Advances in Cognitive Systems 6 (2018) 1-20 Submitted 6/2018; published 8/2018

Monitoring Scene Understanders with Conceptual Primitive Decomposition and Commonsense Knowledge

Leilani H. Gilpin LGILPIN@MIT.EDU
Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA, 02139 USA

Jamie C. Macbeth JMACBETH@SMITH.EDU
Department of Computer Science, Smith College, 100 Green Street, Northampton, MA 01063 USA

Evelyn Florentine EVELYNF@MIT.EDU
Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA, 02139 USA

Abstract

Although there have been many key advancements in connecting text and perception, computer-generated image captions still lack common sense. As a first step towards constraining these perception mechanisms to commonsense judgment, we have developed *reasonableness monitors*: a wrapper interface that can explain if the descriptive output of an opaque deep neural network is reasonable. These monitors are a stand-alone system that use careful dependency tracking, commonsense knowledge, and conceptual primitives to explain a perceived scene description to be reasonable or not. If such an explanation cannot be made, it is evidence that something unreasonable has been perceived. The development of reasonableness monitors is work towards generalizing that vision, with the intention of developing a system-construction methodology that enhances robustness at runtime (not static compile time), by dynamic checking and explaining of the behaviors of scene understanders for reasonableness in context.

1. Introduction

Recent advances in deep learning and machine learning have demonstrated vast improvements in automated systems which perform tasks such as image labeling and captioning. These systems are *perceptual* in the sense that their inputs are raw image sensor data and their outputs are judgments about the objects in the image, their spatial relations, their movements, and activities in the scene. Although these systems are often termed "scene understanders," recent well-documented evidence of adversarial examples indicates that many of these systems are not actually perceiving and understanding scenes using the kinds of commonsense reasoning that people use, but they are performing image understanding and labeling by exploiting patterns in large data sets that are not generalizable (Jia & Liang, 2017; Moosavi-Dezfooli et al., 2016; Nguyen et al., 2015; Szegedy et al., 2013).

Over time, autonomous systems such as unmanned aerial vehicles, self-driving cars, or domestic or industrial robots are adopting these technologies to make more actions and decisions previously entrusted to humans. Their misperceptions of their environment can present a serious challenge to the proper functioning of these systems, as well as a major safety issue. For example, a perception system embedded in a self-driving car architecture could mistake the identity of one or more objects in a scene, the background, the location, or the direction of movement or orientation of objects. Many misperceptions can be conveyed completely by the natural language text of the scene description that is generated, and many scene descriptions, for example, "a mailbox is crossing the street" or "an elephant is flying through the sky", can be judged as highly unlikely and nonsensical by a human without ever seeing an image of the original scene.

One way to address these drawbacks of contemporary data-driven, machine learning-based perceptual systems is to take a "society of mind" approach (Minsky, 1988) and view them as part of a larger system of cooperating autonomous agents. Monitoring systems are agents that can be attached to existing deployed systems to make them work slightly better by performing a limited augmentation that detects and addresses rare and unusual misbehaviors. This suggests the potential utility of a *reasonableness monitor*, a cognitive system endowed with commonsense knowledge which can form judgments of the reasonableness of the perceptions of other agents using only the natural language description of the perception.

In this paper, we present a prototype of a commonsense reasonableness monitoring system. It takes a natural language sentence as input, infers a reasonableness judgment of the sentence, and offers as output an explanation of that reasonableness judgment in natural language. As of now, the current system is developed to focus on input sentences describing physical acts and events performed by and on physical objects which are both animate and inanimate, to represent descriptions of scenes and situations that a perceptual image understanding or scene understanding system would produce. The prototype system uses a standard part-of-speech tagger on the input sentence and uses a commonsense knowledge base to represent the act or event described representing concepts as complex combinations of conceptual primitives. The system then applies constraints based on the conceptual primitive representation to determine the reasonableness of the original sentence and to generate a natural language explanation of the reasonableness judgment.

This paper proceeds as follows: In the next two sections, we provide general theory and background on integrating perception and reasoning in cognitive systems, and on building reasonableness monitoring systems in real-world environments. This motivates an example showing the implementation of our reasonableness monitor prototype in Section 4. In Section 5, we describe our reasonableness monitor architecture in full, and motivate the use of a commonsense knowledge base and conceptual primitive decomposition for enforcing constraints. We describe a set of studies of the reasonableness monitor and show the study results in Section 6. Section 7 reviews prior and related work in monitoring, commonsense reasoning, and perceptual algorithm understanding, and in Section 8 we conclude the paper with a discussion of limitations and future work.

2. Background: Integrating Perception and Reasoning

The integration of perception, recognition, and higher reasoning capability is a hallmark of human intelligence modeling, from Gestalt psychology (Michotte, 1963) and early machine vision systems (Roberts, 1963; Ullman, 1989) to cutting-edge standardized models of human cognition (Laird et al., 2017). Human cognition integrates perception and "pure" object recognition with reasoning and symbolic manipulation, allowing for both "bottom-up" and "top-down" processing of sensory inputs. Standard cognitive models suggest that a key part to this integrated process could be components which take raw perceptions that have been transformed to symbolic representations of recognizable objects and reason about them. The products of reasoning processes may be fed back into the recognition, which may result in confirming or disagreeing with the perception and, where perceptions are fuzzy or uncertain, it may transform the perceptions to be more in agreement with reasoning processes.

Some cognitive theories emphasize the strong influence of symbolic processes on perception. For example, while scientific evidence supports the view that normal perceptual experiences may rely more on knowledge than modes of sensation (Traynor, 2017), the reality of cognitive penetrability is the subject of significant debate in philosophical circles (Lammers et al., 2017). Winston's (2012) "inner" language hypothesis states that the internal language that humans use to construct complex symbolic descriptions of situations, knowledge, and events enables humans to shape perceptions and marshal perceptual systems in service of understanding and problem solving. A major part of this work to create a reasonableness monitor is devoted to integrating subsystems to represent the "inner", physical, non-linguistic representation and reasoning domains of mental models (Johnson-Laird, 1983) and imagery (Pearson & Kosslyn, 2015) that are theorized to be distinct from humans' "outer" language of familiar lexical items and linguistic forms.

