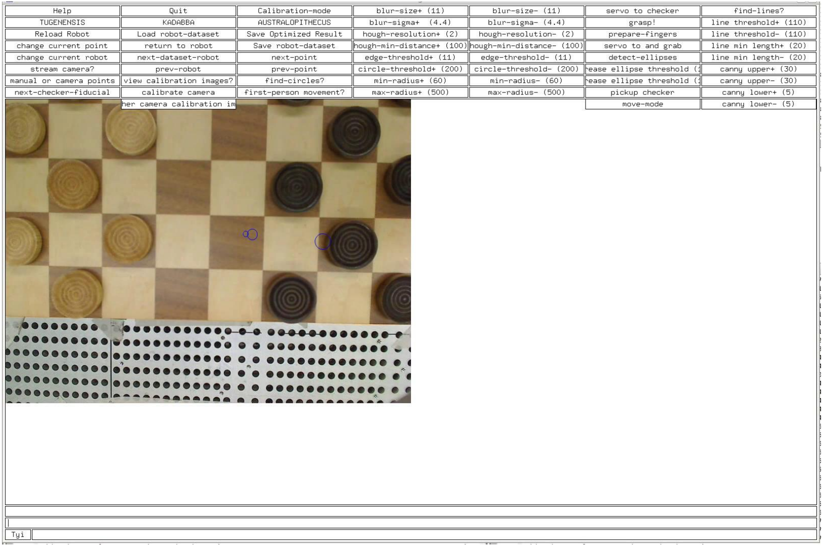
Two Robots Playing Checkers



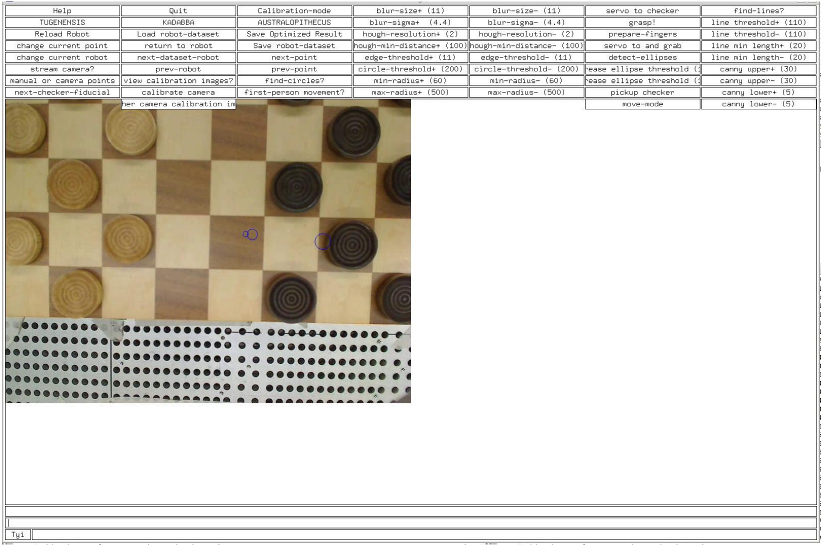
Two Robots Playing Checkers



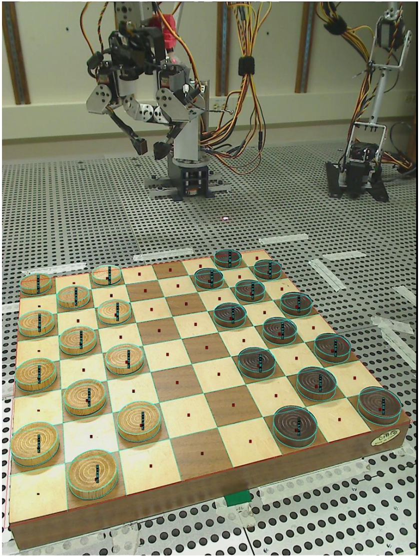
View from the Palm Camera



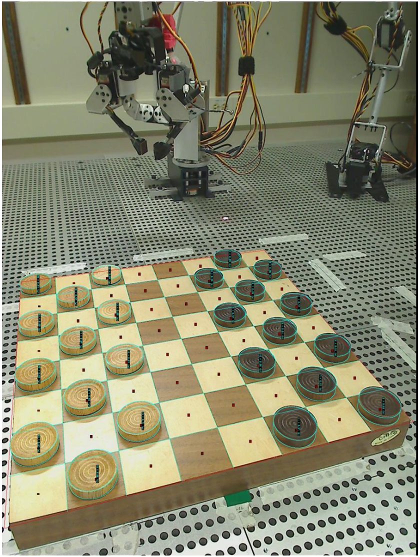
View from the Palm Camera



Recovering Checkers Game State from Computer Vision



Recovering Checkers Game State from Computer Vision



Sources of English Rules for Checkers

1. http://boardgames.about.com/cs/checkersdraughts/ht/play_checkers.htm
2. http://simple.wikipedia.org/wiki/Checkers
3. http://www.darkfish.com/checkers/rules.html
4. http://www.ducksters.com/games/checkers_rules.php
5. http://www.7is7.com/software/games/checkers/rules-en-us.html
6. http://www.chesslab.com/rules/checkersbasics.html
   http://www.chesslab.com/rules/checkersrules.html
7. http://www.learnplaywin.net/checkers/checkers-rules.htm
8. http://www.gametableonline.com/pop_rules.php?gid=20
9. http://www.indepthinfo.com/checkers/setup.shtml
   http://www.indepthinfo.com/checkers/play.shtml
   http://www.indepthinfo.com/checkers/crowning.shtml
10. http://winning-moves.com/images/kingmerulesv2.pdf
11. http://www.itsyourturn.com/t_helptopic2030.html
12. http://www.wikihow.com/Play-Checkers
13. http://www.yourturnmyturn.com/rules/checkers.php
14. http://www.flyordie.com/games/help/checkers/en/games_rules_checkers.html
15. http://brainking.com/en/GameRules?tp=7
16. http://www.netintellgames.com/checkersrules.htm
17. http://www.pcmag.com/article2/0,2817,1161217,00.asp
18. http://www.howcast.com/videos/297-how-to-play-checkers/
19. http://www.gamblingsites.com/skill-games/checkers/
20. http://www.mundigames.com/multiplayer/checkers/rules/

Rule Set #1

Part 1

Checkers is played by two players. Each player begins the game with 12 colored discs. (Typically, one set of pieces is black and the other red.)
The board consists of 64 squares, alternating between 32 dark and 32 light squares. It is positioned so that each player has a light square on the right side corner closest to him or her.
Each player places his or her pieces on the 12 dark squares closest to him or her.
Black moves first. Players then alternate moves.
Moves are allowed only on the dark squares, so pieces always move diagonally.
Single pieces are always limited to forward moves (toward the opponent).
A piece making a non-capturing move (not involving a jump) may move only one square.
A piece making a capturing move (a jump) leaps over one of the opponent's pieces, landing in a straight diagonal line on the other side. Only one piece may be captured in a single jump; however, multiple jumps are allowed on a single turn.
