Augmenting the SIFT Descriptor Set to Create a Navigation Assistant for the Visually Impaired

Vineet Kosaraju, Rishabh Jain

Abstract—The issue of mobility is considered among the foremost of concerns for visually impaired individuals. This work develops a Navigation Assistant that aids the visually impaired with mobility in a known environment such as a household, by combining the principles behind Scale-Invariant Feature Transform (SIFT) and Content-Based Image Retrieval (CBIR). Furthermore, this work proposes and tests several techniques such as Color Segmentation, Difference of Images, and Contrast Manipulation to improve the Navigation Assistant. Using a video stream that was obtained from a small camera that the user carries, key objects in the household were identified using the software developed. The Navigation Assistant created using the enhancements was markedly improved in both accuracy and speed from the control Assistant, and is therefore applicable to real-life usage by visually impaired individuals.

Index Terms—CBIR, SIFT, Color Segmentation, Difference of Images.

I. INTRODUCTION

A. Background

In the United States alone, visual impairments, including blindness, affects 6.6 million individuals and costs the federal government 5.5 billion dollars annually [1, 2]. Almost 50% of the blind people feel, to some degree, cut off from their social environment. Yet, help is often seriously lacking: a study from the United Kingdom found that “in the year after registration, less than a quarter (23%) of people who lost their sight say they were offered mobility training to help them get around independently” [3].

In particular, the issue of mobility is often the most pressing to blind people. Mobility is so relevant to blind individuals because it can have applications as varied as locating an object in one’s home to navigating around a crowded city. Currently, the most common aids to mobility are the white cane and the guide dog, both of which are accurate, easy to use, and affordable [4]. Therefore, it is crucial that any technological approach to guiding blind individuals and assisting in mobility must be as affordable, easy to use, and accurate as a guide dog or white cane.

Several technologies have been developed in the last few decades to assist blind people with mobility. However, each has one or more drawbacks related to the three criteria stated, as discussed below.

Surgical approaches to assisting blind individuals with mobility include retinal implants and using videos to directly stimulate the visual cortex of the brain. However, such technology, in its current state, produces pixelated vision, and is highly invasive [6]. Additionally, surgical procedures are often expensive, costing thousands of dollars [7].

Manuscript Received March, 2014.

Vineet Kosaraju, The Harker School, San Jose, California, USA.
Rishabh Jain, The Harker School, San Jose, California, USA.

Other approaches include the usage of Global Position System (GPS) technology and Radio-Frequency Identification (RFID) tags, each with its own drawbacks. For instance, GPS technology is only accurate with a radius of ten meters [8]. For a blind person who is trying to find a backpack in a room, a GPS-based system would not prove effective. Although RFID tags can address this issue of low resolution, they have the drawbacks of high cost, low durability, and overload of sensory data, which would hinder rather than assist mobility [9].

Another key method of increasing the mobility of the blind involves processing the surroundings, identifying an image, and converting it into a voice command. This research focuses on applying principles of computer science and image processing to assist the visually impaired (VI) in navigating their environment. In particular, this research features original software combining the principles of content-based image retrieval (CBIR) with scale invariant feature transform (SIFT), a method for detecting key features in an image [5]. The proposed navigation system for the visually impaired comprises of a navigation assistant that identifies the objects in the surroundings and conveys the information to the user by using Text-To-Speech technology [10].

II. NAVIGATION ASSISTANT FOR THE VISUALLY IMPAIRED

The autonomous guidance system developed mainly focuses on effective navigation in a known environment, such as a household. As such, two main features must be emphasized. First, the vision system must be able to detect objects, regardless of size and orientation, in real time. Furthermore, the vision system must be robust in detection regardless of noise, illumination, and contrast of the setting.

The created system uses a real-time video stream, which can be obtained from any device with a camera. In the stream, a frame rate of 30 frames per second was achieved; however, this large amount of data is superfluous in the detection of objects in real-time. Instead, a frame is extracted, analyzed, and used in the system every 0.1 seconds, resulting in 10 frames per second.

