2010 IEEE International Conference on Multisensor Fusion and
Integration for Intelligent Systems
University of Utah, Salt Lake City, UT, USA, Sept. 5-7, 2010

Computing with Uncertainty in a Smart Textile Surface for Object Recognition

Mahsan Rofouei, Wenyao Xu, Majid Sarrafzadeh
University of California, Los Angeles (UCLA)
{mahsan, wenyao, majid}@cs.ucla.edu

***Abstract*— A wearable surface capable of performing object recognition on objects placed on it has many applications in health care such as surgery, assisted living posture monitoring, specifically movement of body parts during sleep and etc. The flexibility and wearability of textile material allows its widespread applications in body-worn contexts. In this work, we propose a portable and wearable smart textile surface which is capable of performing object recognition on a set of prior known objects. We integrate data from multiple sensors to gain knowledge about objects in the environment.**

The uncertainty present in such systems can lead to inaccurate interpretation of the data which is crucial in various medical applications. The most significant part of this uncertainty is due to effects of multiple sensors on each other. We look at different sources of uncertainties in such systems and formulate them. We modify vision algorithm to account for these uncertainties and in the end we present precision bounds for the accuracy of the system.

I. Introduction

In this paper we present the Smart Textile Surface: a surface capable of performing object recognition and localization on objects placed on it. This surface is made from E-textile material which gives the touch and feel of fabric. This feature enables the widespread use of this surface in wearable contexts as a non-intrusive device. Object localization is possible without the use of RFID tags placed on objects. Pressure map based object recognition, is a complementary way of image processing based recognition. However, using cameras and images for performing object recognition is not used here in order to limit the system's knowledge to specific objects. Cameras will provide unlimited knowledge about the environment under test, whereas the Smart Textile Surface limits the system's knowledge to specific objects that are introduced to the system, not violating privacy. Therefore the level of privacy can be controlled in such system based on application needs. Occluded object recognition is a challenge in image processing; however our approach can obtain information about blocked objects based on weight information.

There are two phases for this system: learning and recognition. In the learning phase each object is placed from every possible stable position on the surface. Features related to each state of the object are then extracted and saved in the database. In the recognition phase objects can be placed in any location and orientation on the surface. The recognition algorithm will perform feature extraction and find the object with same or similar features in the database as a match.

The Smart Textile Surface is composed of an array of pressure fabrics, each of which is a three layer structure where a resistive textile is sandwiched between two conductive layers. Objects placed on this surface will produce variable resistances at different elements of this sensor array, from which information about object position, weight and shape can be inferred.

A smart surface capable of performing real-time object recognition has several applications in health care: One example is in a "bed sleep sensor". Mobility measurement during sleep is an important consideration in assessing subject's health and quality of sleep [1]. For example, body movements can be indicators of physical and mental health [13]. This information can also be used to monitor patients during long term illnesses. Our proposed Smart Textile Surface can be used as bed sheets to offer such real-time measurements and assessments. Smart Textile Surface, being made of fabric, will have no different feeling compared to normal sheets, but will be capable of performing a vast range of analysis. Each human body part can be considered an independent object (e.g. head, hands, arms, legs, etc.). From this view, the location and orientation of each object corresponding to each human body part can be monitored during subject's sleep. This information can enable body movement identification [8], posture tracking [5], sleeping pattern identification [4], sleep quality assessment [7] and monitoring of specific body parts, e.g. Head movement caused by respiration to classify respiration status. Currently there are application specific products targeting a human body part. For example in [1] they use a sensor pillow system to monitor respiration. An additional benefit of such system would be its fabric structure which allows non-intrusive measurement.

Another application of such a system is a surgery tray. Millions of patients undergo surgical procedures, but there are cases where surgical instruments are left in patient's bodies causing disorders. The Agency for Healthcare Research and Quality (AHRQ) reports the number of surgical instruments such as forceps, scalpels, dilators and etc., left in a patient's body after a surgery, 1 in every 7000 surgeries [2]. In order to prevent this, instruments are typically counted by nurses before and after an operation prior to incision closure. Even so AHRQ also reports that this procedure is not often accurate due to staff fatigue and interruptions such as changing teams.

With the use of our smart textile surface in surgery trays an accurate count of objects at all times can be maintained using shape and weight properties of surgery instruments. In addition to an accurate count, the type of the missing object can also be identified using object recognition algorithms described in section III.

In the US near 1/3 of seniors above 65 will experience falls and 9500 of them will die as a result [11]. Therefore a good application of the Smart Textile Surface can be in smart floors, used for monitoring elderly living alone. The floor can be covered with the textile surface which allows identifying the location and status of the subject (walking, standing, not moving, falls, etc.) at any time without being intrusive. A similar product to this is the Smart Carpet [12].

