Leveraging Human Computations to Improve Schematization of Spatial Relations from Imagery

HUAMING RAO, Nanjing University of Science and Technology,

University of Illinois at Urbana-Champaign

SHIH-WEN HUANG, University of Washington

WAI-TAT FU, University of Illinois at Urbana-Champaign

The process of generating schematic maps of salient objects from a set of pictures of an indoor environment is challenging. It has been an active area of research as it is crucial to a wide range of context- and location-aware services, as well as for general scene understanding. Although many automated systems have been developed to solve the problem, most of them either require predefining labels or expensive equipment, such as RGBD sensors or lasers, to scan the environment. In this article, we introduce a prototype system to show how human computations can be utilized to generate schematic maps from a set of pictures, without making strong assumptions or demanding extra devices. The system requires humans (crowd workers from Amazon Mechanical Turks) to do simple spatial mapping tasks in various conditions, and their data are aggregated by filtering and clustering techniques that allow salient cues to be identified in the pictures and their spatial relations to be inferred and projected on a two-dimensional map. In particular, we tested and demonstrated the effectiveness of two methods that improved the quality of the generated schematic map: (1) We encouraged humans to adopt an allocentric representations of salient objects by guiding them to perform mental rotations of these objects and (2) we sensitized human perception by guided arrows superimposed on the imagery to improve the accuracy of depth and width estimation. We demonstrated the feasibility of our system by evaluating the results of schematic maps generated from indoor pictures taken from an office building. By calculating Riemannian shape distances between the generated maps to the ground truth, we found that the generated schematic maps captured the spatial relations well. Our results showed that the combination of human computations and machine clustering could lead to more-accurate schematized maps from imagery. We also discuss how our approach may have important insights on methods that leverage human computations in other areas.

Categories and Subject Descriptors: F.2.2 [Nonnumerical Algorithms and Problems]: Geometrical Problems and Computations; H.5.3 [Group & Organization Interfaces]: Collaborative Computing

General Terms: Design, Experimentation, Performance

Additional Key Words and Phrases: Human computations, crowdsourcing, picture schematization, spatial cognition

ACM Reference Format:

Huaming Rao, Shih-Wen Huang, and Wai-Tat Fu. 2016. Leveraging human computations to improve schematization of spatial relations from imagery. *ACM Trans. Intell. Syst. Technol.* 7, 4, Article 54 (March 2016), 21 pages.

DOI: <http://dx.doi.org/10.1145/2873065>

The project was partially supported by Office of Naval Research Grant No. N00014-12-1-0486 and by National Science Foundation NSF Grant No. 14-19297.

Authors' addresses: H. Rao, Department of Computer Science, Nanjing University of Science and Technology, Nanjing, Jiangsu, China P. R.; email: huaming.rao@gmail.com; S.-W. Huang, Department of Computer Science, University of Washington, Seattle, WA, US; email: shwenhuang98@gmail.com; W.-T. Fu, Department of Computer Science, University of Illinois, Champaign, IL, US; email: wfu@illinois.edu.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.

© 2016 ACM 2157-6904/2016/03-ART54 \$15.00

DOI: <http://dx.doi.org/10.1145/2873065>

1. INTRODUCTION

Building machines that can *see and understand* the environment like humans has always been a notoriously difficult problem. However, given its critical role in an ever-increasing range of applications—from human-robot interactions to autonomous driving to image search engines—there has been a growing literature on various methods that tackle specific aspects of scene understanding to improve performance of such intelligent systems. The goal of this article is to contribute to this growing literature by focusing on the potential of human computations in scene understanding. In particular, we will focus on a small but important task in the context of scene understanding: schematization of spatial relations of salient objects from imagery.

The task of schematizing spatial relations of salient objects is challenging because it involves not only object labeling but also the generation of a schematic map (i.e., a two-dimensional abstract representation) that captures the relative distances and spatial locations of salient objects in a scene. Because this scene structure maintains the essential spatial relations while abstracting away visual details, it is often regarded as an important step for a wide range of applications. It is also considered an important step for demonstrating human-level intelligence, as identification of salient objects in different scenes is considered a critical step in the general problem of scene understanding. It involves questions with regard to what and why certain objects are considered salient in a scene, what the spatial and semantic relations of these objects are, and how these relations give *meanings* to the scene that can be useful for different applications.

Besides its critical role in the general context of scene understanding, schematization has become an active area of research on its own in a wide range of disciplines. For example, schematizing indoor pictures are useful for indoor navigation and robotic guidance, as they help to provide overall spatial structures of the environment that is essential for spatial planning. In human-computer interaction research, schematic maps of scenes can be shared during remote collaboration to allow two persons to focus on essential objects during communication. Schematic maps are also found to be useful for location-based or content-based image retrieval, as spatial structures of images can be extracted and used as cues for retrieval in addition to keywords associated with the images. Last, picture schematization can also be used for positioning and object manipulation in augmented reality applications.

Despite years of research, the process of generating schematic maps from images remains a challenging problem. The process involves at least the following important components: (1) detection of salient objects in the three-dimensional (3D) scene of the picture, (2) distance estimation between the salient objects, (3) coordinate alignment between the 3D image and the 2D map, and (4) spatial inferences based on integrating and mapping of objects in both the 3D image and the 2D map. The first component of the task is an object detection task that is typical of many computer vision systems. The second component is basically a problem of image depth extraction, which is typically solved by either stereo vision or exploiting parallax information [Labayrade et al. 2002; Marr and Poggio 1979]. The third and fourth components are essentially a 3D scene reconstruction task. Each single of the above components is a quite complex problem by itself in the field of computer vision, let alone the combination of them. In fact, most of the automation methods are still far from the level of performance by humans.

At the same time, human computations (or crowdsourcing), as an emerging technique, has demonstrated its power to solve problems that are difficult to automate but relatively easy to be handled by humans [Bernstein et al. 2010; Bigham et al. 2010; Quinn and Bederson 2011; Snow et al. 2008; Xu et al. 2014]. Especially, it has been actively applied in image labeling tasks [Deng et al. 2013; Hara et al. 2013; Sorokin and Forsyth 2008; Von Ahn and Dabbish 2004], so it is reasonable to utilize human computations to perform the step of object identification in picture schematization.