When a piece is captured, it is removed from the board.
If a player is able to make a capture, there is no option -- the jump must be made. If more than one capture is available, the player is free to choose whichever he or she prefers.
When a piece reaches the furthest row from the player who controls that piece, it is crowned and becomes a king. One of the pieces which had been captured is placed on top of the king so that it is twice as high as a single piece.
Kings are limited to moving diagonally, but may move both forward and backward. (Remember that single pieces, i.e. non-kings, are always limited to forward moves.)

Rule Set #1

Part 2

```
Kings may combine jumps in several directions -- forward and backward -- on
the same turn.  Single pieces may shift direction diagonally during a multiple
capture turn, but must always jump forward (toward the opponent).
A player wins the game when the opponent cannot make a move.  In most cases,
this is because all of the opponent's pieces have been captured, but it could
also be because all of his pieces are blocked in.
```

Rule Set #2

Part 1

```
In most games of checkers, there are two players.  The players are at opposite
ends of the board.  One player has dark pieces, and one player has light
pieces.  They take turns moving their pieces.  Players move their pieces
diagonally from one square to another square.  When a player jumps over their
opponent's (the other player's) piece, you take that piece from the board.

English checkers.
Most English-speaking people call English checkers "draughts".  English
'checkers' is played on an 8x8 chess board.  Only the dark squares are used
(the light squares are never used).  For that reason, good players play
differently in the left and right corners.

Pieces.
The pieces are flat and round.  They are referred to as "men".  They are
usually colored red and white.  For this reason, the darker pieces are usually
called "Red" and the lighter pieces are always called "White." Some checkers
sets have red and black pieces.  Then the red pieces are called "White" and
the black pieces "Red." And many sets simply use black and white draughts.
There are two kinds of pieces: plain (single) pieces and "kings".  A king is
made by putting one plain piece on top of another.

Starting position.
Each player starts with 12 pieces on the three rows closest to their own side.
The row closest to each player is called the "King␣Row".  The darker colour
moves first.
```