As discussed earlier, the flexibility and wearability of the textile makes this a good platform to be used in wearable systems. This surface can be placed on an individual's hand or used as a glove to perform object recognition possibly for the blind. Algorithms designed for this system should be designed differently to account for the orientation and deformation of the surface. There exists a diverse range of applications that can leverage this platform. Examples are Human Computer Interface and Security.

There are several related platforms for tabletop interfaces. Most of these methods require an external object to be placed on the object under test, such as tags, visual markers or other markers. Microsoft Surface [9] is a recent tabletop interface that detects objects using markers called *Dotcode* and some vision algorithms. [10] uses load sensing for performing object recognition.

Our focus however is to obtain methods to account for uncertainties and be able to measure system accuracy. Therefore our contribution in this work is twofold: we first introduce and model the errors and uncertainties present in such textile based systems. Based on this modeling we propose an algorithm for thresholding and edge detection that accounts for uncertainties and results in more accurate data. We then present precision bounds on the accuracy of our proposed system.

The rest of the paper is organized as follows. Section 2 describes the sensor structure and the system design. The uncertainty and interference present in this system is described and modeled in the first part of Section 3. It is then tailored to the algorithms proposed for performing object recognition. In section 4 experimental results are depicted and finally conclusions are drawn in section 5 together with a description of future work.

II. System Design

A. Textile Technology and Characteristics

E-textile is a composite yarn made of fibers coated with conductive polymer. The natural structure is loose and inside fibers are air gapped. The initial throughout resistor between the top-bottom surfaces is low. When extra pressure actuates on the surface, the intra fibers will be squeezed together and the throughout resistor becomes smaller. Here the resistor is inversely proportional to the pressure imposed. Given this characteristics, E-textile is normally used in pressure sensor fabrication. Considering its portability, flexibility and affordability, E-textile based wearable design becomes increasingly promising in carry-on healthcare devices.

B. Sensor Structure and Fabrication

An E-Textile based sensor has a three-stacked layer structure. The EEontex material is sandwiched within two conductive pads like the structure shown in Fig. 1.

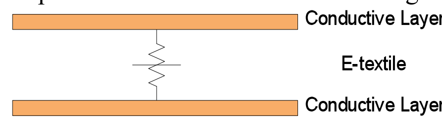

Fig. 1. The basic sensor structure

In this sensor, E-textile acts like a pressure sensitive resistor. When the force is applied, the resistance of E-textile will decrease. The conductive layer can be conductive fabrics, copper foil tape or conductive threads. In order to maintain the flexibility of the sensor and textile feel we use conductive thread and conductive paper for the conductive layer. Fig. 2, shows the use of different conductive material.

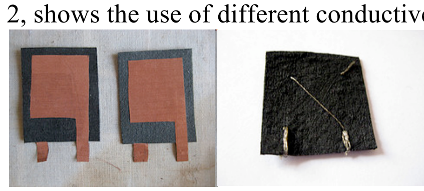

Fig. 2. Sample textile sensors

Here, we also investigate the characteristic of the sensor. For a 10mm by 10mm sensor, Fig. 3 shows the dependence between applied force and resistance.

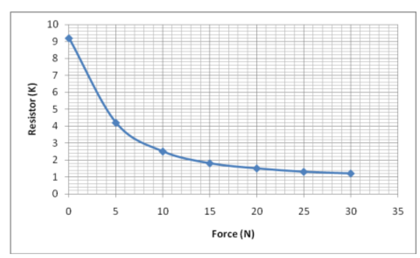

Fig. 3. Electrical resistance over applied load

C. Sensor Array Design

We use the above structure as the basic sensor unit to build the pressure array. Orthogonal zebra patterns could get larger sensing density with less connection pins in sensor array design. In this structure, all the sensors in the same row or column, share the same contact pad, which makes the corresponding scanning circuit more complicated and less power efficient. For simplicity and efficiency, we built a large array as shown in Fig. 4. In this design, each sensor has an independent contact pad. In order to make this design easy for mass production, all the sensors share a sheet of E-textile. This results in a dependency of sensors to each other. This is because a pressure applied on one sensor location might result in a change the resistance of the fabric in other neighboring sensor locations. This neighboring effect will be discussed in detail in the next section.

The current implementation of the system has a separate connection for each sensor location. The routing of such system becomes difficult with larger numbers of sensors. On the other hand, the size of such a system is also limited. For larger numbers of sensors we can use a different design where columns on one row/column share the same wire. In this design methodology, a voltage is applied at the column of interest and the current flowing in each row is measured. The entire array is scanned column by column.