3. Implementing Reasonableness Monitors in Real-World Perceptual Environments

How do we build systems that connect perception with knowledge and reasoning capability to help ensure developers, debuggers, or designers that their machines are perceiving and acting reasonably? And further, how do we facilitate the study of these systems in real-world environments? One may employ a modular approach, connecting existing perceptual systems to reasoning systems with representations of knowledge about commonly-perceived objects and their interactions. This requires components with particular characteristics, and raises certain general challenges to connecting them together to create a reasonableness monitor. Here we list factors that we considered in making specific choices in constructing a reasonableness monitor prototype and connecting it to working perceptual systems:

1. The output of a perceptual system will serve as an input to the reasonableness monitoring system. For the prototype reasonableness monitor described in this paper, we targeted image captioning and scene description systems as representations of machine vision and, more generally, perception systems that may be used in autonomous vehicles.

2. Perception should abide by the rules of commonsense. This requires a subsystem with a wealth of *structured* world knowledge that is indexed in a way such that knowledge that is

relevant to the perception can be found efficiently. The world knowledge can be a commonsense knowledgebase that represents "reasonable", sensible, or normal perceptions to be compared the perceptual system's behaviors.

3. Because commonsense knowledgebases typically represent objects and their interactions as words or phrases in natural language, and available perceptual systems are trained to produce language symbols as recognition outputs, natural language can be used as the interface between them. This may require subsystems that parse or process natural language in some way when it serves as an interface between these or other components.

4. However, we believe that a major challenge with using natural language as an interface is *linguistic variation*—the fact that are a myriad of different ways of expressing the same concept in "outer" language (Walker, 2010). This may also affect knowledgebases and semantic networks when knowledge in them is expressed using natural language terms. This underscores the need for inner language ontology representations for data and knowledge.

5. A reasoning system is needed to construct a model of the perception in the inner language ontology representation, which incorporates knowledge about the perceived objects and a set of rules or constraints that govern their typical interactions. It makes a reasonableness judgment by identifying confluences between perception and reason, or by identifying divergences between them in the form of rule conflicts and constraint violations.

6. Because engineers, system designers, and scientists need to be able to interact with the monitor and understand its output, we require that the system be able to *explain* and verify what it has done by *describing* the core reasons and support for a reasonableness judgment. The reasoner either finds that all constraints are met and it displays the premises supporting those constraints, or it explains the contradictory premises.

In the following example, we show how we use the output of a machine perception algorithm—a natural language description—and evaluate the output for reasonableness. We put this description into a reasonableness monitor, which transforms the description into a set of conceptual primitive frames, builds evidence by efficiently searching for relevant knowledge from ConceptNet, and then explains the judgment of reasonableness.

4. Reasonableness Judgments: An Example

In this section we provide a step-by-step example of how our prototype system processes an input perception description, "the mailbox crossed the street," determines it to be unreasonable, and outputs an explanation of its judgment of unreasonableness.

The input of the reasonableness monitor is a perceived scene description sentence that contains (at least) a subject and verb. The system performs a part-of-speech tagging of the input which is shown in Figure 1. Currently we use the Python NLTK part-of-speech tagger and a regex parser to find the noun phrase, verb phrase, objects, and prepositional phrases.

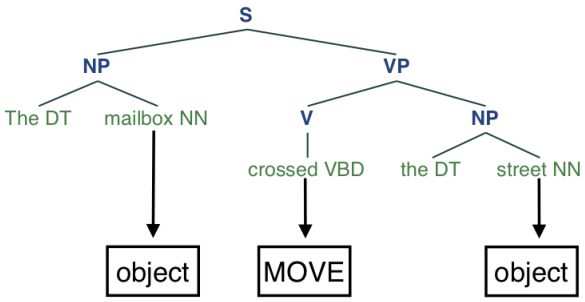


Figure 1. Parse tree and anchor point bindings for the input example, "A mailbox crossing the street".

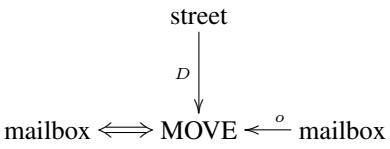
4.1 Anchoring

The system queries a commonsense knowledge base for knowledge related to the concepts in the parsed sentence. The system uses this knowledge to construct a frame around an abstracted conceptual primitive which we will use to apply reasonableness constraints.

In our prototype system we use ConceptNet (Speer & Havasi, 2013), a freely-available semantic network and commonsense knowledge base, to construct a frame representing a primitive act using the subject, object, and verb of the parsed sentence. The system constructs Trans-frames (Minsky, 1988) in Conceptual Dependency (Schank, 1972), which works to represent concepts as complex combinations of language-free conceptual primitives for an inner language ontology. Conceptual Dependency (CD) represents acts, events, spatial relationships, and changes of state using a small number of abstract primitives. The CD primitives we use in the system are fully explained in Sections 3.2 and 3.3.

To build a CD conceptualization corresponding to the input, the system first determines which primitive best represents the act described in the sentence, using the commonsense knowledge in ConceptNet. ConceptNet represents its knowledge as a graph with concepts as nodes and relations between concepts as edges. We select certain concept nodes in ConceptNet to act as *anchors*—the best representatives of the non-linguistic primitives of Conceptual Dependency. We determine which primitive to use to construct the CD conceptualization transframe by querying ConceptNet to determine if there is an edge relation between the verb and the anchor for that primitive. In this case, there is an edge in ConceptNet between "cross", which was tagged as the verb in the input, and "move", the system's anchor node for the PTRANS and MOVE primitives of Conceptual Dependency. PTRANS (short for Physical TRANSfer) is used to represent events where an object, thing, or substance changes location, while MOVE represents events where an animate object moves a part of its body. Our system merges these two primitives as MOVE-PTRANS, which we abbreviate to MOVE.

The system then instantiates a MOVE primitive act frame, which has slots for an actor, object, and a direction case. In this case, both the actor slot and the object slot are filled by the subject of the input sentence, and the direction case by the object of the sentence. This results in the CD conceptualization shown in Figure 2.