Research on human perception shows that humans are in general good at detecting depth information in pictures by various perceptual cues such as textures, alignment, scales, and so on [Patterson et al. 1992], so it is therefore reasonable to assume that, without making strong assumptions or demanding for extra devices, humans are capable of detecting salient objects as well as their spatial relations from a picture of a scene. So the main question is How and to what extent can one leverage untrained humans (such as crowd workers from Amazon Mechanical Turks (AMT)) to help generate schematic maps from pictures of a scene and are these outcomes reliable? Another important question is in regard to how different maps generated by humans can be merged effectively in ways such that the human data can be differentially weighted based on their quality and be integrated in ways that increase the robustness of the final schematic maps.

In this article, we present a prototype of a human computation system that is capable of generating 2D schematic maps of objects in sets of pictures of a scene. Though the generated maps are different parts of the whole scene, it is the first step towards building a complete map by combining different pictures in a large scene. We performed a series of experiments to test the methods we used to generate crowd data. The results provided direct support to the proposed crowd-powered system. To preview our results, the current article has the following contributions:

- We introduce a prototype system to show how human computations can be utilized to generate a schematic map from a set of pictures of an indoor environment, without making strong assumptions or demanding for extra devices.
- We demonstrated that spatial rotations of relative locations of objects could improve the quality of human data.
- We demonstrated that emphasized dimensions indicated by the arrows on the pictures could sensitize depth and width perception of humans, thereby further improving quality of human data.
- We demonstrated the feasibility of a human computation system that generates 2D maps from multiple pictures of a scene, and the results match well with the actual maps.

2. RELATED WORK

As indoor location-aware mobile devices are used extensively around the world, they create so many business opportunities and it is becoming more obvious that indoor maps will open a new frontier of the indoor location-based services. In the fields of cartography, geodesy, and GIScience, a lot work has been done to advance standards and industry cooperation aimed at convergence of indoor positioning technologies, such as GeoSPARQL standard, IndoorGML, and Industry Foundation Classes (IFC) Building Information Model (BIM) standard [Battle and Kolas 2012; Bazjanac 2008; Li and Lee 2013]. In the meantime, researchers in the field of computer science are also following the trend to make contributions to this domain with the techniques of computer vision, augmented reality, human computations and crowdsourcing, and so on. Before we present our proposed system and the experiments, we will first review relevant literature that motivates the main components of the proposed system.

2.1. Computer Vision Techniques

The general procedure for automatically reconstructing a 3D scene typically consists of two steps: object detection and depth estimation. Visual features are extracted using various techniques in the area of computer vision to detect salient objects from a picture of a scene [Bay et al. 2006; Espinace et al. 2010; Swadzba and Wachsmuth 2011]. Typical techniques to extract distance information (e.g., depth information)

of an object from a picture of a scene include the use of stereo vision or parallax information [Labayrade et al. 2002; Marr and Poggio 1979]. Most of these traditional techniques, however, ignore the task of semantic understanding (e.g., what kinds of objects are likely in a scene) and estimate depth information or geometry of objects from machine-readable image features; and in many cases, they still need to make strong assumptions about the structures of indoor environments [Liu et al. 2010]. Some other techniques generate indoor floor maps with the help of additional external inputs, such as a CAD file [Schafer et al. 2011] and radio signals, like laser range data [Turner and Zakhor 2014], or of additional expensive equipment, such as the Google Tangle smartphone, which is embedded with a PrimeSense chip powered by Apple Technology.¹ To summarize, most of the existing computer vision techniques for generating schematic maps require predefining labels on the picture or only work for known objects, and some other techniques require expensive equipment, such as RGBD sensors or lasers, to collect depth images.

2.2. Human Computation Systems

Crowdsourcing and human computations have been applied to many different domains in computer science because of their ability to utilize human intelligence to solve problems that are difficult or unsolvable by computers alone [Bernstein et al. 2010; Bigham et al. 2010; Quinn and Bederson 2011; Snow et al. 2008; Xu et al. 2014]. Take the field of computer vision, for example: On one hand, humans are crucial for data collection, annotation, evaluation, and knowledge extraction, and, on the other hand, computer vision systems and humans can collaborate to break new ground and accomplish challenging tasks. Image labeling is such a task where human computations have been used extensively because humans have much better ability than computers in object recognition. Von Ahn and Dabbish [2004] built the ESP game, a computer game that allows the players to generate image labels while they are playing the game. Sorokin and Forsyth [2008] showed that one can get cheap and high-quality labeled data by hiring workers from Amazon Mechanical Turk. Deng et al. [2013] showed how humans can help computers select discriminative features to accomplish fine-grained recognition. Hara et al. [2013] found that crowd workers can identify 93% accessibility problems in pictures from Google street view. Rao et al. [2013] showed that crowd workers can locate a camera view in a two-dimensional map. Our study complements these previous studies by examining whether and how we can apply human computations to generate 2D schematic maps of object locations from sets of pictures of a scene in an indoor environment.

Typically, crowdsensing is the most commonly used technique for inferring semantics of indoor environments. *CheckInside*, presented by Elhamshary and Youssef [2014], leverages data mined opportunistically from the users' phone sensors and data about venues from traditional location-based social networks to fingerprint each venue. Chon et al. [2012] present *CrowdSense@Place* as a framework that exploits opportunistically captured images and audio clips from smartphones to link place visits with place categories. They also propose an autonomous place naming system [Chon et al. 2013] by integrating sensing system and exploiting crowdsourcing to gather data to provide place names from a person's perspective, beyond raw location coordinates. Alzantot and Youssef [2012] present *CrowdInside* for automatic construction of indoor floor plans based on the accurate motion traces collected from users' mobile devices, which are then processed to detect the overall floor plan shape as well as higher-level semantics. Instead of using crowd-sensed data, which usually requires large volumes collected

¹<http://www.google.com/atap/project-tango/>.

from users, our proposed system is aimed at generating 2D schematic maps from a set of indoor pictures by directly instructing a few crowd workers to do some simple spatial mapping tasks in various conditions and using filtering and clustering techniques to identify salient cues in pictures and inferred their locations on a 2D map.

One of the biggest challenges in crowdsourcing and human computations is quality control, because it usually knows little about who is doing the task, and it could not guarantee whether the worker is well trained with specified skills, which make it hard to verify the answers of the workers [Ipeirotis et al. 2010]. The simplest and most common way for people to control the quality is to use majority vote [Quinn and Bederson 2011], which collects multiple labels from crowd workers and selects the label with the highest number of votes from the crowd workers. Ipeirotis et al. [2010] further utilized an expectation-maximization-style approach to estimate the quality of the work and workers. Dai et al. [2010] demonstrated that one can use a decision-theoretic model to find the optimal actions of a human computation system. In addition, researchers also showed that task design highly affects the quality of the workers' work. Oleson et al. [2011] found that a human computation system can insert a gold standard question (i.e., question with a known correct answer) to filter out low-quality data. Huang and Fu [2013] showed that making the workers collaborate or compete with each other can better motivate them to generate high-quality work. In this study, we also identify the special challenges to improve the quality of collected data for object location schematization task and propose novel solutions to the challenges.