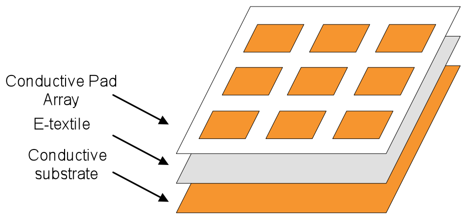

Fig. 4. Pressure Sensor Array

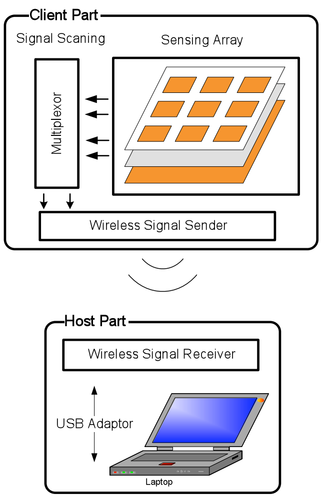

Fig. 5. System Architecture

D. System Architecture

Fig. 5 shows the framework of our proposed system. The system is composed of a client part and a host part. In the client part, the pressure sensor array is scanned with the use of multiplexors. After the sensing data is acquired by the microcontroller, it is packaged and transferred to the host receiver side through wireless RF circuit. The microcontroller we chose is MSP430f2274. The A/D converter's resolution is 10 bits, and the sample rate is 100kps. CC2500 is used for the wireless chip. The communication protocol used to transfer the data is SimpliciTI. SimpliciTI network protocol is a proprietary low-power radio frequency protocol targeting simple, small RF networks [3]. This network protocol can be actually a complement to ZigBee (suitable for larger networks). In the current implementation we have one client and a single host. The client is connected to a surface of a 16x16 array of textile sensors. With the SimpliciTI network protocol the number of end devices can be extended to 100 and each of the surfaces can be much larger, simply by adding multiplexers in front of the ADC converters.

In the host side, the sensed data is processed by a PC when received by the RF chip. A UART to USB convertor (based on MP2010) is designed for interface compatibility.

III. Algorithms in the Presence of Interference and Uncertainty

Each surface reading is a two dimensional array of values between 0-255 representing the pressure map of the objects placed on it. This is very similar to the input of many computer vision algorithms, processing images.

The idea of directly applying existing computer vision algorithms is attractive but there are significant differences between the two that requires major modifications in the algorithm design. The most important difference is sensor uncertainty and error existing in the system. Although errors of such are also apparent in images, however due to different sources and nature of errors the algorithms dealing with them are completely different. Sensor errors may present significance importance in cases where each sensor area is big relatively to the object's area. Therefore, in this section we first derive a model of the errors in the system in part A and then present modified algorithms in parts B and C using this model. One assumption made here is that the textile surface is much larger than the objects placed on it.

A. Uncertainty and Interference

In this section we use an experimental approach to model interference existing in the system. This interference is mainly due to the neighboring effect which is described below. As described in Section II.B, a single sensing layer is shared among all the array sensors, and sensors are within certain spaces to each other. When applying pressure on one sensor, the effect would be inevitably posed on its surrounding sensors on account of mechanical linkage. Here note that the mechanical linkage between the sensors is minimal if they are not adjacent. We define the accumulating pressure sensed on each location due to forces applied not directly to itself but to its surrounding locations, the *neighboring effect*.

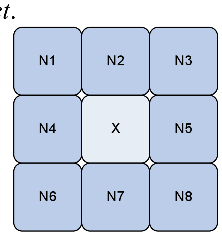

Fig. 6. Neighboring effect

The experimental setup is as follows: a range of weight scales were applied to one sensor location (X in Fig. 6) and all the values read from its 8 neighbors (N_1 to N_8 in Fig. 6) were recorded.

Below in Table 1, these values are shown for weights between 50g – 250g for each of the neighbors. Note that the values shown for X are sensor reading and not weights.

TABLE I
NEIGHBORING EFFECT ON 8 NEIGHBORS FOR WEIGHTS 50G TO 250G

Value read from X	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8
10	1	4	2	6	6	1	5	1
18	1	7	2	8	8	2	7	0
29	1	11	3	10	11	1	12	2
34	1	14	2	12	12	2	14	1
46	3	18	3	15	15	2	18	2

From Table 1, we can see the neighboring effect of locations N_2, N_4, N_5 and N_7 (above, left, right and below X) are similar to each other, and different from locations on the diagonal positions. This is better shown in Fig. 7, where similar locations are grouped together. Values in Fig. 7 correspond to objects with weights in the range of 50g-1kg.