Word	POS Tag	ConceptNet Anchor	CD Role	Constraint Violations
mailbox	NN	'object'	actor	not animate
crossed	VBD	'move'	MOVE	N/A
street	NN	'object'	object	none

Figure 2. (Left) A Conceptual Dependency diagram representing the statement "The mailbox crossed the street." MOVE is the CD act primitive, double arrows point from MOVE to the actor, and single arrows marked "o" and "D" indicate the object and the directional case respectively. (Right) The part-of-speech tags, anchors, and roles in the CD MOVE primitive act frame for words in the input statement. Violations of constraints are determined based on the ConceptNet anchor and the role of the word in the conceptual decomposition.

4.2 Applying Conceptual Primitive Constraints

Now that the system has a populated primitive conceptualization frame, it can determine violations of the frames constraints by its constituents. Firstly, the MOVE primitive requires that the actor role must be filled by an "animate" object that can make other objects move or change location. Here the actor role is filled by "mailbox". To determine whether it satisfies or violates the constraint on the actor role, the system attempts to anchor the mailbox concept to a concept in ConceptNet to determine if it is an animate object or not. In this case there is a path through ConceptNet's graph between "mailbox" and "object", the anchor for inanimate objects, and no path between "mailbox" and "animal", the anchor for animate objects. So the system determines that "mailbox" is inanimate and flags a violation of the actor constraints for the MOVE primitive act.

The system now turns to the object role. The MOVE primitive requires that the object role must be filled by a physical object, thing, substance, or person. The object role for MOVE may be animate or inanimate, but it must also be an object that is non-stationary and can be moved. In this case the object role is also filled by "mailbox". Because the mailbox concept is anchored to "object" in ConceptNet, it satisfies the constraint on the object role.

Finally, the direction role for MOVE is constrained to represent a direction in reference to a physical object or a physical location. In this case, "street" is anchored to "object", so no constraint is violated. Although the verb in the sentence is "cross", which implies moving towards the street, on it, and then past it, the system reduces "crossing" to a single MOVE act.

4.3 Providing Explanations of Reasonableness and Unreasonableness

Based on the violation of the actor constraint on the MOVE frame, and because no other information was present in the sentence to recover from this violation (e.g. by introducing a concept other than "mailbox' to serve as the actor role), the system concludes that the statement is unreasonable.

The system then uses the concepts in the original sentence along with the anchor points and MOVE frame roles and role violations to construct a detailed natural language explanation of its judgment of unreasonableness. It generates a sentence stating the specific violation of the actor constraint by the mailbox, and a sentence stating that the original input sentence was unreasonable:

```
A mailbox is an object or thing that cannot move on its own.
So it is unreasonable for a mailbox to cross the street.
```

5. Reasonableness Monitor Architecture

Reasonableness monitors are wrappers around the subsystems or components of a machine that check their behavior for reasonableness. Reasonableness monitors build an explanation of a problem (or reasonable state) by examining the premises supporting the observation of a contradiction (or consistency). Monitors search through a knowledge base for premises and generate explanations of inconsistencies. The premises are used as evidence for explaining inconsistent (or consistent) information.

5.1 Input Parsing

The input of the reasonableness monitor is a perceived scene description that contains, at a minimum, a subject, or actor of the sentence, and verb. The first thing that the monitor does is parse the description and find the relevant concepts and primitives. The Python NLTK part-of-speech (POS) tagger is used to tag each element of the input. From there, a regular expression-based parser maps these POS tags to specific noun, verb and object phrases.

5.2 ConceptNet and Conceptual Primitive Decomposition

The kinds of image and video scene descriptions that are important for a reasonableness monitor to handle involve dynamic acts and events—people and objects moving and interacting with other people and objects against a background or at a particular location. Each reasonableness monitor has its own knowledge base: a set of behaviors that are considered to be reasonable. For monitoring in this context, ConceptNet 5, a semantic network and commonsense knowledge base (Speer & Havasi, 2013, 2012), is used as a knowledge base of reasonableness. We chose ConceptNet because it is freely available and has several million commonsense knowledge assertions about people and everyday objects, activities, and events.

However, like most commonsense knowledge bases, ConceptNet contains mainly "positive" assertions and little "negative" knowledge (Minsky, 1994) that could be use to deduce unreasonableness. It also does not have a facility for constructing a structure that combines concepts into representations of acts or events to facilitate reasoning about them through the knowledge in ConceptNet. Finally, while ConceptNet contains useful knowledge in a semantic form, it does not conform to a strict ontological hierarchy.

Our approach to solving these problems is to add inner language ontology structures by processing the input texts into abstract role frames representing animate and inanimate objects and their interactions in the scene. The role frames are combined with knowledge from ConceptNet about the concepts placed in each role, and used to apply constraints on reasonableness. We chose to construct role frames using primitives of Schank's Conceptual Dependency (CD) because the primitives are abstract and because they claim to represent physical acts in a universal, language-free conceptual base (Schank, 1972). The CD primitives are also small in number, and only six "physical" primitives were needed in our prototype system.

5.3 Conceptual Primitive Frames

Conceptual Dependency offers constraints based on the ways that conceptualizations are formed from the conceptual primitive, and the conceptual "cases", which include the actor, the object, and direction cases. In this section we provide descriptions of the six "physical" CD primitive acts and their constraints on reasonableness. In the following, a "thing" can refer to an object, substance, person, animal, or vehicle, and may be either an animate actor or inanimate.

- *PTRANS*: The PTRANS act primitive represents the event of a thing changing location from one place to another. The PTRANS act typically has an object case, representing the thing which moved or was moved, an actor case representing the actor which performed or caused the movement, and a direction case indicating the start and end point of the movement.
- *MOVE*: The MOVE act primitive represents the event of a thing moving a part of its body or part of itself. The MOVE act has an object case, representing the body part that was moved, an actor case representing the actor which performed the MOVE, and a direction case indicating the start and end point of the movement.
- *PROPEL*: The PROPEL act primitive represents the event of a thing applying a force to another thing, or a moving thing striking or impacting another thing. The PROPEL act typically has an object case, representing the object which was struck or has a force applied to it, an actor case representing the actor which performed or caused the PROPEL, and a direction case indicating the direction of the force.
- *INGEST*: The INGEST act primitive represents the event of a thing moving, being forced, or forcing itself to go from the outside to the inside of another thing.

