# OLIV: An Artificial Intelligence-Powered Assistant for Object Localization for Impaired Vision

Linda Wang — University of Waterloo, ON, Canada
Anshuman Patnik* — University of Waterloo, ON, Canada
Edrick Wong* — University of Waterloo, ON, Canada
Justin Wong* — University of Waterloo, ON, Canada
Alexander Wong — University of Waterloo, ON, Canada
*: Equal contribution

## Abstract

This paper introduces OLIV, a novel end-to-end artificial intelligence-powered assistant system designed to aid individuals with impaired vision in their day-to-day tasks in locating displaced objects. To achieve this goal, OLIV leverages the current advances in AI-based speech recognition, speech generation, and object detection to understand the user's request and give directions to the relative location of the displaced object. OLIV consists of three main modules: i) a speech module, ii) an object detection module, and iii) a logic unit module. The speech module interfaces with the user to interpret the verbal query of the user and verbally responds to the user. The object detection module identifies the objects of interest and their associated locations in a scene. Finally, the logic unit module makes sense of the user's intent along with the localized objects of interest, and builds a semantic description that the user can understand for the speech module to convey verbally back to the user. Initial results from a proof-of-concept system trained to localize four different types of objects show promise to the feasibility of OLIV as a useful aid for individuals with impaired vision.

## 1 Introduction

There are an estimated 253 million people who live with visual impairment and 36 million of those are legally blind [1]. For these individuals, undertaking menial daily tasks, such as navigating, detecting obstacles in the way, identifying and locating objects, are demanding and require prior knowledge of the scene to accomplish individually. Using recent advances in artificial intelligence (AI), specifically in the areas of speech recognition and object detection, AI technology can empower these individuals, as well as increase independence and productivity in their daily lives.

Current AI technologies for visual impairment have the ability of reading printed or handwritten text, describe scenes, recognize currency, as well as recognize faces and emotions [2, 3, 4]. There has also been ongoing research in areas of mobility aid products to reduce the stress and discomfort when traveling to unfamiliar environments, such as using a robotic guide powered by computer vision and RFIDs, to navigate and detect obstacles [5, 6, 7]. In addition, there has also been research into combining computer vision and speech models to educate children in their primary task of learning to identify objects without supervision [8]. Although there have been many advances in solutions for people who are visually impaired, there are still many obstacles that they face in their daily lives that have not been addressed. Furthermore, the majority of current solutions require the use of hand-held devices, which may not be convenient when both hands are required. As such, we are motivated to investigate and explore hands-free solutions to tasks currently not well addressed with existing solutions.

Through user interviews with members of the Canadian National Institute for the Blind (CNIB), one particularly important task that has not been addressed or well explored, and is found to be of significant benefit for those living with visual impairment is the localization of displaced objects. Currently, these individuals use the mental models they build of the environment to remember where items are placed. However, if an individual places an item in the wrong location by accident, he or she is unable to locate the item since the mental model has not been updated and have to ask someone for help.

Motivated to tackle this important problem faced by those with visual impairments, we propose OLIV (**O**bject **L**ocalization for **I**mpaired **V**ision) to tackle this problem leveraging the advancements of AI-based speech recognition, speech generation, and object detection to direct the individual to a displaced item without using a hand-held device in an office environment. In addition, OLIV also reduces their cognitive workload of remembering where each item is while increasing their independence.

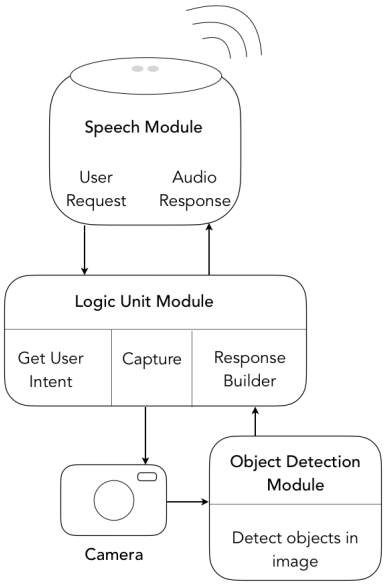


*Fig. 1:* System overview of OLIV, which consists of three main components: a) a speech module that enables the user to interact with the system and for the response to be communicated back to user, b) an object detection module that identifies the objects in the captured image and provides type and location information of each object, and c) a logic unit module that processes user request to understand user intent, initializes scene capture, and constructs an answer from object detection results and user intent.

