Visual Turing test for computer vision systems

Donald Geman[a], Stuart Geman[b,1], Neil Hallonquist[a], and Laurent Younes[a]

[a]Department of Applied Mathematics and Statistics, Johns Hopkins University, Baltimore, MD 21287; and [b]Division of Applied Mathematics, Brown University, Providence, RI 02912

Contributed by Stuart Geman, December 16, 2014 (sent for review July 14, 2014)

Today, computer vision systems are tested by their accuracy in detecting and localizing instances of objects. As an alternative, and motivated by the ability of humans to provide far richer descriptions and even tell a story about an image, we construct a "visual Turing test": an operator-assisted device that produces a stochastic sequence of binary questions from a given test image. The query engine proposes a question; the operator either provides the correct answer or rejects the question as ambiguous; the engine proposes the next question ("just-in-time truthing"). The test is then administered to the computer-vision system, one question at a time. After the system's answer is recorded, the system is provided the correct answer and the next question. Parsing is trivial and deterministic; the system being tested requires no natural language processing. The query engine employs statistical constraints, learned from a training set, to produce questions with essentially unpredictable answers—the answer to a question, given the history of questions and their correct answers, is nearly equally likely to be positive or negative. In this sense, the test is only about vision. The system is designed to produce streams of questions that follow natural story lines, from the instantiation of a unique object, through an exploration of its properties, and on to its relationships with other uniquely instantiated objects.

scene interpretation | computer vision | Turing test | binary questions | unpredictable answers

Going back at least to the mid-20th century there has been an active debate about the state of progress in artificial intelligence and how to measure it. Alan Turing (1) proposed that the ultimate test of whether a machine could "think," or think at least as well as a person, was for a human judge to be unable to tell which was which based on natural language conversations in an appropriately cloaked scenario. In a much-discussed variation (sometimes called the "standard interpretation"), the objective is to measure how well a computer can imitate a human (2) in some circumscribed task normally associated with intelligent behavior, although the practical utility of "imitation" as a criterion for performance has also been questioned (3). In fact, the overwhelming focus of the modern artificial intelligence (AI) community has been to assess machine performance more directly by dedicated tests for specific tasks rather than debating about general "thinking" or Turing-like competitions between people and machines.

In this paper we implement a new, query-based test for computer vision, one of the most vibrant areas of modern AI research. Throughout this paper we use "computer vision" more or less synonymously with semantic image interpretation—"images to words." However, of course computer vision encompasses a great many other activities; it includes the theory and practice of image formation ("sensors to images"), image processing ("images to images"), mathematical representations, video processing, metric scene reconstruction, and so forth. In fact, it may not be possible to interpret scenes at a semantic level without taking at least some of these areas into account, especially the geometric relationship between an image and the underlying 3D scene. However, our focus is how to evaluate a system, not how to build one.

Besides successful commercial and industrial applications, such as face detectors in digital cameras and flaw detection in manufacturing, there has also been considerable progress in more generic tasks, such as detecting and localizing instances from multiple generic object classes in ordinary indoor and outdoor scenes; in "fine-grained" classification, such as identifying plant and animal species; and in recognizing attributes of objects and activities of people. The results of challenges and competitions (4, 5) suggest that progress has been spurred by major advances in designing more computationally efficient and invariant image representations (6–8); in stochastic and hierarchical modeling (9–12); in discovering latent structure by training multilayer networks with large amounts of unsupervised data (13); and in parts-based statistical learning and modeling techniques (14–16), especially combining discriminative part detectors with simple models of arrangements of parts (17). Quite recently, sharp improvements in detecting objects and related tasks have been made by training convolutional neural networks with very large amounts of annotated data (18–22).

More generally, however, machines lag very far behind humans in "understanding images" in the sense of generating rich semantic annotation. For example, systems that attempt to deal with occlusion, context, and unanticipated arrangements, all of which are easily handled by people, typically encounter problems. Consequently, there is no point in designing a "competition" between computer vision and human vision: Interpreting real scenes (such as the ones in Fig. 1) is virtually "trivial" (at least effortless and nearly instantaneous) for people whereas building a "description machine" that annotates raw image data remains a fundamental challenge.

We seek a quantitative measure of how well a computer vision system can interpret ordinary images of natural scenes. Whereas we focus on urban street scenes, our implementation could easily be extended to other image populations and the basic logic and motivations remain the same. The "score" of our test is based on the responses of a system under evaluation to a series of binary questions about the existence of people and objects, their activities and attributes, and relationships among them, all relative to an image. We have chosen image-based rather than scene-based queries (*Images of Scenes*).

Suppose an image subpopulation $\mathcal{I}$ has been specified ("urban street scenes" in Fig. 1), together with a "vocabulary" and a corresponding set of binary questions (*Vocabulary and Questions*). Our prototype "visual Turing test" (VTT) is illustrated in

Significance

In computer vision, as in other fields of artificial intelligence, the methods of evaluation largely define the scientific effort. Most current evaluations measure detection accuracy, emphasizing the classification of regions according to objects from a predefined library. But detection is not the same as understanding. We present here a different evaluation system, in which a query engine prepares a written test ("visual Turing test") that uses binary questions to probe a system's ability to identify attributes and relationships in addition to recognizing objects.