2.3. Incorporating Human Perception into the System

The main thrust of the system is to leverage the capability of human perception and cognition to improve spatial schematization from imagery. To this end, we review the literature in cognitive science, which generally lists four major cognitive operations that are involved as humans process spatial information in the environment: triangulation, mental rotation, image comparison, and translation [Aretz 1990]. In particular, studies have shown that mental rotation is a cognitively demanding task. In fact, experiments show that when a mental rotation is performed, there is a linear relationship between response time and the angular disparity between the map and the forward view [Shepard and Cooper 1986; Cooper 1976]. In addition to the general linear relation between angles or rotation and reaction times, errors also tend to increase with the angle of rotation. However, unique patterns of errors rates are often observed, in which multiple peaks tend to occur as the angle of rotation approaches the oblique directions (i.e., 45° , 135° , 225° , 315°) [Shepard and Metzler 1971]. As we will discuss later, these features motivate us to design studies that test how mental rotation may help improve the overall quality of human data in picture schematization.

Studies on human perception show that depth perception involves monocular and binocular depth cues. Parallel lines, for example, sometimes help people focus on a dimension to help them better utilize depth cues [Julesz 1960]. Patterson et al. also conducted some interesting experiments to test the factors that affect depth perception in stereoscopic displays [Patterson et al. 1992] and found that viewing distance, stimulus size, and exposure duration all have an impact on human depth perception. Purves and Lotto [2011] developed a wholly empirical theory of vision and provided evidence to indicate that monocular and binocular depth perception can be understood in terms of empirical associations that link images whose meanings are inherently undetermined to their behavioral significance. In general, although humans are good at depth perception, there are many factors that make the task difficult. Depth cues that help people focus on certain dimensions have proved to be helpful for better depth perception [Patterson et al. 1992].

Fig. 1. Spatial mapping task: (1) It requires one to label objects in the image (2) and map their spatial relations in a 2D representation.

3. SYSTEM DESIGN

There are four types of human computation or “crowd-powered” systems [Geiger et al. 2011]: information processing systems, rating systems, problem-solving systems, and creative systems. Our system falls into the categories of rating and problem-solving systems, because it has the following characteristics: (1) The system does not know the correct answers a priori, (2) it relies on large quantities of contributions from humans, (3) contributions are not evaluated individually but rather in a collective way, and (4) contributions are aggregated to a collective response to the given task, usually by statistical approaches. Following the general design guidelines of existing crowd rating systems, we will describe how the current system is designed to address these characteristics.

3.1. Spatial Mapping Task

We created a spatial mapping task that required crowd workers to draw and label salient objects from an image and map their relations on a 2D grid, as shown in Figure 1. The interface of the task (see Figure 2) was designed with the imagery on the left and the 2D projection on the right. In the spatial mapping task, participants are first shown a picture taken from an environment (see **B** in Figure 2) and are instructed to identify at least three objects that are most salient, such as “landmarks” that help others to navigate in the environment. Then they are instructed to draw rectangles on the left image to indicate the location of the salient objects and are asked to label each object with a single or two words (e.g., “sofa” and “restroom”). After drawing the rectangles, the corresponding labels would appear at the top-left corner of the 2D grid in **C**. The workers then were required to drag and drop those labels to the locations on the 2D grid such that they matched as close as possible to the locations that they perceived in the image. They could edit the label or delete it in **D** at any time during this process.

The interface provided two linked sets of data for a given image. The first set was the set of rectangles indicating salient objects in the image, together with the corresponding labels of these objects. The second set was the projected locations of the labels on the 2D grid, which captured the spatial configuration of the identified salient objects in the image. As we will explain below, these two linked data sets provided valuable information that allowed us to derive the schematic map of the environment in the image.

3.2. Challenges and Solutions

Our system focuses on solutions that tackle two major challenges:

Challenge 1. In picture schematization, one needs to focus on the overall level of coherence of the relative locations of objects in an allocentric frame of reference (i.e., a birds’ eye view). However, an image of an environment often shows objects from a first-person egocentric point of view (e.g., from the view of the camera). So the first

Fig. 2. System interface: (A) instruction; (B) an indoor image where participants are asked to draw rectangles and label the objects. The figure indicates the standing point, and the arrows indicate the orientation. (C) A 2D grid where participants are asked to move the labels (corresponding to rectangle) to the locations relative to where the figure indicates. The figure indicates the standing point, and the arrows indicate the orientation. (D) A text box where they can change the name of the label or delete it.

main challenge is a robust transformation of the frames of references, that is, from egocentric to allocentric, such that the spatial configurations of the salient objects in a scene can be abstracted from the imagery. This transformation is known to be difficult for machines, in the sense that robust mapping of salient objects from imagery often depends on knowledge of the specific scenes or situations, and the quality of the images, and so on. Humans, on the other hand, are clearly a lot more robust and flexible in this task than machines. On the other hand, it is also well known that even humans make a lot of mistakes in such tasks, especially when the number of objects is high. The main challenge in utilizing human computations therefore lies at how to derive methods that increase the robustness of mapping as data are aggregated and processed to generate the final schematic maps.

Solution: To increase robustness of the mapping, we capitalize on our knowledge in human spatial cognition to create scenarios that encourage better performance from the crowd workers. Specifically, in addition to asking humans to directly transform the 3D environment represented by the image to the 2D map, as shown in Figure 3(b), we created additional scenarios with different references on 2D grids, such that crowd workers were required to do mental rotation (90° , 180° , or 270°) to align the coordinates between the 3D scene in the image and the rotated 2D representation in the grid.