The database of tagged images must be generated once in the VI’s house. Thus, a personal aide must enter the tags into the system once by taking pictures of the various objects in order to ensure the validity of the tagged database. After the initial database has been created with minimal efforts from the personal aide, the VI can then use the system. By analyzing the video stream provided by the camera, the system will detect objects within a configurable distance by matching each frame of the video stream against the newly created database and will then state the object detected using Text-To-Speech technologies (Figure 1).

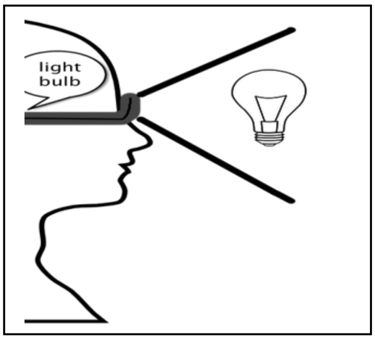


Figure 1. An example prototype of the navigation assistant.

Here, the device is shown to be resting on the user's head while it detects a light bulb. Using Text To-Speech technologies, the device says "Light Bulb" out loud.

A. Implications of using SIFT for Navigation Assistant

The authors propose to use SIFT for the navigation assistant for three main reasons: affordability, ease of use, and accuracy. These three points are essential for a successful navigation system as they provide features lacking in other aforementioned systems. The device specifications for our algorithm are flexible. The only necessary components are a processor and a camera. We were able to create a real-time assistant on an iPhone with an Apple A5 chipset; however, processor specifications are open-ended. As the resolution of photographs is reduced in the system, a high-resolution camera is also unnecessary. The algorithm's ease of use allows for greater control while keeping the visually impaired user input to a minimum. Furthermore, the real-time nature of the algorithm provides the user with instantaneous contextual feedback. A flowchart depicting the assistance process is shown below (Figure 2).

Path finding systems for the visually impaired are in existence today; however, few capitalize on the ability to detect objects [12]. While path finding remains important, another important augmentation to a visually impaired system is the detection of objects so users may understand their surroundings at a deeper level.

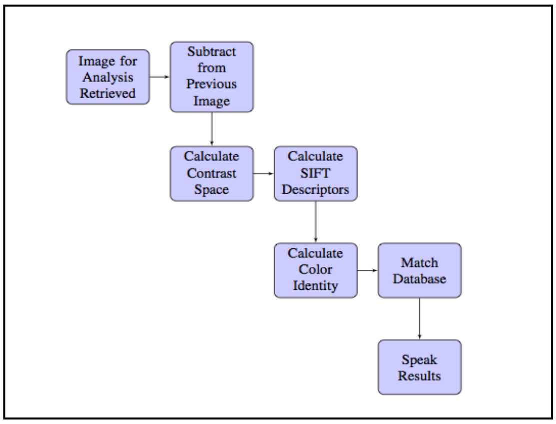


Figure 2. The assistance process is depicted here.

First, an image is retrieved from the video stream. Next, the Difference of Images algorithm removes redundant keypoints and the contrast space algorithm ensures the SIFT system is robust. Afterward, SIFT descriptors are created and the object is separated by colors and then matched. Finally, the name of the object is spoken to the visually impaired user As SIFT provides scale-invariant, rotation-invariant, and illumination invariant descriptors, this algorithm provides the basic framework for a robust object detection system; however, it must be improved in terms of speed and accuracy to be fully applicable to the navigation assistant [14]. While SIFT is robust, it is computationally intensive and cannot be run in real-time on a small device such as a mobile phone without several key enhancements. Thus, we have proposed several key improvements so that SIFT can become applicable to the navigation assistant for the visually impaired.

Although the current SIFT algorithm is scale, rotation, and illumination invariant, its accuracy is significantly reduced when matching blurry images [13]. This would be a serious obstacle for the assistant, because sharp and sudden movements from the user could cause blurry frames from a real-time video, which would then lead to misinterpreted objects, as identified keypoints would not be indicative of the original image [11]. In order to make SIFT more robust, the authors propose a technique designed to improve the accuracy of the algorithm on blurry images. Additionally, SIFT can generate hundreds to thousands of keypoints per image, and matching these keypoints between images using a brute force approach is very time consuming. Therefore, in order for SIFT to be applicable to a navigation assistant, keypoint generation and comparison must be sped up, and the accuracy of the system must be increased.

