2C Vision Game: Visual Acuity Self-Testing Using Mobile Devices

Dina Najeeb
California State University, Northridge
Los Angeles, CA, USA
dina.najeeb.903@my.csun.edu

Ani Nahapetian
California State University, Northridge
Los Angeles, CA, USA
ani@csun.edu

Abstract

This paper explores non-intrusive, engaging, and cost effective mobile health methods to detect and monitor changes in visual acuity caused by diabetic retinopathy (linked to diabetes) and macular degeneration (linked to aging). These prevalent conditions can both lead to blindness if undetected early in their gradual progression. By leveraging mobile computing platforms and sensor technology, a serious game for use by patients at risk for these debilitating conditions is provided to monitor and to alert of any changes in vision in advance of clinical visits. The interface engages users in reading street signs within view, while the software determines the accuracy of the readings and the distance of the signs from the user, and then uses that information to track any changes in the persons vision over time.

Categories and Subject Descriptors

C.5.3 [**Microcomputers**]: Portable Devices—*mobile technology*; D.2.3 [**Coding Tools and Techniques**]: Object-oriented programming—*mobile computing*

General Terms

Design, Measurement

Keywords

Acuity testing, visual acuity, optical character recognition (OCR), distance calculation, sensor fusion

1 INTRODUCTION

Technological advances in sensor technology and mobile computing have paved the way for affordable self-health monitoring tools. Here, we leverage advances in software and sensor research, coupled with technology already experiencing pervasive use in the lives of patients, to delay the onset of vision impairment with its drastic consequences.

This work addresses monitoring visual degradation in patient groups at risk for retinal complications and their progression to blindness, by monitoring visual acuity using an interactive serious game that engages users in reading street signs.

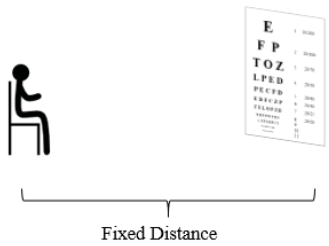
By determining the accuracy of the users registration of street signs and the movement/distance of the user from the signage, any changes in vision in the user are tracked overtime and alerted. The system, built for mobile devices, leverages software approaches and frameworks, including image processing, sensor fusion, localization, optical character recognition (OCR), and speech to text recognition.

The 2C Vision Game application adapts the traditional clinical vision testing techniques, with users reading existing street signs within their vicinity. The game allows mobile users to self-monitor their acuity level by collecting points for each accurately read text from street signs within a certain distance range. The farther the user is from the sign, the higher the score, reflecting a better vision. Our 2C system detects changes in vision prior to scheduled yearly checkups and alerts users to seek clinical treatment, preventing delay, further degradation in vision, and increasing the potential treatment options. The approach is inherently cost effective, requiring software and mobile technology only; it is non-invasive, lacking the need for pupil dilation or any other such procedures; it can occur outside of a health delivery environment, which increases its impact in underserved communities and in impoverished areas around the world; and finally, the approach is fun and encourages continual, non-burdensome testing of ones vision, leading to earlier detection of symptoms and hence earlier treatment. The growing epidemic of diabetes is the leading cause of blindness in the United States, in most part due to the onset of diabetic retinopathy as a result of blockages in retinal blood vessels due to high blood sugar levels. Macular degeneration is one of the leading causes of blindness in older adults, resulting in loss of central vision due to damage of the retina. The target population includes diabetic patients who are at risk for diabetic retinopathy, degradation of retina and eyesight due to blood sugar modulation. Specifically, in this population we consider children and young adults, who we hope to engage with game interfaces and mobile technology. Additionally, we consider patients who are at risk for blindness due macular degeneration, a condition linked to aging. Both target diseases, diabetic retinopathy and macular degeneration,

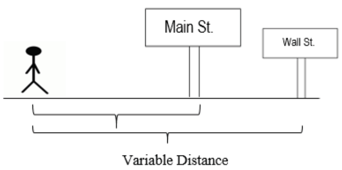
BODYNETS 2014, September 29-October 01, London, Great Britain