The main rationale was that this kind of mapping will encourage crowd workers to first mentally represent the spatial configurations of salient objects using an allocentric frame of reference *before* they performed the mental rotations. One should note that this was an inherent part of the task that was often omitted when workers adopted an egocentric frame of reference. If workers, for example, only paid attention to the location of a single object when placing it on the grid, then they had to perform mental rotation for each placement. On the other hand, if they encoded the relative locations of a group of objects and rotated them, then they only had to perform the mental

Fig. 3. The solutions to the two challenges: For “Different Viewing Angles,” the one assigned to the “up” and “down” arrows is expected to focus on the depth information; the one assigned to the “left” and “down” arrows is expected to focus on the width information. For “Mental Rotations,” the one assigned to 0° has no need to do mental rotation, and the one assigned to 90° , 180° , or 270° needs to do mental rotation to get the right answers.

rotation once and then place the group of objects on the rotated grid. In fact, because mental rotation was cognitively effortful and errorful, encoding the relative locations of multiple objects would likely lead to more accurate results. In case crowd workers did not represent groups of objects correctly, any error that they made in mental rotation would to locations of objects that were far from the correct locations, that is, error patterns tend to follow a multimodal distribution [Shepard and Metzler 1971]. Because of these reasons, we expect that the aggregated data distribution would allow us to easily select the better answers from the bad ones. To summarize, the goal of this rotated grid setting is to motivate workers to put more effort into generating the outcomes and help them focus on the relative locations of objects that allow us to improve the quality of the data.

Challenge II. The second challenge is how one can make sure that the data distribution is not biased and as close to the ground truth as possible. For example, Surowiecki [2005] states that diversity is one of the most important factors needed to form a wise crowd, as this will allow the correct answers to more easily stand out from the “noise,” as well as to decrease the likelihood that biased answers will cascade and impact the overall data distribution. Diversity can be introduced by encouraging crowd workers with different backgrounds or knowledge to perform the task or by providing different instructional guidance to crowd workers.

Solution: To tackle the second challenge, we instructed crowd workers to estimate the 3D information of an image from different viewing angles. Since Patterson et al. [1992] has shown that depth cues are helpful for better depth perception, as shown in Figure 3(a), we manually drew an arrow in the image in one of the cardinal directions (i.e., up, down, left, right). Crowd workers were told to imagine themselves standing in the environment and looking in the direction of the arrow when creating the 2D maps of salient objects. The goal of the arrow was to direct the attention of the crowd workers to adopt an egocentric perspective and focus on a single dimension while estimating the depth or width information of the 3D scene in the image.

For any given image, we published the spatial mapping task to a group of crowd workers and randomly assigned each of them a viewing angle (indicated by the arrow in the image) and a rotation (indicated by the corresponding arrow in the 2D grid). We will demonstrate the effectiveness of these two methods in Section 4.

Fig. 4. The general framework of the proposed system. I: Object Identification, including (1) Image rectangle clustering. (2) Objects' area identification. II: Position Extraction, including (3) Coordinate alignment for different grid rotations. (4) Label grouping. (5) Filtering labels' position by clustering and generating 4 schematic maps for each viewing angles. (6) Combination of four schematic maps from different viewing angles.

3.3. System Overview

The general framework of our proposed system is illustrated in Figure 4, which consists of two main components: One component is for object identification from the rectangles drawn in the image, and another component is for object position extraction from the labels mapping on the 2D grid. The output of the system is the information of the scene structures represented in the input picture, indicating not only what kinds of objects exist in the scene but also their 2D spatial relationship. We will provide further details of the two components in the rest of this section.

3.3.1. Object Identification. The prerequisite of picture schematization is to know what objects are located in the image. Since each crowd worker was told to draw rectangles to locate the objects and name them while performing the spatial mapping task, we could make use of the locations of the rectangles to identify the most salient objects in the image following steps.

First, we can calculate the center point of each rectangle and then apply the Density Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm (DBSCAN is a classic clustering algorithm for the task of class identification in spatial databases [Ester et al. 1996]) to cluster the center points of the rectangles (see Figure 4-(1)). Each cluster corresponds to an object in the image. In some cases, if some objects completely overlap each other, the current method would fail to figure them out, but these cases are rare and could be avoided by changing angles while taking pictures.

Second, we can calculate the overlapping areas of all the rectangles whose center points are within the same cluster as the area of the corresponding object in the image. The labels belonging to the same cluster are referring to the same object in the image but might have different words. We then process the text of all labels by eliminating extra white spaces, converting them all to lower case, removing stop-words, and performing stemming. We then choose the most frequently mentioned label as the

name for each cluster. Some clusters might have identical labels, and in this case the system automatically adds a suffix number to them, such as “chair 1” (see Figure 4-(2)).

After the above steps, we obtain one of the output results: the objects and their corresponding areas in the image (see Figure 4-C). In order to obtain the mapping positions of these objects in 2D grids, we still have to know to which object each rectangle refers. One intuitive way is to use the above clustering result to do that, that is, the rectangles are assigned to the object extracted from its belonging cluster. However, while some rectangles might not be clustered using their centers through the above steps, they might still overlap more than half with the area of some identified objects, and this was usually because the object was large. Instead of throwing those rectangles away as outliers, we choose to keep them because they still provide valuable information of the mapping positions in the 2D grid. So we calculate the percentages of the overlap areas of each rectangle with the identified objects and assign the most overlapped object to it. Those rectangles, which are not overlapping with any of the identified objects, should be discarded as spam. We will use this grouping information in the next steps (see Figure 4-(4)).

3.3.2. Position Extraction. The last step and the ultimate goal of the system is to generate a schematic map of a two-dimensional abstract representation that captures the relative distances and spatial locations of objects in the scene. Because each crowd worker was also told to place labels (corresponding to objects they draw in the image) onto a 2D grid from a different viewing angle while performing the spatial mapping task (see 3.2), the position data needed to be normalized before they were aggregated. The system applies the following steps to accomplish the goal.

First, due to the rotations of the 2D grid, the positions of the labels were in different coordinates. Therefore, they needed to be transformed to the same the grid orientation, as shown in Figure 4-(3). Then we merged the position data of different grid rotations into the same orientation from each viewing angle, that is we obtained four sets of merged positions data for each of the viewing angles of “up,” “down,” “right,” and “left.”

Second, for each set of merged positions data, the system utilized the grouping information obtained in the steps of object identification to tell which labels belonged to which identified object (see Figure 4-(4)). Note that if the true positions of two objects were close to each other, then they would likely be clustered together, and it would be hard to separate them apart using only clustering methods without knowing the group information.

Next, as shown in Figure 4-(5), for each group of labels, we again used the DBSCAN algorithm to cluster and filter out those labels that could not be clustered together. Then we calculated the center of each cluster as the position of its corresponding object on the 2D grid. We could then obtain four schematic maps for different viewing angles, as shown in Figure 4-J. In Section 4, we will discuss how we chose the parameters for the DBSCAN algorithm used in this step during the experiment and also demonstrate how mental rotations affected the clustering results in this step.