  The INGEST act has an object case, representing the thing which moved or was moved to the inside of another thing, an actor case representing the actor which performed or caused the movement, and a direction case indicating the start and end point of the movement. Often the end point of an INGEST is a part of the body of the actor. Eating, for example, is an INGEST of something where the end point of the object's movement is the mouth or stomach of the actor.
- *EXPEL*: The EXPEL act primitive represents the event of a thing moving, being forced, or forcing itself to go from the inside to the outside of another thing. The EXPEL act has an object case, representing the thing which moved or was moved from inside to the outside of another thing, an actor case representing the actor which performed or caused the movement, and a direction case indicating the start and end point of the movement. Often the start point of an EXPEL is a part of the body of the actor. If a surgeon removes a bullet, a tumor, or a parasite from another person's body, however, the surgeon is the actor, but the start point of the movement of the object is a body part of another individual.
- *GRASP*: The GRASP act primitive represents the event of a thing grasping or becoming attached to another thing. The GRASP act has an object case, representing the thing which is being GRASPed, and an actor case representing the actor which performed the GRASPing.

PTRANS and MOVE are very similar primitives because they both involve movement from one place to another: for PTRANS an entire thing moves, while for MOVE, an animate thing only moves part of its body. In building our prototype system we found it difficult to find ConceptNet anchors allowing us to determine whether a verb should instantiate a PTRANS or MOVE. Because of this, we chose to combine the PTRANS and MOVE primitives into a single primitive, which we call MOVE-PTRANS, or simply MOVE. For the remainder of this paper, any referral to MOVE is the MOVE-PTRANS primitive.

5.4 Building Conceptual Primitive Frames

A major part of our reasonableness monitor processes the natural language input to construct a Conceptual Dependency transframe using knowledge present in ConceptNet about the verb of the sentence. It then applies further ConceptNet knowledge to apply reasonableness constraints to the concepts that take on roles in the CD transframe. To perform these functions we devised a system of *anchor points*: nodes in the ConceptNet semantic network that we have assigned to serve as broad categorizations that represent CD primitive acts and the constraints on concepts that take on roles in a CD transframe.

5.4.1 Anchor Points for Primitive Acts

As a first step to building a Conceptual Dependency transframe that best corresponds to the event expressed in the input, the system needs to determine which CD primitive to use. It does this by searching for paths between the verb and the anchor points in ConceptNet. The anchor points used for selecting the conceptual primitive act based on the verb are shown in Table 1. A concern with this method is that ConceptNet nodes are identified by words and phrases in "outer" natural language, while CD primitives are meant to be inner-language conceptual representations; care was taken to choose anchor points to be best representatives of of the conceptual primitive, and for some primitives multiple anchor points are used.

For example, for the MOVE primitive, "move" and "action" are used as verb-to-primitive anchors. The specific name of the conceptual primitive is not always used, and sometimes we used certain words as anchor points because they were better represented in ConceptNet. For example, the verb "ingest" has very few edges in the ConceptNet network, so instead we use "eat", "drink", and other words with similar meanings for the INGEST anchor points. A table of the anchor points for each CD primitive can be found in Table 1.

Our system queries ConceptNet using the stemmed and lemmatized form of the verb, searching for paths from the verb to the anchor point representatives of the CD primitives. The verb is anchored to the closest anchor point in terms of `IsA` hops in ConceptNet's semantic network. The system then instantiates a CD transframe of the corresponding primitive to represent the event.

5.4.2 Anchor Points for CD Transframe Roles

Once the conceptual primitive frame is instantiated, a different set of anchor points is used on the concepts filling the actor, object and direction roles of the frame to determine if they satisfy or

Table 1. The Conceptual Dependency (CD) "physical" primitives used, and the ConceptNet anchor points used to bind them to incoming verbs.

CD Primitive	Anchor Point(s)
INGEST	eat, drink, ingest
EXPEL	expel
GRASP	grasp, grab
MOVE-PTRANS	move, action, go
PROPEL	propel, hit

violate the frame's constraints. Using the definitions of each CD primitive frame, we can set unique constraints for the subject and object of our sentence.

The reasonableness monitor uses six anchor points for these roles: person, plant, animal, object, vehicle, and weather. These anchor points were chosen for two reasons. The first reason is that they fit our use case for autonomous vehicles. The second, more significant, reason is that each anchor point is broad enough to include a variety of items, but just restrictive enough so that each anchor point has different properties that allow it to perform certain actions.

For example, an animal can move on its own, and thus it can serve as the actor role in a MOVE-PTRANS CD transframe, but an object cannot move on its own, unless it is a vehicle. We assume that vehicles are controlled by humans and therefore they can move, so we also have an anchor point so that cars or other automobiles will not be categorized as objects, but as vehicles specifically.

Table 2. List of ConceptNet anchor points used for actor and object roles in the CD transframe, and constraints on where a concept may reasonably serve in the role. The Actor Constraints column lists the CD primitives where a concept bound to the anchor point may serve in the actor role for that primitive, while the Object Constraints column lists the CD primitives where a concept bound to the anchor point may serve in the object role. "None" appears in cases where a concept bound to the anchor point may never appear in the role. Constraints for the direction case are described in the text and not shown here.

Anchor Point	Actor Constraints	Object Constraints
person	EXPEL, GRASP, INGEST, MOVE, PROPEL	GRASP, MOVE, PROPEL
animal	EXPEL, GRASP, INGEST, MOVE, PROPEL	GRASP, INGEST, MOVE, PROPEL
plant	none	GRASP, INGEST, MOVE, PROPEL
object	GRASP	GRASP, INGEST, MOVE, PROPEL
place	none	none
weather	PROPEL	none
confusion	PROPEL	none
vehicle	EXPEL, GRASP, INGEST, MOVE, PROPEL	MOVE, PROPEL

5.5 Primitive Act Constraints

For the "physical" primitives described above, all primitive acts are subject to constraints that can be applied to the actor, object, and direction cases of the frame to determine reasonableness. For all of these primitives:

- The actor must be an "animate" object or thing capable of
    1. making other objects move (in the case of MOVE, INGEST, EXPEL);
    2. moving or applying a force in order to GRASP another object (in the case of GRASP);
    3. applying forces to other objects (in the case of PROPEL);
- The object must be a physical object, thing, substance, or person;
- The direction case should represent a direction in reference to a physical object or a physical location or place.