B. Speed Enhancements

1) Analysis and Segmentation of Color Distributions

The major problem with SIFT that prevents its usage in applications such as navigation is that it takes too much time for computing descriptors and for comparing these features of images [13]. The authors propose a new Color Segmentation technique to speed up the search process, rather than modifying the SIFT descriptors and sacrificing accuracy in the process.

In order to speed up the search process, each image was characterized by a three-character string, with each character representing the three components of color - red, green, and blue, as a number from 1 to 5, inclusive. Each of the letters was determined by the scaled intensity of the average of the colors at the image's keypoints. For example, all values of red between 0 and 51 would be characterized as 1, and all values between 204 and 255 would be characterized as 5, as displayed in Figure 3, and as calculated by the code snippet in Algorithm 1.

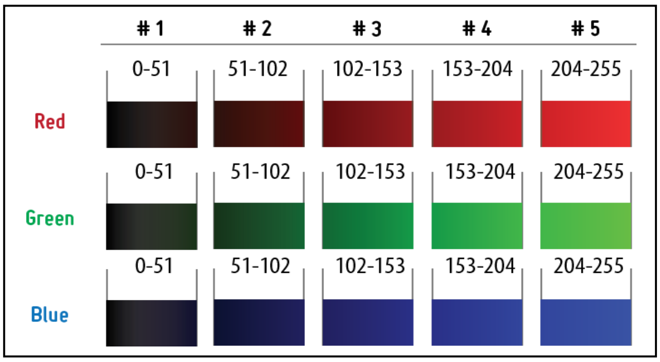


Figure 3. Example bin separation of RGB components.

The RGB components for an image are each placed into a bin, represented by an integer from 1 to 5, the number of bins. An image in the database is only compared to the image frame if at least 2 out of 3 of their RGB components fall in the same bins.

Algorithm 1 AssignColorToImage (*image, imagekeypoints*)
red, green, blue = 0
redRGB = *sampleRedComponentsOfKeypoints()*
greenRGB = *sampleGreenComponentsOfKeypoints()*
blueRGB = *sampleBlueComponentsOfKeypoints()*
for *binNum from 1 to numBins* **do**
if *redRGB _ 255_numBins_binNum* **then**
red = binNum
if *greenRGB _ 255_numBins_binNum* **then**
green = binNum
if *blueRGB _ 255_numBins_binNum* **then**
blue = binNum
end
return red + green + blue

This method of identifying an image by its color provides an additional way to fingerprint images, and additionally, several images with similar RGB component values can be assigned the same color string. This method of "binning" is useful for pruning the database to only test frames on images with similar colors. Although the issue of varying illumination in pictures might initially seem to reduce the accuracy of Color Segmentation, if the system contains a flashlight attached to the camera device, all photos will be taken with the same illumination by dynamically toggling the light based on environmental luminescence, and colors will be similar.

Additionally, our system takes into account that colors may slightly vary across images by comparing images with similar string representations that have two out of the three-color components matching. Therefore, an image with a color representation of 111 would be compared to an image with a color of 115 or a color of 111, but it would not be compared to an image of color 124, as the image with color components 124 would be sufficiently different from that with color components 111, making this comparison unnecessary. By splitting up the images into separate bins, some objects can be pre-eliminated without having to check the descriptors. Even if all images in the database except one fall into the same bin, the Color Segmentation algorithm will outperform the SIFT algorithm without enhancements. In the worst case scenario where every object in the household is the same color, all the images fall into one bin, and the algorithm takes slightly longer than the SIFT algorithm due to the time taken for calculating colors. However, as this worst case scenario is very unlikely, the SIFT speed enhancements discussed increase computational speed an overwhelming majority of the time.

2) *Difference of Images Algorithm*

A Difference of Images technique is proposed by the authors in order to further reduce computational time and to make SIFT more suitable for real-time usage. Due to the number of frames analyzed per second, superfluous data such as the environment external to an object and previously detected objects are captured for most subsequent frames. As a result, images can be contrasted against a previously taken image in order to remove redundant keypoints. Effectively, the improved SIFT algorithm has a "memory" for previously detected objects and excludes the environment from keypoint comparison, thus greatly reducing the number of keypoints that need to be compared, and consequently cutting computational time (Figure 4).