Last, we merged the four schematic maps extracted from different viewing angles into a single one (see Figure 4-(6)). We used the following procedure to merge them: (1) Find the objects appearing in all four schematic maps and order them by the number of labels belonging to them; (2) select the first object as the origin point; (3) for the next object, calculate its relative position to the origin point on each schematic map and then average the four positions as the final position for the object; (4) repeat the above steps until all objects are processed. In Section 4, we will discuss how the arrows on the pictures can sensitize crowd workers to the emphasized dimensions and also demonstrate how it can be applied to enhance the above procedure to improve the accuracy of the final schematic map.

We built a prototype system based on the above framework, which consisted of server and client components. The server side was implemented on a Linux server using the ExpressJS² framework and MongoDB,³ API (Application Programming Interface). The client side was implemented on browsers using the AngularJS⁴ framework to provide the interface for crowd workers to perform the spatial mapping task.

4. EVALUATION

As discussed in Subsection 3.2, we used two methods to improve the human computational part of the system: (1) we used rotated grids to encourage humans to represent salient objects as a group, such that they can pay more attention to the spatial relations of the objects as they schematize the imagery, and (2) we used arrows on the pictures to sensitize human perception to depth and width in the 3D environments represented by the 2D images. In order to verify whether these two methods could improve the system in steps (5) and (6) in Figure 4, we designed an experiment to showcase how the human data differed under different conditions.

4.1. Experiment Setup

We took some indoor images in a typical indoor environment of an office building to test the system. Those images were taken in ways such that they had the same level of difficulty in identifying and estimating the relative locations of salient objects. In particular, they had similar sizes of spaces and were taken at a similar distance and from the same perspective (i.e., straight with no angle). Although the system could take any image, we chose these images mostly to focus on the effects of the two methods, and thus we carefully controlled for possible effects induced by the contents of the images. We ran a pilot study by asking three participants to identify the objects for each image as many as possible. After manually checking their answers, we selected 12 images that had roughly the same number of objects, with similar consistencies among the pilot participants, and had no more than 20 objects in each image.

We did the experiment from July 29, 2014, to August 26, 2014, with a price tag of \$0.16 for each HIT, which required each AMT worker to perform the spatial mapping tasks on 4 images randomly chosen from the 12 images, which were presented in a random order. Each worker was restricted to do it only once based on their worker IDs and IP addresses. For each HIT, each image was randomly assigned to one of the four viewing angles (up, down, left, or right) and grid rotations (0°, 90°, 180°, or 270°). The randomization was done in a way such that each combination of viewing angles and grid rotations was assigned to at least 10 workers, which means that all viewing angles and rotations were tried with all images.

Once workers clicked on the experiment link, a tutorial would walk them through how they should perform the task as shown in Figure 2. They were asked to imagine that they were trying to help someone to identify the location from which the images were taken. After they finish the spatial mapping task on all four images, they were given a random code that they could use to fill in on the HIT site and get the reward. Given that sizes of the images and 2D grids could vary on different displays, we also collected the actual displayed sizes and saved the position data in ways such that they were on the same scale.

²<http://expressjs.com>.

³<http://mongodb.org>.

⁴<http://angularjs.org>.

A total of 512 workers were recruited, and they created 6,442 rectangles, with an average of 3.15 rectangles on each image from them, and each image had an average of 536.83 rectangles.

4.2. Data Preprocessing

We normalized the positions of the rectangles and labels based on their actual displayed sizes to the same scale (image size: $1,200 \times 1,600$; grid size: 800×800).

Then we performed the step of “Object Identification” described in Section 3.3.1 to extract the salient objects for each image and assign the rectangles to their corresponding objects. In this step, we applied the DBSCAN algorithm to cluster the center points of the rectangles. DBSCAN has two parameters that allow us to control the quality of the clusters: *eps* (Reachability distance) determines the maximum distance that two points are considered in the same cluster, and *MinPts* (Reachability minimum number of points) determines the minimum number of data points in a cluster. Because there were at most 20 common objects in each image, and each image had an average of 536.83 rectangles, we set $MinPts = 536.83/20 \approx 30$ (i.e., it should have at least 30 workers drawing rectangles at the same location to form a cluster). After measuring the actual spaces of those images, the average width and the height were around 10m, and we needed to control the offset of the drawn rectangles of each object within 0.5m, so by comparing that to the size of the images, we set $eps = \frac{1200+1600}{2} * \frac{0.5}{10} = 70$. After this process, we got 8 objects on average for each image. Then we calculated the percentages of the overlap areas of each rectangle with the identified objects and assigned the most overlapped object to it. We will use this group information to group their corresponding labels’ position data. On average, for each image, there were around 3.5% rectangles that did not overlap with any of the identified objects, which were discarded as spam. We illustrate those processed for Image #1 in Figure 5.

For each image, we split the positions data into 16 sets based on the viewing angles and grid rotations while they were collected. We aligned the coordinates of the positions data for the grid rotations of 90° , 180° , and 270° to the same coordinate of rotation 0° . Afterwards, we utilized the grouping information obtained above to classify them into groups based on their corresponding objects for each set of positions data. We illustrate these processes in Figure 6 for Image #1.

4.3. Effects of Rotated Grids

The first effect we tested was whether the rotated grids would allow us to more easily differentiate the good answers from the bad ones (and make the good ones more consistent), thereby improving the aggregated results. To test this effect, we analyzed the data from workers in different rotated grids and compared them. Specifically, for each set of positions data, we used the DBSCAN algorithm to cluster each group belonging to the same object and marked the nonclustered ones. We set *MinPts* to 80% of group size, which led to clusters that had at least 80% of the workers agreeing to the locations. In order to choose a proper *eps*, as illustrated in Figure 7, we plotted a graph of the relationship between different *eps* and the overall percentage of nonclustered points, given the grid image size 800×800 . Note that the percentages dropped sharply before the point of 400 and started to drop slowly afterwards. This was also consistent with the intuition that the point (400) was around half of the grid size, because if there were any mental rotations inducing the errors, the labels should be likely far away in another corner. So we decided to set *eps* to 400. After this process of clustering, overall there were around 12% of points identified as nonclustered ones.