DOI 10.4108/icst.bodynets.2014.258235

are slow progressing conditions that are monitored annually via ophthalmologic exams carried out by physician. The exams include intraoptic visualization of the retina and the optic nerve for assessing for changes in the cellular matrix of the retina. A second component of the exam includes an eye exam using the Snellen chart, the well-known eye chart starting with the letter E, for assessing visual acuity during the physician visit. As modeled in Figure 1 (a), this test is typically performed by placing the patient at a fixed distance from the eye chart and covering one eye at a time, while the patient reads the letters out load. The user-system interaction involves the user being prompted to read signs, including business signs and street signs, and then voice what is seen. The expected usage includes outside on the street, while sitting in a car, and indoors, Figure 1 (b) summarizes the interaction model. On the systems end, the distance between the user and the sign is calculated, the text of the signs is determined through image processing, and the accuracy of the reading is determined. Adjustments are made for the use of any prescription lenses. These calculations over multiple readings are combined to determine the user current visual acuity. Any changes in acuity over time are used to alert the user to visit a physician, and hopefully speedup the discovery of any retinal degradation, while treatment is still an option. The key challenges addressed include: 1) the diversity of fonts, spacing, lighting conditions, elevations, angles of view, and distances of the signs used in the vision monitoring game for correctly and cost-effectively carrying out optical character recognition (OCR); 2) limitations, in terms of sensor availability, precision, and accuracy, of sensor technology on mobile platforms for the prescribed purpose; 3) user interface challenges to address user engagement and frequency of participation, specifically from our target patient populations; and 4) data fusion of diverse sensor information in noisy (in terms of signal noise) environments for accurate and sufficiently precise vision degradation capture.



(a) Typical Clinical Snellen Test



(b) 2C Acuity Testing

Figure 1: Comparison of Current Vision Test with 2C Vision Assessment

2 RELATED WORK

Research in body area networks has focused on various important physical health activities, including sleep monitoring [1], food consumption [14], muscular system behavior [2][15][16], vital signs [13], and cardiovascular health [3][17]. Additionally, monitoring exposure to detrimental environment factors, such as ultraviolet radiation, has been addressed with body area networks [4]. Monitoring vision, which we explore with this work, however, has received less attention. Current mHealth solutions focused on the eye include the commercial Welch-Allyn iPhone Adapter [5] and the research eyeMitra [6] from the MIT Media Lab. Both systems focus on retinal inspection and require pupil dilation, limiting the scope of use of the systems to healthcare professionals, albeit in the field There are a few existing mobile system applications dedicated to measuring nearsightedness and farsightedness on the markets. The apps typically have the user hold the smart phone away at a fixed distance, while the symbols on screen periodically change in size. The best ranked Visual Acuity Test [7] Android application by Advent Mobile Designs, for example, has users conduct a self Snellen chart test from a fixed distance. The approach is centered on displaying the Landolt C, a standardized symbol for vision testing, on the devices screen from a stable distance. Then the patient has to select the exact direction of the symbols opening each time that Landolt C symbol changes in size. The application is difficult cumbersome to use, however, as it requires the user to enter the direction of the symbol opening on the tablet itself manually while sitting or standing away from the device at a fixed distance. Getting closer to the device gives the user a chance to peek at the figure, potentially invalidating the self-evaluation process.

3 OVERVIEW

2C Vision Game provides an enjoyable and engaging approach to evaluate visual acuity, with better demonstrated eyesight resulting in higher game scores. Key to the assessment and monitoring are the text viewable by the user and the texts distance from the user. The overview 2C visual acuity evaluation is provided in Figure 2 and detailed in the remainder of this section. First, we review the procedure

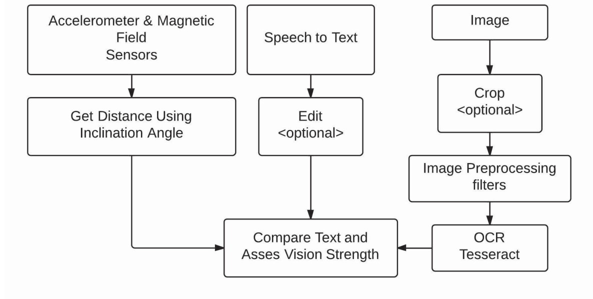


Figure 2: 2C Vision Game High Level Overview

used to identify the text displayed on the street signs. Then, the protocol for computing the distance from the patients to the sign in focus is presented. Afterwards, we describe how 2C assesses vision strength and its user interface.