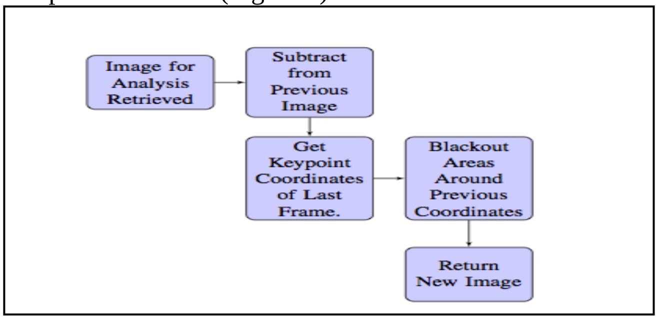
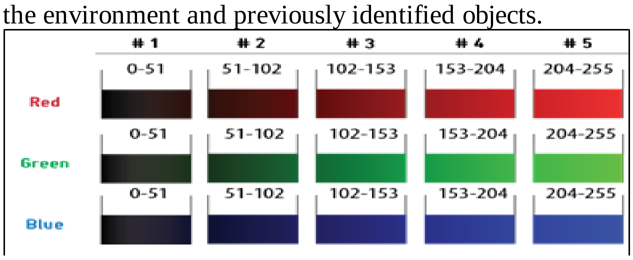


Figure 4. The flow for the Difference of Images algorithm. First, the image is retrieved; next it is subtracted from the previous image. Then, the coordinate of the previous image's keypoints are identified and those coordinates are blacked out in the new image.

The algorithm focuses on identifying new objects by removing re-occurring, irrelevant objects and backgrounds. The removal of re-occurring objects and backgrounds is done by first subtracting a frame from the previous frame by simply subtracting each pixel intensity by its counter-part in the other image, which was taken 0.1 seconds beforehand (Figure 6). During those 0.1 seconds, portions of a new object can be introduced to the system. Every pixel in the two images is subtracted from its counterpart, and the system then removes a window around the coordinates of previously identified objects. Due to this subtraction, objects that have been recognized are blacked out, reducing the number of keypoints that need to be analyzed. Furthermore, the environment is also removed, as it will closely match that of the previous image. Thus, subtracting the images removes the environment and previously identified objects.



Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

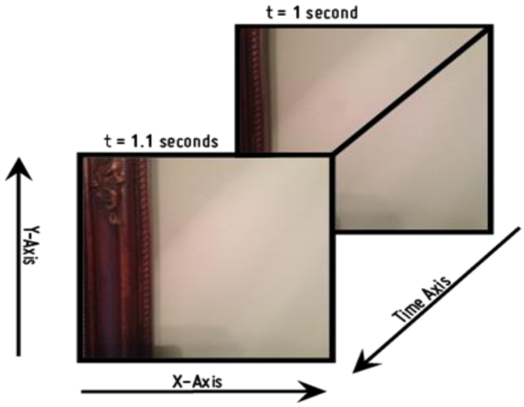


Figure 5. Two frames shown in a 3D space. Here, part of a new object is being introduced. These images will be subtracted, removing the overlapping pixels.

C. Accuracy Enhancements using Contrast Manipulation

To increase the potency of the system in the event of a blurry image, we applied a more robust contrast system to the SIFT algorithm. This enhancement is especially needed in any environment because blurry images have few keypoints due to the lack of defining features [13]. Varying the contrast makes the gradients more pronounced, thus increasing the number of unique keypoints in the blurry image. During the training of the original image database, images of varying contrast are artificially created by modifying the individual pixel intensities at each point and then analyzing the images for SIFT descriptors (1).

$$g(i,j) = \propto \cdot f(i,j) \qquad (1)$$

As a result, varying keypoints to account for the potential illuminations are introduced to the database. As later shown in section 4.2, doubling the size of our system does not affect the computational runtime as many of these new keypoints are excluded by our Difference of Images technique and Color Segmentation algorithm. Furthermore, by artificially creating contrast images and calculating unique features by the SIFT algorithm for each of these contrast images; the original image is represented much more thoroughly than before. In summary, the speed and accuracy enhancements make the assistant feasible for real-time object detection and lead to a higher rate of correct object identification. Specifically, Color Segmentation and our Difference of Images algorithm allow the system to be used in real time while the Contrast Manipulation technique increases the accuracy of the system.