Author contributions: D.G., S.G., N.H., and L.Y. wrote the paper; D.G. and S.G. conceived the idea and supervised the research; N.H. collected the data and programmed the algorithm; and L.Y. supervised the research.

The authors declare no conflict of interest.

Freely available online through the PNAS open access option.

[1]To whom correspondence should be addressed. Email: stuart_geman@brown.edu.

This article contains supporting information online at www.pnas.org/lookup/suppl/doi:10.1073/pnas.1422953112/-/DCSupplemental.

Fig. 1. Urban street scenes. *Top* row shows Athens, Baltimore, Busan, and Delhi. *Bottom* row shows Hong Kong, Miami, Rome, and Shanghai. (*Top*, *Left* to *Right*, second from *Left*) Images modified with permission from Doug Dawson (Photographer), Wikimedia Commons/Carey Ciuro, Wikimedia Commons/McKay Savage. (*Bottom*, *Left* to *Right*) Images modified with permission from Wikimedia Commons/Jmschws, Wikimedia Commons/Marc Averette, Zaid Mahomedy (Photographer), Wikimedia Commons/Michael Elleray.

Fig. 2. Questions are posed sequentially to the computer vision system, using a "query language" that is defined in terms of an allowable set of predicates. The interpretation of the questions is unambiguous and does not require any natural language processing. The core of the VTT is an automatic "query generator" that is learned from annotated images and produces a sequence of binary questions for any given "test" image $I_0 \in \mathcal{I}$ whose answers are "unpredictable" (*Statistical Formulation*). In loose terms, this means that hearing the first $k-1$ questions and their true answers for I_0 without actually seeing I_0 provides no information about the likely answer to the next question. To prepare for the test, designers of the vision systems would be provided with the database used to train the query generator as well as the full vocabulary and set of possible questions and would have to provide an interface for answering questions. One simple measure of performance is the average number of correct responses over multiple runs with different test images.

Current Evaluation Practice

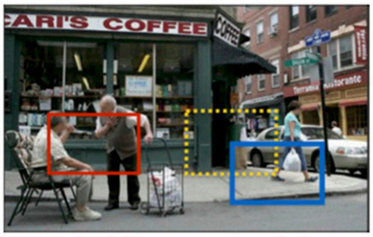
Numerous datasets have been created to benchmark performance, each designed to assess some vision task (e.g., object detection) on some image domain (e.g., street scenes). Systems are evaluated by comparing their output on these data to "ground truth" provided by humans. One well-studied task is classifying an entire image by a general category, either at the object level ("car," "bike," "horse," etc.), where ImageNet (5) is a currently popular annotated dataset, or at the scene level ("beach," "kitchen," "forest," etc.), for example the SUN dataset (23). A natural extension of object-level image categorization is detecting and localizing all instances from generic classes in complex scenes containing multiple instances and events; localization refers to either providing a "bounding box" per instance or segmenting the object from the background. Popular datasets for this task include the Pascal dataset (4), the LabelMe dataset (24), and the Lotus Hill dataset (25), all populated by relatively unconstrained natural images, but varying considerably in size and in the level of annotation, ranging from a few keywords to hierarchical representations (Lotus Hill). Finally, a few other datasets have been assembled and annotated to evaluate the quality of detected object attributes such as color, orientation, and activity; examples are the Core dataset (26), with annotated object parts and attributes, and the Virat dataset (27) for event detection in videos.

Why not continue to measure progress in more or less the same way with common datasets dedicated to subtasks, but using a richer vocabulary? First, as computer vision becomes more ambitious and aims at richer interpretations, it would seem sensible to fold these subtasks into a larger endeavor; a system that detects activities and relationships must necessarily solve basic subtasks anyway. Then why not simply require competing systems to submit much richer annotation for a set of test images than in previous competitions and then rank systems according to consistency with ground truth supplied by human annotators? The reason, and the justification for the VTT, is that the current method does not scale with respect to the richness of the representation. Even for the subtasks in the competitions mentioned earlier, the evaluation of performance, i.e., comparing the output of the system (e.g., estimated bounding boxes) to the ground truth, is not always straightforward and the quality of matches must be assessed (28). Moreover, annotating every image submitted for testing at massive levels of detail is not feasible. Hence, objectively scoring the veracity of annotations is not straightforward. As in school, answering specific questions is usually more objective and efficient in measuring "understanding." Finally, some selection procedure seems unavoidable; indeed, the number of possible binary questions that are both probing and meaningful is virtually infinite. However, selecting a subset of questions (i.e., preparing a test) is not straightforward. We argue that the only way to ask very detailed questions without having their answers be almost certainly "no" is sequential and adaptive querying—questions that build on each other to uncover semantic structure. In summary, the VTT is one way to "scale up" evaluation.

Proposed Test: Overview

Images of Scenes. Our questions are image centered, but images capture 3D scenes. Whereas we pose our questions succinctly in the form "Is there a red car?", this is understood to mean "Is there an instance of a red car in the scene partially visible in the image?". Similarly, given a designated rectangle of image pixels (Fig. 2 shows some examples), the query "Is there a person in the designated region?" is understood to mean "Is there an instance of a person in the scene partially visible in the designated image region?". The universal qualifier "partially visible in the image" (or in the designated region) avoids the issue of the scope of the scene and leads naturally to instantiation and story lines.