3.1 TEXT RECOGNITION

To extract the text displayed on the sign viewable by the user, we use an optical character recognition (OCR) tool that accepts an image as its input and outputs the characters recognized in the image. Tesseract / Liptonica OCR library, an open source library provided by Google, was selected as it can be integrated internally within application without the need to send the image to a remote server for processing. Additionally, it is viewed as one of the most accurate and widely used OCR engines available on the market [9]. Since the OCR library requires an image, the user snaps a picture of the sign where the intended text is displayed and saves it in the device external storage, the secure digital card SD Card. To improve OCR accuracy, the captured images are preprocessed by filtering to reduce color noise using the OpenCV4Android library, a fast and easy to use open source computer vision library for real time image processing. Additionally, the images are grayscaled and binary colored, again to improve the OCR results. The OpenCV features applied on the image are outlined in the following pseudocode:

```
Image _preprocessing _filter (image)
1. Convert colored image to greyscale
2. Convert greyscale image to binary colored
   (black and white) image
3. Carryout noise reduction
4. Return processed image
```

Before filtering the image, we need to convert the original format from bitmap to Mat object because the OpenCV functions implemented only accept the Mat format. Mat is a special class that holds two kind of information: a matrix header containing image data and a matrix of the images pixels values. Since OpenCVs binary threshold function only takes greyscale images as an input, it is essential to convert the original colored photo, with an example shown in Figure 3 (a), to a greyscale image, with an example shown in Figure 3 (b) before applying the binary filter. The threshold function results in a binary colored image, black and white, with an example shown in Figure 3 (c). To improve the final product, we smooth the slightly pixelated picture by passing it through another filtering layer called the Gaussian smoothing filter. This function will reduce color fuss; therefore, enhance the quality of text in an image, shown in Figure 3 (d). The result is outputted as a Mat object that can be converted back to bitmap and sent to the OCR engine for text detection.

Before treating the image with the OpenCV filters, the user has the option to either crop the part of the image that contains text. Cropping the image greatly enhances the quality of text recognition, which in return rises the accuracy of 2C acuity testing. Text detection software can also be used to determine the location of the text in the image.

(a) Original Image (b) Original Image After Greyscale Filter

(c) The Greyscale Image After Binary Filter (d) The Binary Image After Gaussian Smoothing Filter

Figure 3: Image Preprocessing Filters

3.2 DISTANCE CALCULATION

Trigonometric rules were applied to calculate the real distance (we will be referring to the real distance later as the value R) from the patients eyes to the sign. To estimate the value R, first we must find two data values: the distance from the device to the base of the sign (will be referred to as the value D) and the complement of vision angle known as $(\theta_{sign})'$. The value D can be estimated by measuring one side and one angle in the triangle formed by the user and the base of the sign. To clarify, we can compute D by using the height of the person holding the mobile device, or simply H, and the value of the inclination angle, or θ_{base}, Figure 4. We use the height of the user for the H value in the calculation.

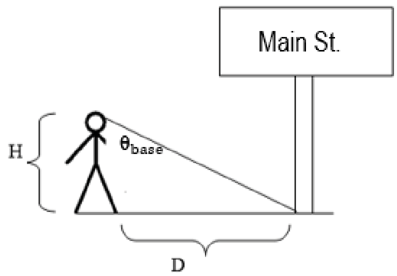


Figure 4: Inclination Angle and Distance to Base

The device inclination angle θ_{base} can be measured by combining data from two motion sensors: the tri-axial accelerometer and the magnetic field sensors. We collect the values retrieved from those two sensors in real time. Angle θ_{base} can be measured by fusing the sensors values to generate the device rotation matrix which reports the orientation of the device, measured by degree units, along the three axis. Fusing the data from the accelerometer and the magnetometer sensors provide us with real world coordination [8]. At that point, we can calculate the distance