There are several additional constraints for particular primitive frames and the complete constraints for actor and object cases of the CD primitives are shown in Table 2. Based on its definition, each primitive imposes constraints on the types of anchor points that the subject and object can be categorized as. In order for the statement to be reasonable, both the subject and the object must share an edge with one of their respective permitted anchor points. If either of them do not share any edge with the permitted anchor points, there is a contradiction and the statement is deemed unreasonable.

In the earlier example, "a mailbox crosses the street", "mailbox" violated the constraint that an actor for MOVE be animate. Taking another example, for the statement "A man pushes the wind," the system creates a PROPEL primitive, and "man" will be categorized as a person anchor point, which fits the subject constraint. However, no edge exists between any of the permitted anchor points and the object, "wind", because wind is not a thing (a person, animal, vehicle, or object). Therefore the statement will be deemed unreasonable.

5.6 Compound Primitive Frames

It is also possible for the reasonableness monitor to construct a Conceptual Dependency transform using multiple primitives. For example, in an example like "Lisa kicked the ball," there are two primitive acts. Firstly, Lisa applies a force (PROPEL), and secondly, the ball is moved (MOVE-PTRANS). Although the system cannot automatically decompose this sentence into compound primitive frames, we can hard-code select verbs to be compound by default (instead of using ConceptNet or anchor points), so that kick decomposes into a PROPEL and MOVE-PTRANS. The explanations generated from these decompositions are seen in Section 6.4.

5.7 Establishing Context

It is possible for the context of a sentence to change its reasonability. For example, we consider that weather may change the reasonability of a statement, as it can easily change the CD primitive

chosen. To illustrate this, we will refer back to our example: "A mailbox crossed the street" (see Section 4). This was deemed unreasonable since "cross" is a MOVE-PTRANS type and a mailbox is an object, which conflicts with the actor constraints of MOVE-PTRANS. If instead the input were "A mailbox crossed the street in a hurricane," then our statement becomes more reasonable. An outside force, such as a hurricane, can move objects, which corresponds to the definition of PROPEL. Therefore, the CD primitive frame becomes a PROPEL rather than MOVE-PTRANS. Since the mailbox satisfies the constraints for PROPEL, this statement is now classified as reasonable.

The system also checks for prepositional phrases as added context for establishing reasonableness. When the sentence is parsed, prepositional phrases are identified and stored as contexts which are additional evidence in the CD primitive structure. In a case like "in a hurricane", the noun phrase within the prepositional phrase is bound to another anchor point, in this case "hurricane". There are also anchor points for extreme conditions that are hard coded, where ConceptNet is not used, i.e., hurricanes, earthquakes, tornadoes, and floods. Then, in the reasonableness checking phase, the monitor also examines this context to determine if the additional context can ameliorate a previously unreasonable description.

One important part of contextual knowledge we added into the system was capitalized names. In this system, we assume that capitalized names belong to people, so they are automatically assigned the person anchor point. We wanted to be sure that capitalized names would automatically be characterized as people, instead of searching through ConceptNet for a not-well-populated name.

6. Experimental Studies

We composed two studies to answer the following two research questions:

1. Can our monitoring system *judge* whether a perceived description is reasonable or not?

2. Can our monitoring system *explain* why a perceived description to be reasonable or not?

The first study evaluated the monitor's judgment of reasonableness. We constructed a test set of examples, hand-labeled if they were reasonable or not, and tested whether the system deemed them reasonable or not. Additionally, we included sentences which change reasonability based on weather and vehicle rules. The test set solely assesses whether a statement correctly determines reasonableness, and does not analyze the justifications that the monitor outputs. A breakdown of failure cases is shown in Table 3.

In the second study, we answer how well the monitor can explain. In particular, we compare the explanatory results for both single primitives and compound primitives. We compared the explanations produced when an input statement is binded to a single CD primitive using anchor points and ConceptNet versus a hard-coded compound primitive. Comparisons and results of the explanations are shown in Section 6.4.

6.1 Data Set

We constructed a test set and composed two studies to evaluate the monitoring system for its judgment of reasonableness, and its explanation of reasonableness. Our data set is inspired by visual

Table 3. A analysis of the causes of the 18 misclassifications on the 100 test cases.

		Classify as:	
		Reasonable	Unreasonable
Label as:	Reasonable		Parser: 2 ConceptNet: 8
	Unreasonable	Parser: 2 ConceptNet: 6	

scenes typically encountered in outdoor driving environments, and contains texts of both reasonable and unreasonable scene descriptions. We focused on descriptions generated in a driving environment because of the focus on *understanding* the objects and perceived actions. For this study, we steered away from image captioning system outputs because they are almost always reasonable; they are generated and trained on captions written by humans who are looking at real (and therefore, reasonable) images.

The purpose of this data set is not to test image captioning specifically, but machine perception generally. For that reason, we constructed our own data set of perception descriptions instead of using an existing data set. Out of the 100 example data set, 50 of the examples were clearly unreasonable (e.g. our previous example of a mailbox crossing the street), and the other half were clearly reasonable (e.g. a man eats food). Each example is composed of an average of 4.47 words.

The examples used 57 unique words, including 14 verbs, 35 nouns, and 8 articles, auxiliary verbs, and prepositions. 23 of the 100 sentences had prepositional phrases. We varied the subject, object, and verb choices to test how well our system categorizes the concepts and reasoning, and also to see how well ConceptNet categorizes the concepts into anchor points and primitives. This also tested whether typical concepts would be in ConceptNet, and whether they would be connected to the anchor points through a path in ConceptNet's network. We constructed the test set without any prior knowledge of how terms we used would be represented in ConceptNet.