## 2 Methods

The proposed OLIV system is designed to aid individuals with visual impairment to locate displaced items, and is comprised of three main components: i) a speech module, ii) an object detection module, and iii) a logic unit module. A system overview of OLIV is shown in Fig. 1. When a user wishes to locate a displaced item, he or she will send a verbal query to OLIV. The speech module then interprets the user's verbal query and sends the interpreted query to the logic unit module to understand the user's intent. The logic unit module then triggers a snapshot of the scene using a camera, which is received by the object detection module. The object detection module then identifies all objects of interest and their corresponding object types and locations, and feeds that back to the logic unit module. Based on the user's intent along with the object type and location information, the logic unit module then calculates the relative location of the queried object to the closest landmark object and constructs a semantic description providing directions for the user to locate the object. Finally, this semantic description is sent to the speech module to be verbally conveyed back to the user. A detailed description of each of the three main components of OLIV are explained in detail in the following sections.

### 2.1 Speech Module

The speech module of OLIV is responsible for interpreting the verbal query from the user as well as providing a verbal response to the user's query to inform the user with directions on where the displaced item is. To realize the capabilities needed for the speech module, off-the-shelf commercial smart assistant solutions such as Microsoft Cortana were leveraged as the interface and feedback to the user not only because of their state-of-the-art capabilities in speech recognition and speech generation, but also based on widespread adoption of commercial devices currently using these voice assistance solutions and results obtained from the interviews we conducted. For example, current products such as Microsoft SeeingAI use audio feedback to communicate the results to the

user [3]. When a user points their device to a scene, SeeingAI uses audio feedback to communicate the description of what is currently in view. When interviewing members of CNIB, they expressed that they prefer audio feedback as a response from a system. Also from the interviews, some individuals with visual impairment stated that they already use voice assistant devices because of its simple interface to set up alarms, timers and other simple tasks. In addition, by utilizing off-the-shelf smart assistant solutions, the devices are not only used for this particular system but also what users currently use them for. Home assistants are also affordable and accessible, making these devices easy to obtain.

## 2.2 Object Detection Module

The goal of the object detection module of OLIV is to identify which objects are in the scene, identify what type of objects they are, as well as locate where they are in the image. In order to achieve this goal, there are two main considerations that must be investigated. First, a suitable dataset that contains objects that are representative of those found in the user's environment needs to be identified. Second, object detection models that can perform well on this type of data need to be identified, trained, and compared based on a number of different factors such as accuracy (via performance metrics such as intersection over union (IoU) and mean average precision (mAP)) and speed to identify which is the most appropriate for the proposed OLIV system. Here, we primarily focused on object models based on deep convolutional neural networks given their demonstrated state-of-the-art performance for a wide variety of object detection tasks in literature [9, 10].

**Dataset** Along with the recent advancements in object detection models, there have also been the introduction of a wide variety of large publicly available datasets, such as Common Objects in Context (MS-COCO) and ImageNet Large Scale Visual Recognition Challenge (ILSVRC), which are used as benchmark for comparisons [11]. The MS-COCO dataset consists of 80 object categories with a total of 330K images [12] and the ILSVRC consists of hundreds of object categories with over a million images [13]. Both these datasets have collected ground truth annotations, including labels and bounding boxes, for each image [12, 13].

By leveraging the existence of these datasets, a model can be trained to detect objects in a workplace. In addition, since this task is limited to just an office environment, only a subset of relevant object categories are needed. The MS-COCO dataset consists of 24 objects that might appear in a workplace and ILSVRC consists of 29 objects.

As a proof of concept, MS-COCO was used to pretrain the model, followed by training with a custom dataset consisting of four objects: bottle, cup, bowl and kettle. The custom dataset consisted of 800 images in total. Data augmentation was performed on the custom dataset via horizontal and vertical flips, leading to a dataset of 1849 total images (subdivided into 1387 images for training and 462 images for testing). MS-COCO was chosen because the dataset is of higher quality and a more widely used for object detection training than ILSVRC [12, 11]. In the future, the goal is to use a subset of ILSVRC in combination with a subset of MS-COCO, which contain workplace objects, to obtain more data for training.

**Models** A challenge in object detection is that, while a number of powerful deep convolutional neural network architectures have been demonstrated to achieve high accuracies in both localization and mAP [11], these networks have very high computational complexities. This results in a speed-accuracy trade-off that becomes critical for OLIV, which has real-time, low-latency requirements. For instance, when a user asks for where an item is, the objection detection model must be accurate enough so that the user is directed in the right direction. However, the model must also be fast enough so that the user is not waiting a long time for a response, allowing a more natural experience.