D by following the formula:
$D = \tan(\theta_{base}) * H$

After successfully measuring the distance from the person to the base of the sign where text is displayed, we need to calculate the distance R, indicating the length from the person to the sign itself. To do so, another set of calculations need to be performed, as highlighted in Figure 5. The new inclination angle θ_{sign} is used to find the value of the angle $(\theta_{sign})'$. We can easily identify the value of $(\theta_{sign})'$ from the known angles with subtraction from 180^o:

$$(\theta_{sign})' = 180^o - (\theta_{sign})'$$

The value of the side R can be found by applying the sine law of a right triangle:

$$R = \frac{D}{\sin(\theta_{sign})'}$$

This value represents the true or the real distance measured from the person to the text. The real distance is significant because it is an important part in the vision acuity assessment approach.

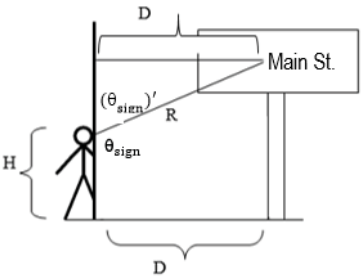


Figure 5: Inclination Angle and the Real Distance to the Sign

3.3 VISION ACUITY ESTIMATION

We can estimate users vision strength based on the same techniques used for Snellen scale test which depends on how far a person can accurately read a text from. The original Snellen test is conducted in a clinical environment where the user must sit in a fixed position at a certain distance from the chart [10]. Then the physician or the examiner asks the patient to read the letters displayed in each row. The size of letters starts large and get smaller as the patient read down the chart. In 2C vision game, the user will be most likely reading the text out of street signs from a certain distance. The further away the text is from the user accompanied with the correct user entry, the higher points will be collected reflecting users vision strength.

We distinguish the user vision sight based on how many points they collect in each stage. The points will be awarded upon accurate text matching within a certain range. Low game score mirror poor eyesight and vice versa high game score means stronger vision.

The point system is as following:

- 5 points, if text is within 0 - 14 ft. from user
- 10 points, if text is within 15 25 ft. from user
- 15 points, if text is within 26 35 ft. from user
- 20 points, if text is within 36 45 ft. from user
- 25 points, if text is within 46 55 ft. from user
- 30 points, if text is within 56 75 ft. from user

In addition, there is a complementary visual scoring system in form of stars. It will be displayed in the upper right corner of the user interface for every 10 points collected by the user. These scoring information will be saved over time to alert the user of any worrisome changes in vision acuity strength.

3.4 USER INTERFACE DESIGN

The user interface design for 2C vision game consists of five simple stages:

1. Snap: capture the street sign.
2. Speech to text: capture user entry
3. Edit speech to text: optional, the user can manually fix the spoken text.
4. Crop: optional, the user can select the area of the image where the sign is displayed.
5. Aim at base: calculate the real distance to the sign and give the appropriate score according to the previously explained scoring system, Section 3.3.

Since the application is targeting patients with variant acuity levels, we have striven to keep its components simple and clear. The buttons are large in size and the text is easy to read. Moreover, the main flow of the application is self-explanatory, Figure 6, and it does not require any training or learning time. The core display contains the device rear camera view and a SNAP button. The SNAP button grant the user the ability to take an image of a street sign. This image is saved in the devices SD Card or its internal storage to be processed by the OCR engine. When the user finishes capturing the photo, a speech to text dialog box automatically prompts the user to say the text on the sign. The vocal user entry is then saved as a string, with the user afforded the option to either edit their spoken entry manually using the device keyboard or to proceed to the next step without editing. Next, two buttons will display: the Skip and the Crop buttons. If the Skip button is pressed, the program will progress without image size manipulation option and if the Crop button is selected, the Crop activity will appear and the user can manually adjust the cropping box to bind the sign