III. DATA ANALYSIS AND RESULTS

A. Basic Navigation Assistant

The runtime and accuracy of the assistant were tested on 45 pictures of randomly selected objects that were found in a household, for both the assistant that contained enhancements and the version without enhancements. Both versions were tested on an iPhone with an Apple A5 chipset, a Dual-Core 1 GHz ARM Cortex-A9 processor, and an 8-megapixel camera. On average, the assistant without enhancements took 3.427 seconds to identify an object and had an accuracy rate of 45.555% (Figure 6). The runtimes of these objects ranged from 6.392 seconds to 1.230 seconds, depending on the number of keypoints of that object. For objects like the plate, which had few decorations, the runtime was low because fewer keypoints were found. On the other hand, family picture number three was a collage of detailed pictures, making it saturated with keypoints. After completing the search, if the assistant could not find a matching object, it informed us that nothing was found.

Accuracy and Computational Time for Navigation Assistant without Enhancements

Thumbnail	Name	Time Taken (s)	Thumbnail	Name	Time Taken (s)	Thumbnail	Name	Time Taken (s)
	Front Door	3.486214 ✓		Family Picture (3)	6.392114 ✗		Outdoor Vase	2.140284 ✗
	Family Picture (1)	3.892216 ✓		Thermostat	3.826599 ✓		Plate	1.482024 ✗
	Family Picture (2)	3.824261 ✓		Outdoors Lamp	3.930124 ✗		Backpack	3.023484 ✓
	Outdoors Fan	5.227483 ✓		Decorational Duck	2.220197 ✗		Book	2.824392 ✗
	Porch Light	3.211315 ✗		Painting (1)	3.953266 ✓		Security Alarm	4.292393 ✗
	PS3 Controller	2.912413 ✓		Wooden Painting	2.039234 ✓		Family Picture (4)	5.029842 ✗
	Energy Guide	3.026638 ✓		Box Lid	1.230223 ✓		Painting (3)	2.028432 ✓
	Laptop Computer	2.562912 ✓		Painting (2)	3.420869 ✓		✓ = found	✗ = not found

Figure 6. For the sake of brevity, only 23 objects are shown here. Out of the 45 objects, only 14 were found when testing the assistant without enhancements. Overall, the assistant was only 45.555% accurate and took an average of 3.427 seconds per image.

B. Navigation Assistant with Enhancements

Both the accuracy and runtime improved significantly for the assistant with enhancements due to Color Segmentation, Contrast Manipulation, and Difference of Images; the algorithm was 66% more accurate and 47% faster than the basic assistant without enhancements (Figure 7). Additionally, the runtimes for all of the images on the enhanced assistant were lower than their counterparts on the assistant without enhancements. Although increasing accuracy usually compromises computation time, the enhanced assistant improved upon both accuracy and runtime [15].

Accuracy and Computational Time for Navigation Assistant with Enhancements

Thumbnail	Name	Time Taken (s)	Thumbnail	Name	Time Taken (s)	Thumbnail	Name	Time Taken (s)
	Front Door	1.053644 ✓		Family Picture (3)	4.463912 ✓		Outdoor Vase	1.461951 ✓
	Family Picture (1)	2.176716 ✗		Thermostat	2.076716 ✓		Plate	0.792973 ✓
	Family Picture (2)	2.289721 ✓		Outdoors Lamp	2.838547 ✗		Backpack	2.075427 ✗
	Outdoors Fan	3.551415 ✓		Decorational Duck	1.480018 ✓		Book	1.254572 ✓
	Porch Light	1.406035 ✗		Painting (1)	2.741376 ✓		Security Alarm	1.518360 ✓
	PS3 Controller	1.160310 ✓		Wooden Painting	1.443829 ✓		Family Picture (4)	3.061600 ✓
	Energy Guide	2.045978 ✓		Box Lid	0.912032 ✗		Painting (3)	1.255703 ✓
	Laptop Computer	1.090909 ✓		Painting (2)	2.505122 ✓		✓ = found	✗ = not found

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Figure 7. This figure is created in a similar manner to the previous image to present the data for the navigation assistant. For the sake of brevity, only 23 objects are shown here. The navigation aide with enhancements had an accuracy of 75.555% and an average time of 1.820 seconds per image, making it 66% more accurate and 47% quicker than the basic guidance assistant without enhancements.