Estimating Uncertainty. The justification for counting all questions the same is the property of unpredictability: At each step k, the likelihood that the true answer for question k is "yes" given the true answers to the previous $k-1$ questions is approximately one-half. However, generating long strings of "interesting" questions and "story lines" is not straightforward due to "data fragmentation": A purely empirical solution based entirely on collecting relative frequencies from an annotated training subset of size n from $\mathcal{I}$ is feasible only if the number of questions posed is



1. Q: Is there a person in the blue region? A: yes
2. Q: Is there a unique person in the blue region? A: yes
(Label this person 1)
3. Q: Is person 1 carrying something? A: yes
4. Q: Is person 1 female? A: yes
5. Q: Is person 1 walking on a sidewalk? A: yes
6. Q: Is person 1 interacting with any other object? A: no
⋮
9. Q: Is there a unique vehicle in the yellow region? A: yes
(Label this vehicle 1)
10. Q: Is vehicle 1 light-colored? A: yes
11. Q: Is vehicle 1 moving? A: no
12. Q: Is vehicle 1 parked and a car? A: yes
⋮
14. Q: Does vehicle 1 have exactly one visible tire? A: no
15. Q: Is vehicle 1 interacting with any other object? A: no
17. Q: Is there a unique person in the red region? A: no
18. Q: Is there a unique person that is female in the red region? A: no
19. Q: Is there a person that is standing still in the red region? A: yes
20. Q: Is there a unique person standing still in the red region? A: yes
(Label this person 2)
⋮
23. Q: Is person 2 interacting with any other object? A: yes
24. Q: Is person 1 taller than person 2? A: amb.
25. Q: Is person 1 closer (to the camera) than person 2? A: no
26. Q: Is there a person in the red region? A: yes
27. Q: Is there a unique person in the red region? A: yes
(Label this person 3)
⋮
36. Q: Is there an interaction between person 2 and person 3? A: yes
37. Q: Are person 2 and person 3 talking? A: yes

Fig. 2. A selection of questions extracted from a longer sequence (one of two shown in *SI Appendix*, section 5 and a third, "Sequence 3," available online at Visual Turing). Answers, including identifying Q24 as ambiguous, are provided by the operator (*Human in the Loop*). Localizing questions include, implicitly, the qualifier "partially visible in the designated region" and instantiation (existence and uniqueness) questions implicitly include "not previously instantiated." The localizing windows used for each of the four instantiations (vehicle 1, person 1, person 2, and person 3) are indicated by the colored rectangles (blue, thick border; red, thin border; and yellow, dashed border). The colors are included in the questions for illustration. In the actual test, each question designates a single rectangle through its coordinates, so that "Is there a unique person in the blue region?" would actually read "Is there a unique person in the designated region?". Image courtesy of Boston Globe/Getty Images.

approximately $\log_2 n$. Our proposed solution is presented as part of *Statistical Formulation* and in more detail in *SI Appendix*; it rests on enlarging the number of images in the dataset that satisfy a given history by making carefully chosen invariance and independence assumptions about objects and their attributes and relationships.

Human in the Loop. The operator serves two crucial functions: removing ambiguous questions and providing correct answers. Given a rich family of questions, some will surely be ambiguous for any specific test image. The solution is "just-in-time truthing": Any question posed by the query generator can be rejected by the operator, in which case the generator supplies another nearly unpredictable one, of which there are generally many. The correct answers may or may not be provided to the system under evaluation at run time. Needless to say, given the state of progress in computer vision, neither of these roles can be served by an automated system. The test can be constructed either offline or "online" (during the evaluation). In either case, the VTT is "written" rather than "oral" because the choice of questions does not depend on the responses from the system under evaluation.

Instantiation. A key mechanism for arriving at semantically interesting questions is instance "instantiation." A series of positive answers to inquiries about attributes of an object will often imply a single instance, which can then be labeled as "instance k." Hence, questions that explicitly address uniqueness are also included, which usually become viable, that is close to unpredictable, after one or two attributes have been established. Once this happens, there is no ambiguity in asking whether "person 1" and "person 2" are talking or whether person 1 is occluding "vehicle 2" (Fig. 2). We regard instantiation as identifying the "players" in the scene, allowing for story lines to develop.

Evolving Descriptions. The statistical constraints naturally impose a "coarse-to-fine" flow of information, from gist to semantic detail. Due to the unpredictability criterion, the early questions can only inquire about coarse scene properties, such as "Is there a person in the left-hand side of the image?" or "Is there a person wearing a hat?", because only these have intermediate probabilities of occurrence in the general population. It is only after objects have been instantiated, i.e., specific instances identified, that the likelihoods of specific relationships among these players become appreciably greater than zero.

Vocabulary and Questions

Vocabulary. Our vocabulary $\mathcal{V}$ consists of three components: types of objects, $\mathcal{T}$; type-dependent attributes of objects, $\{\mathcal{A}_t, t\in\mathcal{T}\}$; and type-dependent relationships between two objects, $\{\mathcal{R}_{t,t'}\}$. For example, for urban street scenes, some natural types (or categories) are people, vehicles, buildings, and "parts" such as windows and doors of cars and buildings. Attributes refer to object properties such as clothing and activities of people or types and colors of vehicles. There may also be attributes based on localizing an object instance within an image, and these provide an efficient method of instantiation (below). Relationships between two types can be either "ordered," for instance a person entering a car or building, or "unordered," for instance two people walking or talking together. And some relationship questions may depend on the position of the camera in the underlying 3D scene, such as asking which person or vehicle is closer to the camera. A complete list of objects, attributes, and relationships used in our prototype is included in *SI Appendix*.