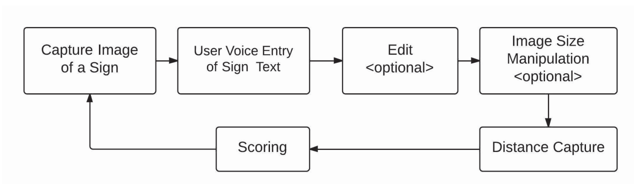


Figure 6: 2C Vision Game UI flow

area of the captured image. When the last phase concludes, the Aim to Base button will appear at the display. The user is asked to direct the devices camera toward the base of the sign and select the button. The score will be given at this point and the number of progress stars will change accordingly. The user continue with the game, as many times as desired.

3.4.1 Crop

Few external factors affect the accuracy of character recognition process. Cropping out the area containing text from the inputted image is one of the high influential factors as it increases the correctness of the text recognition results. Manual cropping is a feasible solution provided as an optional feature, giving the user total control whether to skip image size manipulation step or to proceed with the cropping intent. Nevertheless, we highly recommend image cropping for the reason that this activity rises the certainty of OCR text realization. To clarify, the original image may contain shapes and contour lines that can be interpreted by the OCR engine as characters whereas cropping the photo eliminates the unwanted shapes and lines as illustrated in Figure 7. If the user selects the Crop button from the application user interface, a new screen is displayed. This new activity exhibits the captured image from the Snap stage and a customizable selection box. The user can resize and relocate the bounding box to the part of the image where the sign is displayed. When finished, the handler can select the done button located on the right upper corner of the screen and the cropped image is saved on the existing device storage location.

Henceforward, we can diminish the need of relying on cropping the sign by using shape or text recognition features using the OpenCV framework features.

(a) Image Before Cropping

(b) Image After Cropping

Figure 7: Image of a Stop Sign with and without Cropping

3.4.2 Speech to Text

The speech to text component is a convenient way to capture user entry without relying on the keyboard functionality specially that 2C app users will be mostly patients with diverse vision strength. Android system has a top notch speech to text utility that detect users vocal entry and transform it to text string. In 2C, the speech to text dialog box will be displayed automatically right after the user take the sign picture. After speaking what the user sees written on the sign, the dialog box will close and the app will display the text string in an editable text field where the user have to manually confirm the accuracy of the entry. The text string obtained by this step will be saved in the program to be compared later with the text recognized by the OCR engine. Using the text strings comparison along with the distance read from, we can assess users acuity level.

4 EXPERMENTAL PLATFORM

The system was developed and tested using the Samsung Galaxy Tab 3 tablet, running Android Jelly Bean 4.1. The tablet has a 3 MP rear camera that is used to capture the image by the user. Also, it has accelerometer and magnetic field sensors used to measure the distance from the device to the sign.

4.1 SYSTEM OVERVIEW

The system design reduces the dependency on the human factor in order to have a robust, easy to use, and interactive game. This was achieved using sensor readings directly in the calculations, without the need for manual user confirmation. In the first phase, we retrieve and save the image of the street sign after the Snap button is pressed. During the image capture, the accelerometer and magnetometer sensors are used to measure the inclination angle and calculate the horizontal distance from the person to the base of the street sign. The image of the sign undergoes some preprocessing including, optional size manipulation in the crop feature and the OpenCV image quality enhancing function. Then the Tesseract OCR library is used to extract the text from the image.

The Google Speech-to-Text API is used to retrieve and store the text that the user is able to see, for use in the scoring of the game. In the Aim to Base stage, we find the new inclination angle of the device and we subtract its value from 180 degrees to obtain the complement angle. The complement angle is used along with the horizontal distance measured earlier at the Snap stage to find the true diagonal distance to the sign using trigonometric calculations.