6.2 Results Summary

We demonstrate two contributions of the system. First is the ability to deem a statement as reasonable or unreasonable. On our test set of 100 examples, the reasonableness monitor performed with 82% accuracy. Most of the errors that occurred were due to binding the incorrect anchor point at the ConceptNet level as demonstrated in Table 3. The second contribution is the ability to construct *human-readable* explanations that motivate and support the reasonable or unreasonable claim. These explanations are also represented as symbolic triples, which could plausibly be re-input into a scene understander to re-evaluate in the case of an unreasonable classification.

6.3 Study One: Resonableness Judgment for Primitive and Compound Decompositions

On our hand-curated test set, most of the errors that occurred were due to badly defined labels of the `IsA` relation in ConceptNet. For example, in the second example in Table 4, a direct ConceptNet

Table 4. A set of examples showing the single primitive decomposition, compound decomposition, anchor points, and reasonability judgment. Reasonability judgment is consistent whether the decomposition is single or compound. Note that MOVE is still a MOVE-PTRANS hybrid.

Statement	Single Primitive	Compound Primitive	Subject Anchor	Object Anchor	Reason -able
Monkey throws an apple	PROPEL	GRASP, PROPEL, MOVE	animal	object	Yes
A flower hits an apple	PROPEL	GRASP, PROPEL, MOVE	plant	object	No
A tree hits a car in a storm	PROPEL	PROPEL, MOVE	weather	plant	Yes
A car leaks gas	EXPEL	EXPEL	person	None	Yes
A man breathes out	EXPEL	EXPEL	person	None	Yes
A man eats food	INGEST	INGEST	person	object	Yes
A giraffe eats leaves	INGEST	INGEST	animal	plant	Yes
A boat eats a plant	INGEST	INGEST	vehicle	plant	No

search bound "flower" to the `person` anchor point instead of `plant`, which lead to an incorrect reasonableness judgment. This is because there are fewer links between flower and person than flower and plant via the `IsA` relation in ConceptNet. This motivating example shows the necessity for anchor points, and the importance of well-defined relations in a semantic, commonsense database. We repeated the single primitive study shown in Table 4, which demonstrated that with compound primitives can come up with the same reasonability judgments, with better descriptions.

6.4 Study Two: Explanation Examples

Take the following three observations: "A dog crossed the street", "A wall crossed the street", and "A mailbox crossed the street during an earthquake". These are determined to be reasonable, unreasonable and reasonable respectively. The generated explanations provide evidence in Table 5.

However, the object role in a MOVE primitive is not limited to animate objects that move themselves. Inanimate objects can also be propelled by concepts with strong forces, like hurricanes or earthquakes. If we add in this kind of information, in the form of a prepositional phrase, then this is taken as context, and the monitor detects a reasonable state.

To demonstrate the benefits of compound primitive decompositions, we show a side-by-side comparison of a compound decomposition and a single decomposition. The monitor will build up a series of constraints (in the case of an unreasonable state, in Table 7) or support (in the case of a reasonable state in Table 6) for each case.

6.5 Discussion and Implications

When this system was originally being developed, the idea was to explain blatantly unreasonable flaws in perception, e.g., if you saw an elephant in the sky, that observed perception is obviously unreasonable. However, if you saw an elephant in the sky but the focus or *context* was in the clouds or in the fog, then this added context, i.e., a different "world view", could explain away

Table 5. Comparison of explanation descriptions for single primitive decompositions.

A dog crossed the street.	A wall crossed the street.	A mailbox crossed the street during an earthquake.
Reasonable	Unreasonable	Reasonable
A(n) dog is an animal and animals can move. So it is reasonable for a dog to cross the street.	A(n) wall is an object or thing that cannot move on its own. So it is unreasonable for a wall to cross the street.	Although a mailbox cannot move on its own, an earthquake can propel a stationary object to move. So it is reasonable for a mailbox to cross the street during an earthquake.

an otherwise narrow, unreasonable situation. This realization lead the idea of different contexts and a new knowledge representation: "confusion" anchor points, where the sole purpose of this mechanism was to add more context-dependent knowledge that could explain away or additively confuse perceptions.

The key idea here is that monitoring should not be invasive; it should provide an additional set of quick "checks" to ensure more reliability and safety. Automobiles and their autonomous counterparts are engineering marvels, and they work quite well most of the time. The idea of reasonableness monitoring is to make them work better by removing blatantly unreasonable situations that can have bad consequences. But monitoring can also be used to alert for help and for developing better benchmarks for safety-critical decisions.

If such a system were deployed in a semi-autonomous or fully autonomous machine, it could alert a safety-supervisor or safety-driver to validate its possibly unreasonable perception. In machine-learning training situations, an reasonableness monitor could be part of the training and evaluation process of deep neural network perceptive systems. Modern complex systems work fairly well in practice, although they are unable to provide insights into their behaviors, especially in the case of their errors. A nice consequence of reasonableness monitors is that they identify the problematic evidence and deductions in the case of a contradiction. The overarching goal is to use monitoring in two ways (1) to explain errors after the fact for better diagnostics and (2) to use the problematic evi-

Table 6. Comparison of single primitive and compound primitive decomposition explanations for the reasonable description "A girl kicked the ball."

	Single Primitive	Compound Primitive
Primitive	PROPEL	PROPEL, MOVE-PTRANS
Explanation	A girl is a person that can apply a force on their own. Further, a ball is a physical object, thing or substance that can be propelled. So it is reasonable for a girl to kick the ball.	A girl is a person that can apply a force on their own. Further, a ball is a physical object, thing or substance that can be propelled. A ball is an object or thing that can be moved by kicking. So it is reasonable for a girl to kick the ball.

Table 7. Comparison of single primitive and compound primitive decomposition explanations for the unreasonable description "A flower hits an apple".