Rule Set #2

Part 2

```
How to move.
A player can move in two ways.  A piece can be moved forward, diagonally, to
the very next dark square.  In some variants, if one player's piece, the other
player's piece, and an empty square are lined up, then the first player must
"jump" the other player's piece.  In this case, the first player jumps over
the other player's piece onto the empty square and takes the other player's
piece off the board.  However, this is an uncommon ruleset not commonly
observed in the Americas.  A player can also use one piece to make multiple
jumps in any one single turn, provided each jump continues to lead immediately
into the next jump and in a straight line.  Sometimes a player may have the
option or a choice of which opponent piece he must jump.  In such cases, he
may then choose which to jump.  If you keep your hand on any piece when you're
moving, you have the choice to put it back and move another piece.
```

Rule Set #2

Part 3

Kings.
If a player’s piece moves into the King Row on the other player’s side, it becomes a king. It can move forward and backward. (Regular pieces can only move forward.) A king cannot jump out of the King Row **until** the next turn. Unlike Regular pieces, Kings can "jump" various empty boxes at a time to capture a regular piece. These "King␣Jumps" may only occur in diagonally aligned boxes. Neither Kings nor regular pieces may move in any direction that is not diagonal.

How the game ends.
The first player to lose all of his or her pieces loses the game. If no players are able to move, the player with the most amount of pieces wins. If the players have the same amount of pieces, the player with the most kings wins. If the players have an equal number of pieces and the same number of kings the game is a draw.

Generated ZRF for Rule Set #1

Part 1

```
(define PIECE-MOVE      ($1 (verify empty?)
        (if (in-zone? KING-transition)
                (add KING)
        else
                add
        )
))

(define PIECE-JUMP      ($1 (verify empty?)
        (if (in-zone? KING-transition)
                (add KING)
        else
                add
        )
))

(define KING-MOVE       ($1 (verify empty?)
                add

))

(define KING-JUMP       ($1 (verify empty?)
                add

))
```

Generated ZRF for Rule Set #1

Part 2

```
(game
        (title "checkers1")
        (players P1 P2)
        (turn-order P1 P2)
        (move-priorities MOVE JUMP)

        (board
                (grid
                        (dimensions
                                ("a/b/c/d/e/f/g/h") ; columns
                                ("8/7/6/5/4/3/2/1") ; rows
                        )
                        (directions
                                (n 0 -1) (w -1 0) (s 0 1) (e 1 0)
                                (ne 1 -1) (nw -1 -1) (se 1 1) (sw -1 1)
                        )
                )
                (symmetry P2 (n s) (s n) (ne sw) (sw ne) (nw se) (se nw))
                (zone (name KING-transition) (players P1)
                        (positions h8 g8 f8 e8 d8 c8 b8 a8)
                )
                (zone (name KING-transition) (players P2)
                        (positions h1 g1 f1 e1 d1 c1 b1 a1)
                )
        )
```

Generated ZRF for Rule Set #1

Part 3

```
(board-setup
        (P1 (PIECE g1 e1 c1 a1 h2 f2 d2 b2 g3 e3 c3 a3) )
        (P2 (PIECE h6 f6 d6 b6 g7 e7 c7 a7 h8 f8 d8 b8) )
)

(piece
        (name PIECE)
        (moves
                (move-type MOVE)
                (PIECE-MOVE nw)
                (PIECE-MOVE ne)

                (move-type JUMP)
                (PIECE-JUMP nw)
                (PIECE-JUMP ne)

        )
)
(piece
        (name KING)
        (moves
                (move-type MOVE)
                (KING-MOVE nw)
                (KING-MOVE ne)
                (KING-MOVE sw)
                (KING-MOVE se)
```