4.2 POWER CONSUMPTION

To measure the performance and power usage of 2C Vision Game, we used a monitoring and analysis tool for Android based applications called LittleEye v2.4.0.0 [11], developed by Little Eye Labs. LittleEye monitors the overall performance of the running app including how much power and memory space the tested app is consuming and it reports the findings in a statistical form. Knowing that energy consumption for any mobile application is highly dependent on the user behavioral interaction with the apps activities [12], it is significant for us to monitor 2C vision game power usage in order to guarantee steady user experience power wise. Figure 8 graphs a run of one cycle of the 2C app game, from the Snap to the Aim at Base stages, with a running time of one minute and 29 seconds. The LittleEye software calculates the expected battery consumption per hour of running the app in the foreground as 9.04% of the device overall battery life.

Analyzing the power consumption by component revealed that most of the power usage is due to the display unit of the application. The CPU consumes only 0.148mAh of power, while the display consumes 2.409mAh. The 2C app received an *A* in the LittleEye performance analysis scoring system. By comparison, a typical activity monitor app, leveraging the accelerometer, receives a score of *B*.

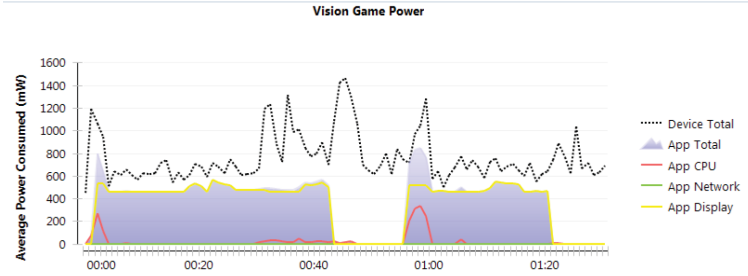


Figure 8: 2C Power Consumption over Time

5 EXPERIMENTAL RESULTS

5.1 Distance Validation

We validated the accuracy of 2C Vision Game distance measurement technique. We first started by measuring the distance to the sign using a regular measure tape and recording the values found for each point. We plugged in the numbers retrieved from the measure tape into the Pythagoras formula:

$$(Diagonaldistancetosign)^2 = (HeightofsignHeightofpersonholdingthedevice)^2 + (HorizontalDistancetothebase)^2$$

We launched the 2C application and recorded the values returned from the same exact standing location and compared the results side by side, Table 1.

Table 1: Distance to Sign Validation

Real Distance to Sign (ft.)	Distance to Sign Found by 2C (ft.)	Error in 2C Estimation (ft.)	$(\theta_{sign})'$ in degrees
5.2	4.17	-1.03	73.57
5.2	6.04	+ 0.84	69.72
7.6	6.49	- 1.11	82.21
7.6	9.33	+1.73	79.71
9.10	7.5	-1.6	82.63
9.10	10.12	+1.02	91.25
11.2	9.46	-1.74	89.19
11.2	11.65	+0.45	90.58
13.4	16.29	+2.89	85.15
13.4	9.45	-3.95	97.74
15	12.36	-2.64	120
15	14.86	-0.14	90.45

The results stated in the table above are very promising as a preliminary study to validate the accuracy of 2C distance calculation method. Most of the readings reported back by the application are within 2 feet of the actual distance. The results show that the $(\theta_{sign})'$, the value of the angle, affects the accuracy of distance approximation. Compare the first and the second readings (5.2 and 7.6), where the error decrease when the complement angle of θ_{sign} is sharper. This information was used to modify the apps user interface by limiting the camera preview screen, so that the user is forced into a sharper angle of the phone to capture the base of the sign.

5.2 OCR Text Recognition Validation

To validate the OCR performance, we used unit validation to ensure the accuracy of the text detection carried out by the OCR libraries. Images of street signs were passed to the recognition engine and the effectiveness of the image preprocessing and filtering was examined. The results, presented in Table 2, give the accuracy of the OCR with and without applying the filtering layer. It is clear that the percentage of characters matched with the preprocessing filter is higher than the ones without filter.