Questions. Each question $q\in\mathcal{Q}$ belongs to one of four categories: existence questions, $\mathcal{Q}_{\text{exist}}$; uniqueness questions, $\mathcal{Q}_{\text{uniq}}$; attribute questions, $\mathcal{Q}_{\text{att}}$; or relationship questions, $\mathcal{Q}_{\text{rel}}$. The goal of the existence and uniqueness questions is to instantiate objects, which are then labeled (person 1, vehicle 3, . . .) and subsequently available, by reference to the label, in attribute and relationship questions ("Is person 1 partially occluding vehicle 3?"). Consequently, questions in $\mathcal{Q}_{\text{att}}$ and $\mathcal{Q}_{\text{rel}}$ refer only to previously instantiated objects. Fig. 2 shows examples drawn from $\mathcal{Q}_{\text{exist}}$ (e.g., 1, 19, 26), $\mathcal{Q}_{\text{uniq}}$ (e.g., 2, 9, 17), $\mathcal{Q}_{\text{att}}$ (e.g., 3, 10, 23), and $\mathcal{Q}_{\text{rel}}$ (e.g., 25, 36, 37). Summarizing, the full set of questions is $\mathcal{Q}=\mathcal{Q}_{\text{exist}}\cup\mathcal{Q}_{\text{uniq}}\cup\mathcal{Q}_{\text{att}}\cup\mathcal{Q}_{\text{rel}}$.

As already mentioned, we use "in the designated region" as shorthand for "in the scene that is partially visible in the designated region of the image." Similarly, to avoid repeated discovery of the same objects, all existence and uniqueness questions include the additional qualifier "not previously instantiated," which is always implied rather than explicit. So "Is there a person in the designated region wearing a hat?" actually means "Is there a person in the scene partially visible in the designated region of the image, wearing a hat and not previously instantiated?".

We assume the answers are unambiguous for humans in nearly all cases. However, there is no need to identify all ambiguous questions for any image. Filtering is "as needed": Given $I_0\in\mathcal{I}$, any question q that is elicited by the query generator but is in fact ambiguous for I_0 will be rejected by the human operator during the construction of the VTT. (Examples include question 24 in the partial stream shown in Fig. 2 and three others in the complete streams shown in *SI Appendix*, section 5.)

Statistical Formulation

Selecting questions whose answers are unpredictable is meaningful only in a statistical framework in which answers are random variables relative to an image population $\mathcal{I}$, which serves as the underlying sample space, together with a probability distribution P on $\mathcal{I}$.

Query Generator. Given an image $I \in \mathcal{I}$, the query generator interacts with an oracle (human being) to produce a sequence of questions and correct answers. The human either rejects a question as ambiguous or provides an answer, in which case the answer is assumed to be a (deterministic) function of I. The process is recursive: given a history of binary questions and their answers, $H = ((q_1,x_1),\ldots,(q_k,x_k))$, $q_i \in \mathcal{Q}$, and $x_i \in \{0,1\}$, the query generator either stops, for lack of additional unpredictable questions, or proposes a next question q, which is either rejected as ambiguous or added to the history along with its correct answer x:

$$H \to [H,(q,x)] \triangleq ((q_1,x_1),\ldots(q_k,x_k),(q,x)), \quad x \in \{0,1\}.$$

Not all sequences of questions and answers make sense. In particular, attribute and relationship questions ($\mathcal{Q}_{\mathrm{att}}$ and $\mathcal{Q}_{\mathrm{rel}}$) always refer to previously instantiated objects, restricting the set of meaningful histories, which we denote by $\mathbb{H}$. A key property of histories $H = ((q_1,x_1),\ldots(q_k,x_k)) \in \mathbb{H}$ produced by the query generator is that each question q_i, given the history $((q_1,x_1),\ldots(q_{i-1},x_{i-1}))$, is unpredictable, a concept that we now make precise.

Given a history H, only some of the questions $q \in \mathcal{Q}$ are good candidates for follow-up. As already noted, references to labeled objects cannot precede the corresponding instantiation questions, and furthermore there is a general ordering to the questions designed to promote natural story lines. For a given query generator, we write $\mathcal{Q}_H$ to indicate the set of possible follow-up questions defined by these nonstatistical constraints. Typically, $\mathcal{Q}_H$ contains many candidates, most of which are highly predictable given the history H and therefore unsuitable.

The set of histories, $\mathbb{H}$, can be viewed as a set of binary random variables: $H = H(I) = 1$ if $H = ((q_1,x_1),\ldots(q_k,x_k)) \in \mathbb{H}$ and if the sequence of questions $(q_1,\ldots,q_k)$ produces the sequence of unambiguous answers $(x_1,\ldots,x_k)$ for the image I, and $H = 0$ otherwise. We write P_H for the conditional probability on $\mathcal{I}$ given that $H(I) = 1$.