IV. DISCUSSION

A. *Mismatch Analysis*

In order to determine the pitfalls of using SIFT without enhancements for a navigation assistant, we analyzed each of the undetected objects in detail. Many of these objects in the basic guidance system did not have enough unique keypoints, so they resulted in incorrect matches, as was the case with the porch light, the third family picture, the outdoor lamp, the outdoor vase, the plate, the fourth family picture, and the alarm. With respect to the decorative duck, the gradients were not distinguished enough, so there were less keypoints, which led to false positives. Finally, the book was simply undetected because the picture was taken out of focus, and therefore, fewer keypoints were found.

On the other hand, the enhanced guidance assistant was able to detect far more of the objects, including the blurry book, due to the improvements that the authors have proposed. The few objects that the enhanced guidance system did not detect either did not have enough keypoints, or their keypoints mismatched against other objects due to their similarity (Figure 8).

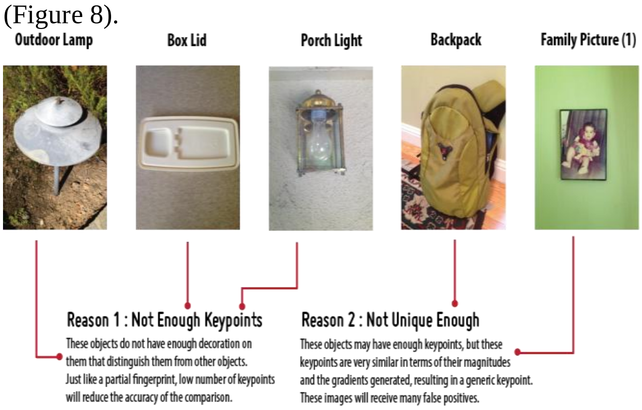


Figure 8. These objects were not found because either they did not have enough keypoints and thus they were indistinguishable from other objects, or that the keypoints were not unique enough (only five shown for brevity).

The following table (Table 1) summarizes the different reasons why some of the objects were identified and others were not with the enhancements.

Name	# Descriptors	Reason
Security Alarm	72	The Contrast Manipulation technique allowed the descriptors to be found in the Assistant with enhancements. In the system without enhancements, the descriptors found were similar to the other descriptors found in other objects.
Box Lid	55	The box lid was not matched in the Assistant with enhancements, as the color was misrepresented due to illumination changes.
Plate	22	The plate was successfully identified in the Assistant with enhancements but not in the Assistant without enhancements as the number of other keypoints from other objects dramatically decreased due to our Color Segmentation algorithm.
Family Picture 1	306	Illumination change caused the Family Picture 1 to be misclassified in the Assistant with enhancements.

Table 1. Discussion on why some of the household objects were found/not found with the Assistant with Enhancements.

B. *Comparison against RFID system*

Solutions proposed by some researchers suggest using RFID tags to allow the visually impaired to scan their surrounding world. However, this approach requires large amounts of user input as the user has to scan each object in order for the RFID tags to be effective. As a blind person, scanning these objects will be quite difficult as the three dimensional position of these objects is unknown. On the other hand, the system proposed by the authors significantly reduces the amount of user input required and is easy to use. Furthermore, the RFID system requires modification of the environment, a requisite not needed by the CBIR system proposed by the authors. Thus, the system proposed has a clear advantage over the RFID approach.

C. *Comparison against GPS and Ultrasonic System*

The DRISHTI system proposed by Helal et al. [12] focuses on using GPS and Ultrasonic to identify potential path finding routes. However, the GPS system does not give the user environmental context if an object is moved or rotated. On the other hand, the robustness of our proposed aide is especially applicable and can be used to identify even smaller objects such as notebooks, a task that is not possible with the DRISHTI system. A change of objects within the known environment will break the DRISHTI system and require a re-evaluation of the environment. Therefore, the system proposed by the authors is seen to be much more robust and applicable to many more scenarios than the DRISHTI system.