We split the labels into two groups: one was instructed with grid rotation (90° , 180° , 270°) and the other without (0°). We then calculated the percentage of labels

Fig. 5. Illustration of object identification for Images #1: (a) The raw data of all the rectangles for Image #1. (b) Cluster the center point of the rectangles and extract the name for each cluster/object. The red ones were outliers; different shapes indicate different clusters/objects. (c) Extract the area of the object by calculating the overlap area of all the rectangles within its cluster. (d) Assign each rectangle to the object with which it overlapped the most; different shapes indicate different objects.

not in the clusters when *eps* was set at 400. We run a Kruskal-Wallis test to see if there was any significant difference of the nonclustered percentages between these two groups. The result (see Figure 8) showed that grid rotation did have a significant effect on the percentages ($H(1) = 8.9127, p < 0.01$), with a mean percentage of 0.17 for the rotated group and 0.10 for the nonrotated group, which indicated that while estimating the position of the same object, participants' answers tended to spread out more while the 2D grid was rotated. The results supported that mental rotation did make more participants' answers differ from others, which meant more data should be discarded while using a majoring voting/clustering technique to aggregate the final results. However, we will demonstrate how we can also benefit from the remaining answers later.

In order to see how the distribution of the label positions change before and after filtering out those nonclustered points, for each image, we selected the top three biggest clusters by the number of labels and split the labels in each cluster into two groups by the grid rotation (TRUE/FALSE) and then respectively calculated the average distance of them. We therefore performed two analyses: one with the nonclustered points and one without the nonclustered points.

Fig. 8. Percentages of points that were not clustered when the cutoff was set at 400 for each cluster with/without rotation.

Fig. 9. Average distance of labels grouped by rotated/nonrotated in each cluster before/after filtering non-clustered points with/without grid rotation.

a significant effect on the values ($H(1) = 6.3432, p < 0.05$). However, after filtering out nonclustered points, the average distances became **smaller** in the rotated group (42.31) than that (52.00) in the nonrotated group (see Figure 9).

Comparing the above analysis results, this indicated that, although grid rotation seemed to make the answers more inconsistent, after filtering out the nonclustered ones, the remaining answers tend to be more consistent. In other words, the rotation made it easier to tease apart the “good” answers and “bad” answers, and it also implied that those participants who put more effort into doing extra mental rotation were more likely to perform better than those who do not do the rotation. These results demonstrated the effectiveness of setting grid rotations to introduce some desirable difficulties that allow us to improve the quality of the data.

4.4. Effects of Emphasized Dimensions by Arrows

The second effect we tested was whether crowd workers would be sensitized to the emphasized dimensions indicated by the arrows on the images. If so, then the aggregation of the positions on the 2D grids would be more precise along the emphasized dimension. To test this, we first calculated the depth-width ratio for each participant. Specifically, for each participant who drew n rectangles for a given image, and for rectangle i the corresponding label coordinates on the 2D grid are $\{x_i, y_i\}$, we calculated

Fig. 10. Depth/width ratios for participant who drew n rectangles for a given image under different dimension emphasis.

the depth-width ratio as:

$$ratio_i = \frac{|y_{i+1} - y_i|}{|x_{i+1} - x_i|}$$

$$ratio = \frac{\sum_{i=1}^{n-1} ratio_i}{n - 1}$$

We calculated the ratios for all participants and wanted to see how these ratios differed between the participants who were given the viewing angles of “up” and “down” (they were expected to put more attention to the depth information) and those who were given “right” and “left” (they were expected to put more attention to the width information). We ran a Kruskal-Wallis test to check the difference of the ratios between these two groups. The result (see Figure 10) showed that dimension emphasis did have a significant effect on the ratios ($H(1) = 51.0737$, $p < 0.001$), with a mean ratio of 2.97 for the depth emphasis group and 1.95 for the width emphasis group. The results therefore supported the hypothesis that the participants were more sensitive to the emphasized dimensions, but it was still not clear whether or how this would lead to a more accurate aggregation result.

So we also wanted to see how the emphasized dimensions could impact the distribution of the label positions along both dimensions (depth/width). To test this, after filtering out those nonclustered points marked in Section 4.3, for each image, we selected the top three biggest clusters by the number of labels and split the labels in each cluster into two groups by its width emphasis (TRUE/FALSE) and then respectively calculated their standard deviations of x . We ran a Kruskal-Wallis test to see if there was any significant difference of the standard deviations of x between the two groups. The result showed that width emphasis (see Figure 11) did have a significant effect on the values of standard deviation of x ($H(1) = 8.1893$, $p < 0.01$), and the standard deviations of x were *smaller* in those groups that were provided by emphasizing width dimension (i.e., arrows of “left” and “right”), with a mean value of 32.27, compared to 42.27 without emphasizing. The similar analysis was conducted on the standard deviations of y by splitting the labels into two groups by its depth emphasis. Likewise, the result (see Figure 11) showed that depth emphasis did have a significant effect on the values of standard deviation of y ($H(1) = 3.99$, $p < 0.05$), and the standard deviations of y values were also *smaller* in those groups that were provided by emphasizing depth dimension (i.e., arrows of “up” and “down”), with a mean value of 44.61, compared to 51.20 without emphasizing.

Fig. 11. Standard deviations of x and y values under different dimension emphasis after filtering out nonclustered points.

Fig. 12. Generating a schematic 2D map of salient objects for Image #1: (a) The raw schematic maps by calculating the centers of each cluster for the image directions of up and down. (b) The raw schematic maps by calculating the centers of each cluster for the image directions of right and left. (c) The ground truth of the map for Image #1. (d) The generated schematic map of salient objects. (e) Image #1.

The above analysis indicated that the arrows not only sensitized workers to the emphasized dimension but also made their answers more consistent along the emphasized dimension. To capitalize this finding (take Image #1, for example, as shown in Figure 12), we used the procedures (see Figure 4-(5)) introduced in our system framework) to first aggregate the data to generate four schematic maps for each image (after filtering out the nonclustered points for each group points). Then we used the following procedure to merge them: (1) We identified the objects appearing in all four schematic maps and ordered them by the number of labels belonging to it (there were four objects found for Image #1: restroom, sofa, white board, and bulletin board). (2) We then selected the most salient object (restroom) as the reference point. (3) For the next

Table I. Riemannian Shape Distances between the Generated Schematic Maps and the Ground Truth for Different Methods

Img.	1	2	3	4	5	6	7	8	9	10	11	12
M1	0.32	0.22	0.16	0.04	0.22	0.19	0.72	0.15	0.71	0.20	0.51	0.22
M2	0.43	0.24	0.26	0.21	0.33	0.27	0.81	0.25	0.84	0.15	0.79	0.34

M1 refers to the method based on emphasized dimensions; **M2** refers to the method of ignoring emphasized dimensions.

most salient object (e.g., sofa), we calculated its relative distance on the y axis from the reference point for the schematic maps with up and down direction and averaged them as the y coordinate of this object; likewise, we calculated its relative distance on the x axis from the reference point for the schematic maps with right and left directions and averaged them as the x coordinate of this object. (4) We repeated the above steps until all salient objects are processed. Note the difference between step (3) comparing to that introduced in 3.3.2. We utilized the emphasized dimension to aggregate the depth and width information respectively instead of treating all of the data equally in step 3). We define the modified version as **M1** and the original version as **M2**.