Consider now the probability under P_H that a question $q \in \mathcal{Q}_H$ elicits the (unambiguous) response $X_q = X_q(I) \in \{0,1\}$, for a given history $H \in \mathbb{H}$:

$$P_H(X_q = x) \triangleq \frac{P\{I : H(I) = 1, X_q(I) = x\}}{P\{I : H(I) = 1\}}. \qquad [1]$$

For simplicity, we have represented the set $\{I : [H,(q,x)](I) = 1\}$ in the numerator of [1] with the more intuitive expression $\{I : H(I) = 1, X_q(I) = x\}$, although this is decidedly an abuse of notation because the function $X_q(I)$ is not defined in the absence of the history H. Still, under P_H, X_q is a binary random variable that may or may not be unpredictable. To make this precise, we define the predictability of $q \in \mathcal{Q}_H$, given the history $H \in \mathbb{H}$, by $\rho_H(q) = |P_H(X_q = 1) - 0.5|$. Evidently, $\rho = 0$ indicates q is totally unpredictable and $\rho = 0.5$ indicates q is totally predictable.

Randomization. In general, many questions have answers with low predictability at each step k. Rather than select the most unpredictable question at step k, we make a random selection from the set of almost unpredictable questions, defined as those for which $\rho_H(q) \le \epsilon$, where H is the history preceding the kth question. (In practice we choose $\epsilon = 0.15$, and we designate all such questions unpredictable.) In this way, we can generate many query streams for a given test image I and develop multiple story lines within a query stream. In doing so, a path to instantiation might be $\{X_{ta} = 1, X_{tb} = 1, X_{ut\{a,b\}} = 1\}$, meaning that once there are instances of object type t with attribute a and also instances with attribute b, then the likelihood of having a unique ("u") instance with both attributes may rise to approximately one-half. Commonly, a designated region serves as an important instantiating attribute, as in the chain $\{X_{ta} = 1, X_{uta} = 0, X_{t\{a,b\}} = 1, X_{ut\{a,b\}} = 1\}$, where a is the designated region. Here, for example, t might refer to a person, of which several are partially visible in region a, but only one possesses the additional attribute b (e.g., "sitting," "female," or "wearing a hat"). There are more examples in Fig. 2, two complete sequences of questions in *SI Appendix*, section 5, and an additional sequence, "Sequence 3," available online at Visual Turing.

Story Lines and the Simplicity Preference. We impose constraints on the set of questions allowed at each step—the set of available follow-up questions given the history H, which we have denoted by $\mathcal{Q}_H$, is a small subset of the set of all possible questions, $\mathcal{Q}$. The main purpose is to encourage natural sequences, but these constraints also serve to limit the number of conditional likelihoods that must be estimated.

The loop structure of the query engine enforces a general question flow that begins with existence and uniqueness questions ($\mathcal{Q}_{\mathrm{exist}}$, $\mathcal{Q}_{\mathrm{uniq}}$), with the goal of instantiating objects. As objects are instantiated, the vision system is interrogated about their properties, meaning their attributes, and then their relationships to the already instantiated objects. After these story lines are exhausted, the outer loops are revisited in search of new instantiations. The query engine halts when there are no more unpredictable existence or uniqueness questions. As already mentioned, all loops include randomization, meaning that the next query is randomly selected from the questions in $\mathcal{Q}_H$ that are found to be unpredictable.

The pose attribute is especially useful to an efficient search for uniquely characterized objects, i.e., instantiation. Once the existence of an object that is partially visible within a region w is established, ensuing existence and uniqueness queries are restricted to w or its subregions. As these regions are explored, the unpredictability constraint then favors questions about the same object type, but annotated with additional attributes. Eventually, either an object partially visible in a subregion of w is instantiated or the collection of unpredictable questions about such an object is exhausted. In the latter case the query engine returns to the outer loop and begins a new line of questions; in the former, it explores the attributes and relationships of the newly instantiated object. (All regions are rectangular and the full set, $\mathcal{W}$, is specified in *SI Appendix*.)

Finally, there is a simplicity constraint that further promotes a natural line of questions. This can be summarized, roughly, as "one new thing at a time." An existence, uniqueness, or attribute question, q, is considered simpler than an alternative question of the same type, q', if q contains fewer attributes than q'. Given the unpredictable subset of $\mathcal{Q}_H$, simpler questions are favored over more complex questions, and questions of equal complexity are chosen from with equal likelihood. Further detail and pseudocode—see Algorithm—can be found in *SI Appendix*.

Estimating Predictability. The conditional likelihoods, $P_H(X_q = 1)$, are estimated from a training set $\mathbb{T}$ in which all answers (or equivalent information—Fig. 3) are provided for each of n images from $\mathcal{I}$. The methods used to gather and annotate the training images are discussed in the next section, on the prototype VTT. The objects, people and vehicles, are located with bounding boxes and labeled with their attributes, and pairs of objects are labeled with their relationships.

The task of estimating conditional likelihoods, and therefore predictability, is guided in part by the ordering of questions built into the query engine, which, as already noted, begins with a search for an instantiated object, immediately followed by questions to determine its attributes, and then finally by an exploration of its relationships with any previously instantiated objects.