	Single Primitive	Compound Primitive
Primitive	PROPEL	GRASP, PROPEL MOVE-PTRANS
Explanation	A flower is an object or thing that cannot propel an object. So it is unreasonable for a flower to hit an apple.	A flower does not have the ability to grasp an object and a flower is an object or thing that cannot propel an object. Further, an apple cannot move on its own. So it is unreasonable for a flower to hit an apple.

dence to make better decisions next time. For the latter point, we want to be able to use the evidence from a monitor to feed back into an existing algorithm, resulting in better decisions, a decrease in false positives/negatives, and an increase in reliability.

7. Related Work

One goal of our monitoring system is to create safe, trustworthy autonomous systems made out of subsystems that themselves are made out of parts. This exemplifies the idea of a "multi-agent system," first coined in the Society of Mind (Minsky, 1988). In fact, Minsky uses the term "agent" to describe any component, subsystem, or part of a cognitive process that is simple enough to understand (Singh, 2012). Since no single method of problem solving will always work, Minsky suggested that we also need to know about pitfalls and corner cases. He encouraged the use of negative expertise in the form of censor and suppressor agents (Minsky, 1994). He explains that negative knowledge and examples are important to create intelligent systems, even suggesting that a way to implement negative knowledge is to divide a complex system into parts that can monitor each other, similar to our monitoring framework.

Roger Schank introduced Conceptual Dependency theory and its conceptual primitives for natural language understanding (Schank, 1972). Similar work in computational semantics, (Jackendoff, 1983), shows that it is necessary to represent these kinds of conceptual structures or thoughts and not simply study language in isolation. Wilks and Fass (1992) and Wierzbicka (1996) are other types of compositional primitive decompositions. Borchardt (1994) is another decomposition method based on a theory of 10 primitives that describe transition space change.

Commonsense knowledgebases are a key tool for developing systems that understand natural language descriptions and produce explanations. Although CYC is regarded as the world's longest-lived artificial intelligence project (Lenat et al., 1990), with a comprehensive ontology and knowledge base including basic concepts and "commonsense rules," there have been significant challenges to using CYC for NLP (Mahesh et al., 1996). Speer and Havasi (2013) demonstrate the usage of ConceptNet5, a freely-available semantic network of commonsense knowledge. Research on SenticNet (Cambria et al., 2018) was inspired by primitive decomposition theories (Schank, 1972), and links ConceptNet concepts to conceptual primitives that help generalize them to overcome linguistic variation.

Our methodology is also a first step towards interpreting deep neural networks by constraining the output to common sense. Zhang and Zhu (2018) outlined six techniques for interpreting deep convolutional neural networks for vision understanding. However, these methods are quite invasive; they require knowledge of all the components, and are driven towards understanding the focus of these opaque algorithms, rather that providing additional, meaningful knowledge.

Reasonableness monitors are a system-methodology to identify and explain anomalies in perception, using common-sense knowledge to determine the reasonableness of perception-derived scene descriptions (Gilpin, 2018). This work was extended to validate scene descriptions from an immersive virtual reality environment (Gilpin et al., 2018). By contrast, in the current paper, we show how using conceptual primitive decomposition with a monitoring system can provide succinct, convincing explanations of unreasonable (or reasonable) perceived scenes.

8. Conclusion

The reasonableness monitor system presented in this paper is designed to give autonomous perception systems commonsense by detecting the most important premises regarding a contradiction, thereby determining the reasonability of the perception. The system is targeted to monitor and evaluate the performance of scene understanding and machine vision systems by testing them and examining their inputs and outputs. The system is built around a commonsense knowledge base with millions of facts collected through crowdsourcing and other human effort. We also strengthened the explanations of this system through the use of conceptual primitives, which provide an inner language representation, allowing the system to apply reasonableness constraints and overcome occasional erroneous assertions in the crowdsourced knowledge base. Additionally, we added a testing framework and expanded the system to work in more complex scenarios. Ultimately, the reasonableness monitor can help autonomous vehicles justify their perceptions.

As autonomous vehicles become more prevalent, they need methods to explain their actions. Since these vehicles will soon be piloted (Drive.ai, 2018), there needs to be a system that can dynamically check the reasonableness of the machines' actions, with limited augmentation to the deployed system. In order to combat modern engineering problems, it is important that we view autonomous machines as "multi-agent systems": systems of interconnected components, subsystems, and parts working together towards a common goal. We must disentangle these complex systems into subsystems and parts that will be simple enough to understand (Singh, 2012). Further, it is important to consider redundancy in autonomous machines. There is no simple solution that will always work, and because of that we need to be constantly introspective and monitor the negative examples and corner cases.

Although the reasonableness monitoring system prototype we have presented works well on a plethora of examples, it has some limitations; mainly with the organization of the commonsense data. One limitation, as noted in the results section, is that not all the `IsA` relations are well-defined in ConceptNet. For example, in ConceptNet, nodes representing common food items that were once animals, like hamburgers, contain an `IsA` link to animal. Therefore, in the example of "A hamburger crossing the street", this is characterized as being reasonable since "a hamburger is an animal that can move on its own." Another limitation of the system is the parser framework. In

particular, we found that the NLTK parser system has trouble recognizing certain verbs that can also be nouns. Future work will explore using other parser tools such as the START Parser, which has been used successfully in question answering and story understanding systems (Morales et al., 2016; Winston, 2014).

Another next step is to test this system on machine-generated image captions from a data set such as Microsoft's Common Objects in Context (COCO) database (Lin et al., 2014). One challenge is that these captions are not necessary unreasonable in the sense that our monitor expects: they are often reasonable because they describe reasonable visual scenes, but incorrect because the description is not correlated with the input image. Another future of area of work is using the monitor to incorporate feedback into the underlying perceptual system. Once a machine's behavior has been deemed reasonable or not, how do we feed this evidence back into the system? Reasonableness monitors should make machine learning processes better; by feeding in reasonableness judgments as training data or as another type of test or validation method. This feedback mechanism could also serve as an evaluation of the system, by measuring the amount of improvement.