Figure 12(d) shows the generated schematic map for Image #1 and the ground truth (c). By making comparisons side by side, generally speaking, the schematic map mostly captured the relative locations of the salient objects identified by the crowd. To evaluate the result quantitatively, we calculated Riemannian shape distances between the generated schematic maps and the ground truth for **M1** and **M2**. Table I showed the distances for the two methods. We did a paired t-test over the distances of these two methods and the result showed that the distances of **M1** were significantly smaller ($p < 0.001$) than those of **M2** obtained by averaging them from both dimensions, which clearly showed the benefits of using only the emphasized dimensions while merging the maps.

4.5. Summary

To summarize, we proposed a general framework for the picture schematization system and tested two methods to improve the quality of human crowd data. Results showed that (1) mental rotations that were encouraged by the rotated grids led to more consistent and accurate relative locations as we aggregated results through simple clustering methods and (2) crowd workers were sensitized to the emphasized dimension indicated by the arrows on the images, and thus the aggregation of the positions on the 2D grids was more precise.

5. DISCUSSION AND CONCLUSION

We showed how human computations could be utilized to be included in a system that identifies salient objects in an image and generates top-down schematic maps indicating their relative locations with reasonable accuracies. Though the generated maps are different parts of the whole scene, it is our first step towards building a system that can generate a complete map by combining different parts of pictures in a large scene.

Compared to existing computer vision methods or similar human computation systems, the proposed system did not make strong assumptions or demand external sensors or devices. In particular, we tested the effects of two manipulations that could improve the quality of outcomes. First, we found that, although asking crowd workers to create the schematic maps on rotated grids would create some difficulty for the workers, the results obtained were of higher quality: For those crowd workers who performed the correct mental rotations, their coordinates tended to be more precise, and answers from those who did poorly in the task tended to be much easier to separate from the good answers. This could possibly be caused by the fact that those crowd workers who

performed the correct mental rotation also tended to be those using allocentric frames of reference to encode the relative locations of objects. In other words, the extra mental rotation actually helped to improve the final outcomes of the schematic maps. Somewhat paradoxically, this results suggests that certain levels of *desirable difficulty* can make human computation systems more robust and accurate, as the extra difficulty will implicitly guide human cognition to work harder in ways that generate better data (which otherwise would have been underconstrained).

Second, we found that drawing arrows on images to sensitize crowd workers on depth and width dimensions represented by 2D images could improve the precision in estimating relative locations, especially along the sensitized dimensions. While we tested only two dimensions, it is reasonable to assume that one could generate more 3D dimensions to boost the accuracies of the final aggregated coordinates of the objects. In order to further improve our system in the future, a spatial cognition model can be introduced to figure out the underlying factors that have a direct effect on the output of the system. However, in general, we found that our results implied that sensitization is an effective method to generate more robust human data that are useful for interpreting 3D information from 2D images.

Based on the results, we showed that simple clustering methods could be used to merge data from multiple crowd workers in different conditions to improve the final schematic maps. The current study is limited in its data size, so more future research is needed to see how well the method can be generalized to other situations and how well it scales when the environment becomes more complex. In this study, the images chosen were taken from an indoor environment with familiar objects, and thus we found that the objects identified by crowd workers were highly consistent. It is possible that in more complex environments, in which many unfamiliar objects may exist, the consistency will be lower and more crowd workers may be needed. It is also possible that other techniques can be used to simplify the tasks even further, such as having workers focus on different parts of the image or to utilize other semantic information that are inferred from the scene. It is also possible that computer vision techniques can be combined with human computations to direct workers' attention to areas that optimize outcomes.

Given that the images used in the study were relatively simple, there was only a small number of objects to identify and therefore the ranking of salient objects was not used when generating the schematic maps (i.e., all objects identified were put on the map). One can imagine, however, that the ranking results could also be useful for knowing how important or salient the objects are for different purposes (e.g., by providing different instructions to crowd workers) and to generate different schematic maps for different purposes (e.g., accessibility, emergency, etc.). It is also possible to cluster these objects based on their semantic similarities, such that a better scene understanding systems can be developed on top of the proposed system.

While the methods we tested in this article were specific to picture schematization, the general idea of leveraging understanding of human perceptual (or cognitive) processes to motivate and nudge crowd workers to generate high-quality data is generally applicable to other tasks. In particular, methods that lead to distributions that generate more robust data for example, by introducing steps that require desirable cognitive or perceptual effort to perform will be a good general principle for people interested in designing better instruction or method for their specific tasks.

ACKNOWLEDGMENTS

Many thanks to the anonymous reviewers for their valuable comments that helped us improve the earlier draft of this work. This article also benefited from the discussion with Prof. Gajos Krzysztof, Prof. Chuancai Liu, and Dr. Anbang Xu.