Table 2: OCR Validation Using Cropped Images with and without Preprocessing Filters

Sign Text	Detected Text No Filter	Detect Text w/ Filter	% Match w/ Filter
High st	High sr	High st 1	100
Clydesdale	M	Clgdesdele	80
No Parking	N O Parking	NO Parking	100
Speed Limit	1 Speed Limit ni	Speed Limit	100
Caution	HZEEDITI	Caotion	90
Keep Right	Keep Right	Keep Rignv	77
Broadway	Wadway	Broodway	87
Stop	Stop	Stop	100
Nissan	L ISSA	Nissan	100
Exit	had	Lx1T	50
Office	0 ficE	0 ffice	83

6 CONCLUSION

In this paper, we presented a low-cost, convenient, and engaging approach to deliver a self-conducted visual acuity exam, with the aim of detecting gradual degradation in vision and preventing the onset of blindness. The system has users read street signs at varying distances and determines the accuracy of the readings to give an overall score to the game. The main two components of the app were validated experimentally and were found to be very promising.

In future, we will further develop techniques to enhance text detection using more advanced and automated text cropping mechanisms.

7 REFERENCES

[1] Sheryl LaFleur and Imad Mahgoub. A personal body area network as a pre-screening surrogate to the polysomnography. In Proceedings of the 8th International Conference on Body Area Networks (BodyNets '13).

[2] Rohith Ramachandran, et al. Body sensor networks to evaluate standing balance: interpreting muscular activities based on inertial sensors. In Proceedings of the 2nd International Workshop on Systems and Networking Support for Health Care and Assisted Living Environments (HealthNet '08).

[3] Marco Altini, et al. Personalizing energy expenditure estimation using a cardiorespiratory fitness predicate. In Proceedings of the 7th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth '13).

[4] Xiaoyi Zhang, et al. See UV on your skin: an ultraviolet sensing and visualization system. In Proceedings of the 8th International Conference on Body Area Networks (BodyNets'13).

[5] Welch-Allyn iPhone Adapter. Accessed 30 May 2014. http://www.welchallyn.com/en/microsites/iexamin er.html

[6] eyeMITRA: Mobile Retinal Imaging and Predictive Analytics. Accessed 30 May 2014. http://eyemitra.com/

[7] Visual Acuity Test. Visual Acuity Android Application. Accessed 23 May 2014. https://play.google.com/store/apps/details?id=org.a md.eyesight

[8] Seyed Amir Hoseini-Tabatabaei, et al. A survey on smartphone-based systems for opportunistic user context recognition. ACM Comput. Surv. 45, 3, Article 27 (July 2013).

[9] Anand D. Joshi, et al. OCRdroid: A Framework to Digitize Text Using Mobile Phones". In roceedings of the 1st International Conference on Mobile Computing, Applications, and Services (MobiCASE 09).

[10] Montgomery, Ted. "Visual Acuity." Visual Acuity. Web. 10 Mar. 2014. http://www.tedmontgomery.com/the_eye/acuity.html/

[11] "Little Eye." Performance Analysis and Monitoring Tools for Android Developers. 10 Apr. 2014. http://www.littleeye.co

[12] Jeff S. Brantley, et al. Optimizing battery lifetime-fidelity tradeoffs in BSNs using personal activity profiles. In Proceedings of the 7th International Conference on Body Area Networks (BodyNets '12).

[13] Myung-kyung Suh, et al 2010. An automated vital sign monitoring system for congestive heart failure patients. In Proceedings of the 1st ACM International Health Informatics Symposium (IHI '10).

[14] Ashley Yu, Ani Nahapetian. Participatory Sensing for Fighting Food Deserts. ACM International Conference on Body Area Networks (BodyNets), September 2013.

[15] Sunghoon Ivan Lee, et al. MARHS: Mobility Assessment System with Remote Healthcare Functionality for Movement Disorders. ACM SIGHIT International Health Informatics Symposium (IHI), January 2012.

[16] Armen Babakanian, Ani Nahapetian. Data Fusion for Movement Visualization in a Remote Health Monitoring System. IEEE International Conference on Mobile Computing, Applications, and Services (MobiCASE), October 2011.

[17] Sunghoon Ivan Lee, Charles Ling, Ani Nahapetian, Majid Sarrafzadeh. A Framework for Data Quality Estimation of On-Body Cardiac Sensor Networks. IEEE Consumer Communications and Networking Conference (CCNC), January 2012.