For instantiation questions, $q \in \mathcal{Q}_{\text{inst}} \triangleq \mathcal{Q}_{\text{exist}} \cup \mathcal{Q}_{\text{uniq}}$, the natural estimator $\hat{P}_H(X_q = 1)$ is the relative frequency (maximum likelihood) estimator

$$\frac{\#\{I \in \mathbb{T} : H(I) = 1, X_q(I) = x\}}{\#\{I \in \mathbb{T} : H(I) = 1\}}. \tag{2}$$

Observe, though, that the number of images in the training set that satisfy the history H [i.e., for which $H(I) = 1$] is cut approximately in half at each step, and hence after about $\log_2 n$ steps direct estimation is no longer possible. Consequently, to generate tests with more than 10 or so questions, we are obliged to make "invariance" assumptions to allow for data pooling to expand the number of images from which these relative frequencies are computed. Specifically, if we assume that X_q, $q \in \mathcal{Q}_{\text{inst}}$, given the history $H \in \mathbb{H}$, depends only on a subsequence, H'_q of H, then the distribution on X_q is invariant to the questions and answers in H that were dropped, and the estimator [**2**] can be modified by substituting the condition $H(I) = 1$ by $H'_q(I) = 1$.

Let $w \in \mathcal{W}$ be the localizing region, possibly the entire image, referenced in the instantiation question q. H'_q is derived from H by assuming that the event $X_q = x$ is independent of all attribute and relationship questions in H and all existence and uniqueness questions that involve localizations $w' \in \mathcal{W}$ that are disjoint from w, with the important exception of uniqueness questions that answered positive ($q' \in \mathcal{Q}_{\text{uniq}}$, $X_{q'} = 1$) and therefore instantiated a new object. In other words, the approximation is that, conditioned on the history, the distribution of an instantiation question depends only on the uniqueness questions that instantiated objects and the existence and uniqueness questions that are localized to regions intersecting w. By preserving the instantiating questions in H, which addresses the potential complications introduced by the implied qualifier "not previously instantiated," we guarantee that $H(I) = 1 \Rightarrow H'_q(I) = 1$ for all $I \in \mathbb{T}$, so that the population of images used to estimate $P_H(X_q = 1)$ with $H'_q(I)$ is no smaller than the one with $H(I)$ and typically far larger. More discussion and a further invariance assumption leading to further improvement in population size are included in *SI Appendix*.

As for attribute questions, $q \in \mathcal{Q}_{\text{att}}$, which are always about the most recently instantiated object and always precede any relational information, the natural (relative frequency) estimator for $P_H(X_q = 1)$ is in terms of the population of labeled objects found in the training images, rather than the images themselves. Given a history H, consider a question of the form $q = o_t a$: "Does object o_t have attribute a?" where o_t is an object of type $t \in \{\text{person, vehicle}\}$ and $a \in \mathcal{A}_t$. The history, H, defines a (possibly empty) set of attributes, denoted A, that are already known to belong to o_t. Let $\mathcal{O}_{\mathbb{T}}$ be the set of all annotated objects in the training set, and, for each $o \in \mathcal{O}_{\mathbb{T}}$, let $\mathcal{T}_{\mathbb{T}}(o)$ be the type of o and $\mathcal{A}_{\mathbb{T}}(o)$ be the set of attributes belonging to o; e.g., $\mathcal{T}_{\mathbb{T}}(o) = \{\text{person}\}$ and $\mathcal{A}_{\mathbb{T}}(o) = \{\text{female, adult, standing}\}$ for the rightmost object in Fig. 3. The relative frequency estimator for $P_H(X_q = 1)$, using the population of annotated objects, is

$$\frac{\#\{o \in \mathcal{O}_{\mathbb{T}} : \mathcal{T}_{\mathbb{T}}(o) = t, A \cup \{a\} \subseteq \mathcal{A}_{\mathbb{T}}(o)\}}{\#\{o \in \mathcal{O}_{\mathbb{T}} : \mathcal{T}_{\mathbb{T}}(o) = t, A \subseteq \mathcal{A}_{\mathbb{T}}(o)\}}. \tag{3}$$

There is again the sparsity problem, which we address in the same way—through invariance assumptions that effectively increase the number of objects. The set of attributes for objects of type t can be partitioned into subsets that can be reasonably approximated as independent conditioned on belonging to a particular object o_t. As an example, if $t =$ person, then crossing a street is not independent of standing still, but both are approximately independent of sex, {male, female}, and of child vs. adult, as well as whether o_t is carrying something or wearing a hat. These conditional independence assumptions decrease the size of the set A in [**3**], thereby increasing the set of $o \in \mathcal{O}_{\mathbb{T}}$ used to estimate $P_H(X_q = 1)$.

The approach to relationship questions, $q \in \mathcal{Q}_{\text{rel}}$, is essentially the same as the approach to attribute questions, except that the training population is the set of pairs of objects in the training images, rather than the individual objects. The independence (invariance) assumptions include relationships that are independent of the attributes of the related objects (e.g., the relationship driving/riding a vehicle is assumed to be independent of the sex of the person driving or riding, as well as whether the vehicle is dark or light colored, or whether its tires are visible) and relationships that are independent of each other (e.g., whether one vehicle is closer to the camera than another vehicle is assumed to be independent of which vehicle is larger). A systematic accounting of the independence assumptions used in our prototype VTT, for both attribute and relationship questions, can be found in *SI Appendix* and its accompanying tables.