Generated ZRF for Rule Set #1

Part 4

```
                    (move-type JUMP)
                    (KING-JUMP nw)
                    (KING-JUMP ne)
                    (KING-JUMP sw)
                    (KING-JUMP se)

          )
    )

    (loss-condition (P1 P2 ) stalemated )
    (loss-condition (P1 P2 ) (pieces-remaining 0) )

)
```

Generated ZRF for Rule Set #2

Part 1

```
(define PIECE-MOVE      ($1 (verify empty?)
        (if (in-zone? KING-transition)
                (add KING)
        else
                add
        )
))

(define PIECE-JUMP      ($1 (verify empty?)
        (if (in-zone? KING-transition)
                (add KING)
        else
                add
        )
))

(define KING-MOVE       ($1 (verify empty?)
                add

))

(define KING-JUMP       ($1 (verify empty?)
                add

))
```

Generated ZRF for Rule Set #2

Part 2

```
(game
        (title "checkers2")
        (players P1 P2)
        (turn-order P1 P2)
        (move-priorities MOVE JUMP)

        (board
                (grid
                        (dimensions
                                ("a/b/c/d/e/f/g/h") ; columns
                                ("8/7/6/5/4/3/2/1") ; rows
                        )
                        (directions
                                (n 0 -1) (w -1 0) (s 0 1) (e 1 0)
                                (ne 1 -1) (nw -1 -1) (se 1 1) (sw -1 1)
                        )
                )
                (symmetry P2 (n s) (s n) (ne sw) (sw ne) (nw se) (se nw))
                (zone (name KING-transition) (players P1)
                        (positions h8 g8 f8 e8 d8 c8 b8 a8)
                )
                (zone (name KING-transition) (players P2)
                        (positions h1 g1 f1 e1 d1 c1 b1 a1)
                )
        )
```

Generated ZRF for Rule Set #2

Part 3

```
(board-setup
        (P1 (PIECE g1 e1 c1 a1 h2 f2 d2 b2 g3 e3 c3 a3) )
        (P2 (PIECE h6 f6 d6 b6 g7 e7 c7 a7 h8 f8 d8 b8) )
)

(piece
        (name PIECE)
        (moves
                (move-type MOVE)
                (PIECE-MOVE nw)
                (PIECE-MOVE ne)

                (move-type JUMP)
                (PIECE-JUMP nw)
                (PIECE-JUMP ne)

        )
)
(piece
        (name KING)
        (moves
                (move-type MOVE)
                (KING-MOVE nw)
                (KING-MOVE ne)
                (KING-MOVE sw)
                (KING-MOVE se)
```

Generated ZRF for Rule Set #2

Part 4

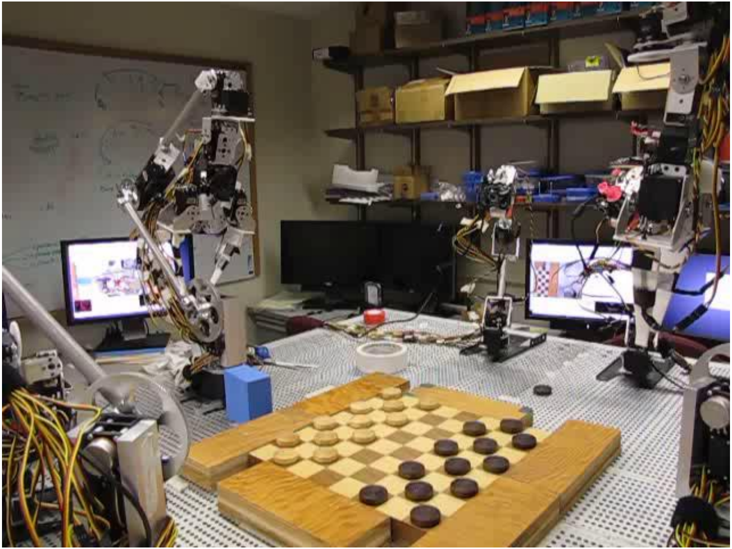
```
                    (move-type JUMP)
                    (KING-JUMP nw)
                    (KING-JUMP ne)
                    (KING-JUMP sw)
                    (KING-JUMP se)

          )
     )

     (loss-condition (P1 P2 ) (pieces-remaining 0) )

)
```

Two Robots Playing from Rule Sets #1 and #2



Two Robots Playing from Rule Sets #1 and #2