Our system is designed to determine reasonableness among existing computer vision system predictions and results. However, in the future, we hope to extend this to *full system design*, as a way to constrain all sensors and subsystems in an autonomous vehicle to a sense of reasonableness. There are two components necessary to extend this system to other agents. One is a "common-sense database" in terms of the level of abstraction of the subsystem. For example, the braking component will need a series of "normal" or "reasonable" braking patterns, probably in terms of signals. Similarly, higher level components, like the route planner, will need higher-level descriptions of reasonability, like the appropriate steps of a "reasonable" right turn, or traffic patterns in symbolic language. The second component is a set of inner language primitives. For the high-level planner, this may be symbolic planning actions, and for the braking components, this will be brake engagement. Together with these two components, we can generalize a system construction for an autonomous vehicle, with limited augmentation to the working system, that constrains the system to reasonableness for each component.

9. Acknowledgements

The authors acknowledge the support of the Toyota Research Initiative (TRI) and the Office of Naval Research Summer Faculty Research Program.

References

Borchardt, G. C. (1994). *Thinking between the lines: Computers and the comprehension of causal descriptions*. Cambridge, MA: MIT Press.

Cambria, E., Poria, S., Hazarika, D., & Kwok, K. (2018). SenticNet 5: Discovering conceptual primitives for sentiment analysis by means of context embeddings. *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*. New Orleans, LA: AAAI Press.

Drive.ai (2018). Drive.ai announces on-demand self-driving car service on public roads in Texas. From `goo.gl/G4oiyH`.

Gilpin, L. (2018). Reasonableness monitors. *Papers from the Twenty-Third AAAI/SIGAI Doctoral Consortium at AAAI-18*. New Orleans, LA: AAAI Press.

Gilpin, L. H., Zaman, C., Olson, D., & Yuan, B. Z. (2018). Reasonable perception: Connecting vision and language systems for validating scene descriptions. *Proceedings of the Thirteenth Annual ACM/IEEE International Conference on Human Robot Interaction*. Chicago, IL: ACM.

Jackendoff, R. S. (1983). *Semantics and cognition*. Cambridge, MA: MIT Press.

Jia, R., & Liang, P. (2017). Adversarial examples for evaluating reading comprehension systems. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (pp. 2021–2031). Copenhagen: ACL.

Johnson-Laird, P. N. (1983). *Mental models: Towards a cognitive science of language, inference, and consciousness*. Cambridge, MA: Harvard University Press.

Laird, J. E., Lebiere, C., & Rosenbloom, P. S. (2017). A standard model of the mind: Toward a common computational framework across artificial intelligence, cognitive science, neuroscience, and robotics. *AI Magazine*, *38*.

Lammers, N. A., de Haan, E. H., & Pinto, Y. (2017). No evidence of narrowly defined cognitive penetrability in unambiguous vision. *Frontiers in Psychology*, *8*, 852.

Lenat, D. B., Guha, R. V., Pittman, K., Pratt, D., & Shepherd, M. (1990). CYC: Toward programs with common sense. *Communications of the ACM*, *33*, 30–49.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. *European Conference on Computer Vision* (pp. 740–755). Zurich: Springer.

Mahesh, K., Nirenburg, S., Cowie, J., & Farwell, D. (1996). *An assessment of CYC for natural language processing*. Technical Report MCCS-96-296, Computing Research Laboratory, New Mexico State University, Las Cruces, NM.

Michotte, A. E. (1963). *The perception of causality*. New York: Basic Books.

Minsky, M. (1988). *Society of Mind*. New York: Simon & Schuster.

Minsky, M. (1994). Negative Expertise. *International Journal of Expert Systems*, *7*, 13–19.

Moosavi-Dezfooli, S., Fawzi, A., Fawzi, O., & Frossard, P. (2016). Universal adversarial perturbations. *ArXiv e-prints*, *abs/1610.08401*.

Morales, A., Premtoon, V., Avery, C., Felshin, S., & Katz, B. (2016). Learning to answer questions from Wikipedia infoboxes. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, TX: ACL.

Nguyen, A., Yosinski, J., & Clune, J. (2015). Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 427–436). Boston, MA: IEEE.

Pearson, J., & Kosslyn, S. M. (2015). The heterogeneity of mental representation: Ending the imagery debate. *Proceedings of the National Academy of Sciences*, *112*, 10089–10092.

Roberts, L. G. (1963). *Machine perception of three-dimensional solids*. Doctoral dissertation, Department of Electrical Engineering, Massachusetts Institute of Technology, Cambridge, MA.

Schank, R. C. (1972). Conceptual dependency: A theory of natural language understanding. *Cognitive Psychology*, *3*, 552–631.

Singh, P. (2012). Examining the Society of Mind. *Computing and Informatics*, *22*, 521–543.

Speer, R., & Havasi, C. (2012). Representing general relational knowledge in ConceptNet 5. *Proceedings of the Eighth International Conference on Language Resources and Evaluation* (pp. 3679–3686). Istanbul: European Language Resources Association.

Speer, R., & Havasi, C. (2013). ConceptNet 5: A large semantic network for relational knowledge. In *The people's Web meets NLP*, 161–176. New York: Springer.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I. J., & Fergus, R. (2013). Intriguing properties of neural networks. *ArXiv e-prints*, *abs/1312.6199*.

Traynor, R. (2017). Seeing-in-for-action: The cognitive penetrability of perception. *Proceedings of the Fifth Annual Conference on Advances in Cognitive Systems*. Troy, NY: The Cognitive Systems Foundation.

Ullman, S. (1989). Aligning pictorial descriptions: an approach to object recognition. *Cognition*, *32*, 193–254.

Walker, J. A. (2010). *Variation in linguistic systems*. New York: Routledge.

Wierzbicka, A. (1996). *Semantics: Primes and universals*. New York: Oxford University Press.

Wilks, Y., & Fass, D. (1992). The preference semantics family. *Computers & Mathematics with Applications*, *23*, 205–221.

Winston, P. H. (2012). The right way. *Advances in Cognitive Systems*, *1*, 23–36.

Winston, P. H. (2014). *The Genesis story understanding and story telling system: A 21st century step toward artificial intelligence*. Technical report, Center for Brains, Minds and Machines, Massachusetts Institute of Technology, Cambridge, MA.

Zhang, Q., & Zhu, S. (2018). Visual interpretability for deep learning: A survey. *ArXiv e-prints*, *abs/1802.00614*.