V. CONCLUSION

We have developed an assistant that enables the navigation of a known environment for the visually impaired. Most systems developed to-date do not take into account objects that enter the field of vision for the user [9, 13]. Our assistant emphasizes the increase in environmental context provided to the users. One of the most important findings of this work is the increase in speed proposed for the SIFT algorithm in real-time applications. By using Color Segmentation and the Difference of Images technique, the authors have made SIFT suitable for real-time usage. The control Navigation Assistant, developed without the authors' enhancements, had an object identification accuracy of 45.555% and an average runtime of 3.427 seconds. The experimental Navigation Assistant, which included the techniques of Color Segmentation, Difference of Images, and Contrast Manipulation created by the authors, had an accuracy of 75.555% and an average runtime of 1.820 seconds. Compared to the RFID, GPS, and Ultrasonic systems, our assistant provides more context while maintaining a real-time computational speed. The navigation assistant implemented is also easily extensible to a much larger system. Such systems can include path finding applications and area geo-location for blind assistance. Our results suggest that our assistant is accurate, efficient, and easy to use.

ACKNOWLEDGMENT

The authors would like to thank Mr. Diwakar Tundulum of Nvidia Corporation for his guidance and technical assistance in the research.

REFERENCES

1. "Statistical Facts about Blindness in the United States (2011)", Internet: https://nfb.org/factsaboutblindnessintheus, 2011 [Sept. 09, 2013].
2. Frick, KD., Gower, EW., Kempen, JH., et al.: "Economic Impact of Visual Impairment and Blindness in the United States," *Archives of Ophthalmology*, Vol. , no. 125.4, pp. 544-50, 2007.
3. "Facts and Figures about Issues around Sight Loss", Internet: https://www.actionforblindpeople.org.uk/about-us/media-centre/facts-and-figures-about-issues-around-sight-loss, 2008 [Sept. 09, 2013].
4. "Free White Cane Program", Internet: https://nfb.org/free-cane-program [Sept. 09, 2013].
5. Lowe, DG.: "Distinctive Image Features from Scale-Invariant Keypoints" *Int. J. Comput. Vision*, vol. 60.2, pp. 91–110, 2004.
6. Young, S.: "MIT Technology Review: What it's like to see again with an artificial retina", Internet: http://www.technologyreview.com/news/514081/can-artificial-retinas-restore-natural-sight/, 2013 [Sept. 09, 2013].
7. Browne, D.: "Towards a Mobility Aid for the Blind" *Image Vision Comput. Nz*, pp. 275–79, 2003.
8. "Location Awareness Programming Guide", Internet: https://developer.apple.com/library/ios/documentation/userexperience/conceptual/LocationAwarenessPG/CoreLocation/CoreLocation.html, 2012 [Sept. 09, 2013].
9. McDaniel, T., Kahol, K., Villanueva, D., et al.: "Integration of RFID and computer vision for remote object perception for individuals who are blind" in *Proceedings of the 2008 Ambi-Sys workshop on Haptic user interfaces in ambient media systems* (HAS '08). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), ICST, Brussels, Belgium, Belgium, Article 7.
10. Kain, A., Macon, MW.: "Spectral voice conversion for text to-speech synthesis" In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, Vol.1, pp. 285-288, May 1998.
11. Van Kleek, M.: "Evaluating the Stability of SIFT Keypoints across Cameras", tech. rep., Agent-based Intelligent Reactive Environments MIT CSAIL.
12. Ran, L., Helal, S., Moore, S.: "Drishti: An Integrated Indoor/Outdoor Blind Navigation System and Service", In *Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications* (PerCom.04), pp. 23-30, 2004.
13. Adrien Auclair, LC., Vincent, N.: "How to use sift vectors to analyze an image with database templates" tech. rep., University Paris-Descartes, vol. 4918, pp. 224-236, 2008
14. Bakken, T.: "An Evaluation of the SIFT Algorithm for CBIR", tech. rep., Telenor R&I Research Note, 16 Aug. 2007.
15. [Gabsi, N., Clerot, F., Hebrail, G.: "Efficient Trade-Off between Speed Processing and Accuracy in Summarizing Data Streams", In *Proceedings of the 14th Pacific-Asia conference on Advances in Knowledge Discovery and Data Mining - Volume Part I*, pp. 342-53, 2010.