In the 2017 ILSVRC object localization challenge, an mAP of 0.73 was achieved using the provided data [14]. The winning team leveraged a residual-inception network architecture as the feature extractor; however, even though this feature extractor is the most accurate, the speed of the resulting object detection network is relatively slow [11]. Based on comparisons in [11], Faster R-CNN with a residual network architecture as the feature extractor was found to achieve the best balance between speed and accuracy. Furthermore, a recent convolutional deep neural network architecture that has gained considerable popularity is SSD with a MobileNet network architecture [15, 16, 11] as the feature extractor, which has been shown to give strong object detection performance while providing fast inference speed. A comparative analysis was performed on Faster R-CNN with a residual-inception network architecture, Faster R-CNN with a residual network architecture, as well as SSD with a MobileNet network architecture based on the accuracy-speed requirements of OLIV, and it was determined that SSD with a MobileNet architecture provided the best accuracy-tradeoff that still meets operational requirements.

Once a model has been trained and tested to meet the requirements of speed and accuracy, the model was incorporated into OLIV. Fig. 2 shows an example of how the object detection module takes input from the camera, adjusts the input to a desired size, sends the image through the model to detect objects in the image and then sends the detected objects to the logic unit module. The object detection module can take in images of any size or resolution, which makes the system robust to different camera configurations. Before sending the image through the model, the image is resized to improve inference speed. In this example, the image is resized from 640x480 to 300x300. After the image is sent through the object detection model, the output is the most probable objects and their corresponding bounding boxes. For this example, the top three most probable objects are the monitor, keyboard and water bottle.

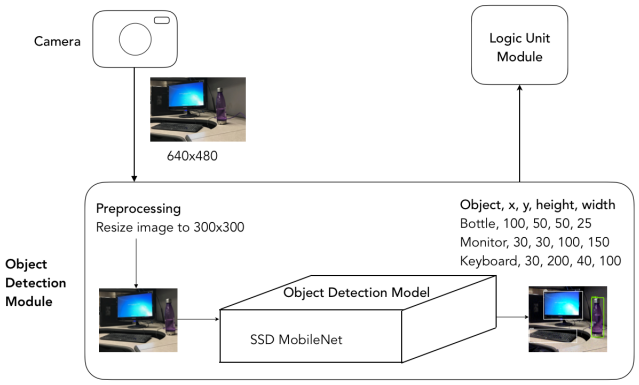


*Fig. 2:* Object detection module overview. Once the camera takes a snapshot of the scene, the image is passed to the object detection module. First, preprocessing is done, such as re-sizing the image to 300x300. Then, the preprocessed image is sent to an object detection model where the most probable object type and bounding box of each object are determined for each object. The object types and corresponding bounding boxes are then sent to the logic unit module.

## 2.3 Logic Unit Module

The purpose of the logic unit module is to interact with the speech module with the object detection module to understand user intent and construct a semantic description based on intent and object detection results so that a verbal response can be provided for the user. The logic unit module achieves these connections using three main functions. The first function is to determine what the object of interest is based on the interpreted query passed in by the speech module. The next function is to initiate image capture so the object detection module can determine what objects are in the scene. The third function is to build a semantic description response to pass back to the speech module based on the objects in the scene and the locations and the user intent for what he or she is looking for. The first and last function of the logic unit module uses a web-hook, which is a way for an application to deliver data to other applications, to relay the request from the speech module and the response to the speech module, respectively. Once the request is received, the sentence is parsed for a word that matches one of the objects in the data set or a synonym of it. For instance, if a user asks for a "bottle", this will correspond to the "water bottle" class in the dataset. For the second function of requesting the camera to capture the scene, the camera is connected to the system as a video source so only a command is needed to activate the camera. Once the semantic description response builder, shown in Fig. 3, receives the type and locations of all the objects identified by the object detector module, it identifies the target object of interest and calculates which landmark object is closest to the target object and where that target object is located relative to the landmark object. A landmark object is an object that the user is already well aware of where it is located and is rarely moved. For instance, in Fig. 3

the landmark objects are “monitor“ and “keyboard“. Using the relative location, such as “lower right“, the response builder builds a sentence for the speech module to reply back to the user verbally. An example is shown in Fig. 3.

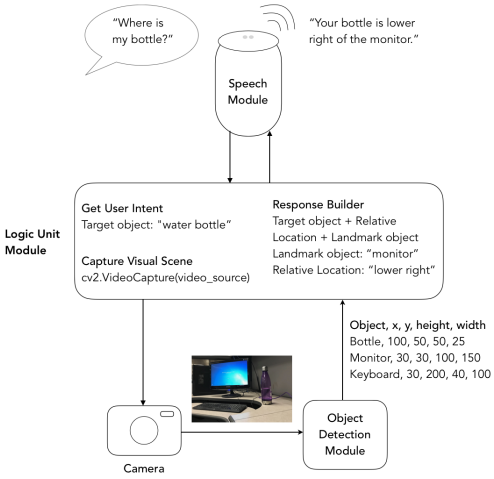


*Fig. 3:* Logic unit module. First, the user request gets processed, then the camera command is executed. After the image is processed by the object detection module, the types and locations of each object identified are sent to the response builder. The response builder finds the closest landmark object to the object of interest based on the user’s intent, calculates the relative location, and builds the user response. Lastly, the response is communicated to the user via the speech module.