REFERENCES

- Moustafa Alzantot and Moustafa Youssef. 2012. Crowdinside: Automatic construction of indoor floorplans. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*. ACM, New York, NY, 99–108.
- Anthony J. Aretz. 1990. *Cognitive Requirements for Aircraft Navigation*. Technical Report. DTIC Document.
- Robert Battle and Dave Kolas. 2012. Enabling the geospatial semantic web with Parliament and GeoSPARQL. *Semantic Web* 3, 4 (2012), 355–370.
- Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. 2006. Surf: Speeded up robust features. In *Proceedings of the European Conference on Computer Vision (ECCV'06)*. Springer, Berlin, 404–417.
- Vladimir Bazjanac. 2008. IFC BIM-based methodology for semi-automated building energy performance simulation. *Lawrence Berkeley National Laboratory* (2008).
- Michael S. Bernstein, Greg Little, Robert C. Miller, Björn Hartmann, Mark S. Ackerman, David R. Karger, David Crowell, and Katrina Panovich. 2010. Soylent: A word processor with a crowd inside. In *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology*. ACM, New York, NY, 313–322.
- Jeffrey P. Bigham, Chandrika Jayant, Hanjie Ji, Greg Little, Andrew Miller, Robert C. Miller, et al. 2010. VizWiz: Nearly real-time answers to visual questions. In *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology*. ACM, New York, NY, 333–342.
- Yohan Chon, Yunjong Kim, and Hojung Cha. 2013. Autonomous place naming system using opportunistic crowdsensing and knowledge from crowdsourcing. In *Proceedings of the 2013 ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN'13)*. IEEE, 19–30.
- Yohan Chon, Nicholas D. Lane, Fan Li, Hojung Cha, and Feng Zhao. 2012. Automatically characterizing places with opportunistic crowdsensing using smartphones. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*. ACM, New York, NY, 481–490.
- Lynn A. Cooper. 1976. Demonstration of a mental analog of an external rotation. *Percept. Psychophys.* 19, 4 (1976), 296–302.
- Peng Dai, Mausam Daniel, and S. Weld. 2010. Decision-theoretic control of crowd-sourced workflows. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence (AAAI'10)*.
- Jia Deng, Jonathan Krause, and Li Fei-Fei. 2013. Fine-grained crowdsourcing for fine-grained recognition. In *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'13)*. IEEE, 580–587.
- Moustafa Elhamshary and Moustafa Youssef. 2014. CheckInside: A fine-grained indoor location-based social network. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, New York, NY, 607–618.
- Pablo Espinace, Thomas Kollar, Alvaro Soto, and Nicholas Roy. 2010. Indoor scene recognition through object detection. In *Proceedings of the 2010 IEEE International Conference on Robotics and Automation (ICRA'10)*. IEEE, 1406–1413.
- Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD'96)*, Vol. 96. 226–231.
- David Geiger, Michael Rosemann, and Erwin Fieft. 2011. Crowdsourcing information systems: A systems theory perspective. In *Proceedings of the 22nd Australasian Conference on Information Systems (ACIS'11)*.
- Kotaro Hara, Vicki Le, and Jon Froehlich. 2013. Combining crowdsourcing and google street view to identify street-level accessibility problems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, 631–640.
- Shih-Wen Huang and Wai-Tat Fu. 2013. Don't hide in the crowd!: increasing social transparency between peer workers improves crowdsourcing outcomes. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, 621–630.
- Panagiotis G. Ipeirotis, Foster Provost, and Jing Wang. 2010. Quality management on amazon mechanical turk. In *Proceedings of the ACM SIGKDD Workshop on Human Computation*. ACM, New York, NY, 64–67.
- Bela Julesz. 1960. Binocular depth perception of computer-generated patterns. *Bell Syst. Tech. J.* 39, 5 (1960), 1125–1162.
- Raphael Labayrade, Didier Aubert, and J.-P. Tarel. 2002. Real time obstacle detection in stereovision on non flat road geometry through “v-disparity” representation. In *Proceedings of the 2002 IEEE Intelligent Vehicle Symposium*, Vol. 2. IEEE, 646–651.
- Ki Joune Li and Ji Yeong Lee. 2013. Basic concepts of indoor spatial information candidate standard IndoorGML and its applications. *J. Kor. Spatial Inform. Soc.* 21, 3 (2013), 1–10.

- Beyang Liu, Stephen Gould, and Daphne Koller. 2010. Single image depth estimation from predicted semantic labels. In *Proceedings of the IEEE 2010 Conference on Computer Vision and Pattern Recognition (CVPR'10)*. IEEE, 1253–1260.
- David Marr and Tomaso Poggio. 1979. A computational theory of human stereo vision. *Proc. Roy. Soc. Lond. B: Biol. Sci.* 204, 1156 (1979), 301–328.
- David Oleson, Alexander Sorokin, Greg P. Laughlin, Vaughn Hester, John Le, and Lukas Biewald. 2011. Programmatic gold: Targeted and scalable quality assurance in crowdsourcing. *Hum. Comput.* 11, 11 (2011).
- Robert Patterson, Linda Moe, and Tiger Hewitt. 1992. Factors that affect depth perception in stereoscopic displays. *Hum. Factors* (1992).
- Dale Purves and R. Beau Lotto. 2011. *Why We See What We Do Redux: A Wholly Empirical Theory of Vision*. Sinauer Associates.
- Alexander J. Quinn and Benjamin B. Bederson. 2011. Human computation: A survey and taxonomy of a growing field. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 1403–1412.
- Huaming Rao, Shih-Wen Huang, and Wai-Tat Fu. 2013. What will others choose? how a majority vote reward scheme can improve human computation in a spatial location identification task. In *Proceedings of the 1st AAAI Conference on Human Computation and Crowdsourcing*.
- M. Schafer, Christian Knapp, and Samarjit Chakraborty. 2011. Automatic generation of topological indoor maps for real-time map-based localization and tracking. In *Proceedings of the 2011 International Conference on Indoor Positioning and Indoor Navigation (IPIN'11)*. IEEE, 1–8.
- Roger N. Shepard and Lynn A. Cooper. 1986. *Mental Images and Their Transformations*. MIT Press.
- Roger N. Shepard and Jacqueline Metzler. 1971. Mental rotation of three-dimensional objects. *Science* 171 (1971), 701–703.
- Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. 2008. Cheap and fast—but is it good?: Evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 254–263.
- Alexander Sorokin and David Forsyth. 2008. Utility data annotation with amazon mechanical turk. In *Proceedings of the 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW'08)*. 1–8.
- James Surowiecki. 2005. *The Wisdom of Crowds*. Anchor.
- Agnes Swadzba and Sven Wachsmuth. 2011. Indoor scene classification using combined 3d and gist features. In *Proceedings of the Asian Conference on Computer Vision (ACCV'10)*. Springer, Berlin, 201–215.
- Eric Turner and Avidesh Zakhor. 2014. Floor plan generation and room labeling of indoor environments from laser range data. In *Proceedings of International Conference on Computer Graphics Theory and Applications*.
- Luis Von Ahn and Laura Dabbish. 2004. Labeling images with a computer game. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, 319–326.
- Anbang Xu, Shih-Wen Huang, and Brian Bailey. 2014. Voyant: Generating structured feedback on visual designs using a crowd of non-experts. In *Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work & Social Computing*. ACM, New York, NY, 1433–1444.

Received January 2015; revised December 2015; accepted December 2015