A Prototype VTT

The data collection and annotation were performed by undergraduate workers at Johns Hopkins University. Unlike "crowd sourcing," this allowed for more customized instructions. Our dataset has 2,591 images, collected online using search engines such as Google street view and required to meet certain basic criteria: Portray a standard city street scene; be obtained during daytime; have a camera height from roughly head level to several feet above; and contain clearly visible objects, attributes, and relationships from our vocabulary. The images are from large cities from many countries.

For annotation, we can rule out directly answering each binary question $q \in \mathcal{Q}$, because the questions make sense only in the context of a history—$\mathcal{Q}_{\text{att}}$ and $\mathcal{Q}_{\text{rel}}$ always refer to instantiated objects, and $\mathcal{Q}_{\text{exist}}$ and $\mathcal{Q}_{\text{uniq}}$ always include the not previously instantiated qualification. As discussed, a history itself can be viewed as a binary function of the image, but there are far too many for an exhaustive annotation. Instead, an essentially equivalent, but more compact and less redundant, representation was used. Bounding boxes were drawn around every instance of an object for which the annotator had no uncertainty about its category (example in Fig. 3). For partially occluded objects, the bounding box was placed over the region of the image that the annotator expected the object would occupy had the object not been partially occluded. Attributes were annotated only for objects in which all of the attributes were unambiguous, which

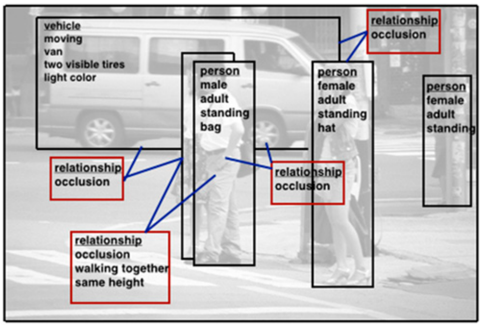


Fig. 3. Annotation provided by human workers.

alleviated the annotation of distant objects. Relationships were annotated only between pairs of objects with bounding boxes and for which at least one relationship from the type-dependent list was present. The complete vocabulary is given in the first two tables of *SI Appendix*.

Level of Difficulty. The vocabulary was selected to avoid query streams that would be considered hopelessly difficult by today's computer-vision standards. Nevertheless, there are plenty of subtleties to challenge, and likely defeat, the best existing systems, e.g., the second stream in *SI Appendix*, section 5.2, which includes an example of extreme occlusion; two examples that require inferring that bicycles are moving, rather than stopped; and another occlusion that rests on the interpretation of a small number of pixels. A few additions to the vocabulary would dial up the difficulty, considerably, say adding the relationship "playing catch" or other objects like windows, signs, and tables and chairs, which are often nearly impossible to identify without context, especially when partially occluded.

Discussion

In the decades following the research of Alan Turing, computer vision became one of the most active areas of AI. The challenge of making computers "see" has attracted researchers from across science and engineering and resulted in a highly diverse set of proposals for formulating the "vision problem" in mathematical terms, each with its ardent advocates. The varying popularity of competing strategies can be traced in the proceedings of conferences.

Debates persist about what actually works and how to measure success. Until fairly recently, each new method was "validated" on homegrown data and with homegrown metrics. Recently, the computer vision community has accepted testing on large common datasets, as reviewed above, and various well-organized "challenges" have been accepted by many research groups. Many believe that adopting uniform metrics has made it easier to sort out what works appreciably better than before and accelerated progress.

However, these metrics, such as false positive and false negative rates for subtasks such as detecting and localizing people, do not yet apply to the richer descriptions that human beings can provide, for example in applying contextual reasoning to decide whether a car is "parked" or is "larger" than another, or a person is "leaving" a building or "observing" something, or two people are "walking and talking together." If annotating ordinary scenes with such precision is accepted as a benchmark for vision, then we have argued for raising the bar and proceeding directly to metrics for full-scale scene interpretation. We have proposed a "written" VTT as a step in this direction.

Many design decisions were made, some more compelling than others. Story lines approximate natural sequences of questions and are well handled by the loop structure of the algorithm. On the other hand, whereas conditional independence assumptions are probably a necessary approach to the data sparsity problem, the prototype lacks a unified implementation. Scaling to substantially larger vocabularies and more complex relationships, and deeper part/whole hierarchies, would be difficult to manage by simply enlarging the existing brute-force tabulation of dependency relationships (*SI Appendix*). Possibly, the right approach is to build full-blown generative scene models, at least for the placements of parts and objects, and object groupings, from which predictability could be estimated via sampling or inferred by direct calculation.

Finally, coming back to a "conversation" with a machine, another possibility is a more free-form, open-ended "oral test": The operator formulates and delivers a query to the system under evaluation, awaits an answer, and then chooses the next query, presumably based on the history of queries and system answers. As before, the operator may or may not provide the correct answer. This has the advantage that the operator can "probe" the system capacities with the singular efficiency of a human, for example detect and focus on liabilities and ask "confirmatory" questions. However, the oral test has the disadvantage of being subjective and requiring rapid, basically real-time, responses from the system. On balance, the written test seems to be more practical, at least for the time being.