## 3 Results

As a proof of concept, an initial OLIV system is constructed, with the object detection module leveraging a SSD object detection model with a MobileNet architecture that is trained on a custom dataset. The model is trained for a total of 700,000 iterations, achieving a test mAP of 96.5% with IoU from 0.5 to 0.95. Inferencing was performed on an Intel Core i7-4700MQ 8-core CPU at 2.40 GHz, with the average runtime of detecting and identifying objects taking an average of 58.1 ms with a standard deviation of 2.8 ms.

| | | | | |
|---|---|---|---|---|
| **User Query** | “Where is my bottle?” | | | |
| **Class** | 1 —> bottle | | | |
| **Confidence** | 99.95% | | | |
| **Bounding box** | x-min | x-max | y-min | y-max |
| | 0.67201275 | 0.81669992 | 0.14336389 | 0.74318022 |
| **Answer** | “The bottle is right of the monitor” | | | |

*Fig. 4:* Proof-of-concept OLIV system. In this example, the user asks ”Where is my bottle?”. Based on the detected ’bottle’ object and its location (shown by a green bounding box) and its location relative to the landmark object (in this case, it is a stationary monitor), the semantic description response verbally conveyed to the user is “The bottle is right of the monitor“.

## 4 Discussion

In this paper, we propose OLIV, an AI-powered system for aiding individuals with visual impairment in locating displaced items in the environment. OLIV offers both advantages and limitations that needs to be considered. One advantage is that OLIV is voice-driven and thus omits the use of hands and is unobtrusive. Users do not need to pull out their mobile device to scan the room. However, a limitation with the current proof-of-concept embodiment of OLIV is that it currently leverages just one camera, which limits the space that it is able to cover. In future work, images of the scene can be captured from multiple camera angles and then stitched together to overcome the limitations of one camera. Another advantage is that OLIV can be used not only by people with visual impairment but also those who experience dementia and may not remember where they placed an object. This system can remind them where the item is, which also increases their independence.

For the object detection module, future work includes training on more objects and exploring more environments, not just the workplace. In addition, further investigation of more architectures will be conducted to better understand the speed accuracy trade-off. For the speech and logic unit module, different ways of conveying the message to be more user friendly and conversational should be explored. Currently for the system to work, there must be a camera placed in the scene. However, it is not feasible to have cameras everywhere, so a more portable hands-free solution, where the user can take the system anywhere, should also be explored to increase independence when in a new environment.

## Acknowledgments

We like to thank NSERC, Canada Research Chairs program, and Microsoft.

## References

[1] Rupert R. A. Bourne et al. Magnitude, temporal trends, and projections of the global prevalence of blindness and distance and near vision impairment: a systematic review and meta-analysis. *The Lancet Global Health*, 5(9):e888 – e897, 2017.

[2] Patrick Clary. Lookout: an app to help blind and visually impaired people learn about their surroundings, May 2018.

[3] Xiaoyong Zhu et al. How to develop a currency detection model using azure machine learning, May 2018.

[4] Jamie Pauls. An evaluation of orcam myeye 2.0, Aug 2018.

[5] Vladimir Kulyukin et al. Robot-assisted wayfinding for the visually impaired in structured indoor environments. *Autonomous Robots*, 21(1):29–41, Aug 2006.

[6] Vladimir Kulyukin et al. A robotic wayfinding system for the visually impaired. *Proceedings of the Sixteenth Innovative Applications of Artificial Intelligence Conference*, pages e864 – e870, 2004.

[7] Alberto Rodriguez et al. Assisting the visually impaired: Obstacle detection and warning system by acoustic feedback. *Sensors*, 12(12):17476–17496, 2012.

[8] B. K. Balasuriya et al. Learning platform for visually impaired children through artificial intelligence and computer vision. In *2017 11th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*, pages 1–7, Dec 2017.

[9] Alex Krizhevsky et al. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[10] Christian Szegedy et al. Deep neural networks for object detection. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 2553–2561. Curran Associates, Inc., 2013.

[11] Jonathan Huang et al. Speed/accuracy trade-offs for modern convolutional object detectors. *CoRR*, abs/1611.10012, 2016.

[12] Tsung-Yi Lin et al. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014.

[13] Olga Russakovsky et al. Imagenet large scale visual recognition challenge. *CoRR*, abs/1409.0575, 2014.

[14] Large scale visual recognition challenge 2017.

[15] Wei Liu et al. SSD: single shot multibox detector. *CoRR*, abs/1512.02325, 2015.

[16] Andrew G. Howard et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *CoRR*, abs/1704.04861, 2017.