ACKNOWLEDGMENTS. We received helpful input from Mark Johnson of Macquarie University and Vincent Velten of the Air Force Research Laboratory. This work was partially supported by the Office of Naval Research under Contract ONR N000141010933, the Defense Advanced Research Projects Agency under Contract FA8650-11-1-7151, and the National Science Foundation under Grant 0964416.

1. Turing AM (1950) Computing machinery and intelligence. *Mind* 59:433–460.
2. Saygin AP, Cicekli I, Akman V (2003) Turing Test: 50 Years Later. *The Turing Test*, ed Moor JH (Springer, Heidelberg, Germany), pp 23–78.
3. Russell SJ, Norvig P (2003) *Artificial Intelligence: A Modern Approach* (Pearson Education, Harlow, UK).
4. Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A (2010) The pascal visual object classes (voc) challenge. *Int J Comput Vis* 88:303–338.
5. Deng J, et al. (2009) Imagenet: A large-scale hierarchical image database. *Proceedings IEEE 2009 CVPR* (IEEE, New York), pp 248–255.
6. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* 60(2):91–110.
7. Zhu Q, Yeh MC, Cheng KT, Avidan S (2006) Fast human detection using a cascade of histograms of oriented gradients. *Proceedings IEEE 2006 CVPR* (IEEE, New York), Vol 2, pp 1491–1498.
8. Yu G, Morel JM (2009) A fully affine invariant image comparison method. *Proceedings IEEE 2009 International Conference on Acoustics, Speech and Signal Processing* (IEEE, New York), pp 1597–1600.
9. Zhu SC, Mumford D (2006) A stochastic grammar of images. *Foundations and Trends in Computer Graphics and Vision* 2(4):259–362.
10. Ommer B, Sauter M, Buhmann JM (2006) Learning top-down grouping of compositional hierarchies for recognition. *Proceedings IEEE 2006 CVPR* (IEEE, New York), pp 194–201.
11. Chang LB, Jin Y, Zhang W, Borenstein E, Geman S (2011) Context, computation, and optimal roc performance in hierarchical models. *Int J Comput Vis* 93(2):117–140.
12. Lu W, Lian X, Yuille A (2014) Parsing semantic parts of cars using graphical models and segment appearance consistency. arXiv:1406.2375.
13. Hinton GE, Osindero S, Teh YW (2006) A fast learning algorithm for deep belief nets. *Neural Comput* 18(7):1527–1554.
14. Fei-Fei L, Fergus R, Perona P (2003) A Bayesian approach to unsupervised one-shot learning of object categories. *Proceedings IEEE 2003 ICCV* (IEEE, New York), pp 1134–1141.
15. Amit Y, Trouvé A (2007) Pop: Patchwork of parts models for object recognition. *Int J Comput Vis* 75:267–282.
16. Felzenszwalb PF, Huttenlocher DP (2005) Pictorial structures for object recognition. *Int J Comput Vis* 61(1):55–79.
17. Felzenszwalb PF, Girshick RB, McAllester D, Ramanan D (2010) Object detection with discriminatively trained part-based models. *IEEE Trans Patt Anal Machine Intell* 32:1627–1645.
18. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. *NIPS* (Neural Information Processing Systems Foundation, La Jolla, CA), pp 1097–1105.
19. Girshick R, Donahue J, Darrell T, Malik J (2013) Rich feature hierarchies for accurate object detection and semantic segmentation. arXiv:1311.2524.
20. Oquab M, Bottou L, Laptev I, Sivic J, et al. (2014) Learning and transferring mid-level image representations using convolutional neural networks. *Proceedings IEEE 2014 CVPR* (IEEE, New York), pp 1717–1724.
21. Zhang N, Paluri M, Ranzato M, Darrell T, Bourdev L (2013) Panda: Pose aligned networks for deep attribute modeling. arXiv:1311.5591.
22. Hariharan B, Arbeláez P, Girshick R, Malik J (2014) Simultaneous Detection and Segmentation. *ECCV 2014* (Springer, Heidelberg, Germany), pp 297–312.
23. Xiao J, Hays J, Ehinger KA, Oliva A, Torralba A (2010) Sun database: Largescale scene recognition from abbey to zoo. *Proceedings IEEE 2010 CVPR* (IEEE, New York), pp 3485–3492.
24. Russell BC, Torralba A, Murphy KP, Freeman WT (2008) Labelme: A database and web-based tool for image annotation. *Int J Comput Vis* 77(1-3):157–173.
25. Yao B, Yang X, Zhu SC (2007) *Introduction to a Large-Scale General Purpose Ground Truth Database: Methodology, Annotation Tool and Benchmarks* (Springer), pp 169–183.
26. Endres I, Farhadi A, Hoiem D, Forsyth DA (2010) The benefits and challenges of collecting richer object annotations. *Proceedings IEEE 2010 CVPR* (IEEE, New York), pp 1–8.
27. Oh S, et al. (2011) A large-scale benchmark dataset for event recognition in surveillance video. *Proceedings IEEE 2011 CVPR* (IEEE, New York), pp 3153–3160.
28. Özdemir B, Aksoy S, Eckert S, Pesaresi M, Ehrlich D (2010) Performance measures for object detection evaluation. *Pattern Recognit Lett* 31:1128–1137.