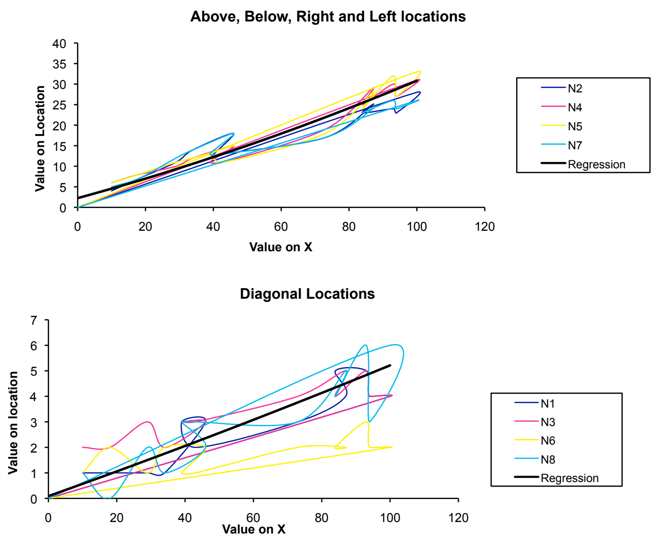

Fig. 7. Regression Analysis

We performed regression analysis on each set of data. This is shown in Fig. 7. Equations 1, 2 show the corresponding relationships together with square R for each set of locations around X.

$$y = 0.0006x^2 + 0.2257x + 2.2443 \quad R^2 = 0.97 \quad (1)$$
$$y = 3E\text{-}05\, x^2 + 0.0478x + 0.0896 \quad R^2 = 0.79 \quad (2)$$

B. Algorithm Overview

Weight and shape are the two main features we use to recognize objects. The weight of the object is assumed to be consistent throughout time. But depending on the 3D shape of objects, they can be placed from various directions on the surface. Therefore the attributes we use to save information about objects are weight and a series of edges that represent objects shape. The system has two phases: a) Learning and b) Recognition. In the learning phase objects are placed from every possible stable position on the surface. The shape of the object is determined through an edge detection algorithm described in part C. This information together with the object weight is stored in the database for that object.

In the recognition phase, various objects are placed on the surface. Our proposed edge detection algorithm determines boundaries of each object together with its weight. This information is searched in the database to find the best match to the measured data. Note that, at this stage we assume non-overlapping objects as inputs of the edge detection algorithm.

C. Computing with Uncertainty

Finding the object's shape is based on edge detection. Without the presence of uncertainty in sensor data, edge detection is a straightforward task. A binary mapping is done from the value of each location based on a threshold value. Locations having a value higher than the threshold are set to 1 to represent object area, while other locations below the threshold are set to 0. The boundary that separates the 0 and 1 locations is identified as object's edges.

However, due to the presence of uncertainty and interference as described in part A, this process is prone to error. Therefore prior to edge detection we perform another mapping that accounts for the neighboring affect. Fig. 8 depicts the mapping process.

$$\left\{\begin{array}{l}0\\ .\\ .\\ 255\end{array}\right. \xrightarrow{\;mapping\;} \left\{\begin{array}{l}0\\ .\\ .\\ 255\end{array}\right. \xrightarrow{\;Binary\ mapping\;} \left\{\begin{array}{l}0\\ 1\end{array}\right.$$

Fig. 8. Stages for edge detection

Fig. 9 shows the proposed algorithm for thresholding and edge detection in the face of interference. Steps 1-3 in this algorithm perform the first mapping in Fig. 8 while step 4 performs the binary mapping in Fig. 8. In summery the algorithm starts at the location with minimum weight. This is because this location has the least effect on its neighbors. Note that our main goal in the binary mapping step is to find correct object area and not an exact pressure value of locations.

1. We first sort the locations based on their weights from small to large. Let w_{mij} be the minimum weight location in the surface. Consider a 3x3 window around it.
2. For each location ij, consider the 8 surrounding neighbors (3 or 5 neighbors for corner and edge locations).
   Calculate the accumulating neighboring effect based on rules in part A for neighbors with larger values than $w_{mij} + \Delta$ (where $\Delta \le sensor\ error$)
   $$\text{Neighboring Effect(Ne)} = \sum_{i=0}^{8} N_i$$
   Make the new value of w_{mij} be $w_{mij}' = w_{mij} - Ne$
   Any negative value will be mapped to zero.
3. Let w_{mij} be the next smallest weight location in the list. Go to 2 until all locations have been processed.
4. Once all locations are processed perform thresholding on all locations.
5. For performing edge detection, for each location, if any of its eight neighbors are 0 that location is marked as an edge.

Fig. 9. Modified thresholding and edge detection algorithm

Note that in each location analysis we only consider its 8 neighbors. Although more distant locations might also have effect on location X's data, we consider their affects recursively onto X's 8 neighbors when processing X. The algorithm above assumes single object detection on the surface. In case of multiple non-overlapping objects, a segmentation phase needs to be also added. In line 2 the reason for choosing $w_{mij}+\Delta$ is to not calculate the effects of locations with similar values on each other. Here, Δ is the sensor error. A simple thresholding method can be to make values higher than 0 become 1.

In our proposed Smart Textile Surface we enable an architecture that allows sensing of objects with weights in different ranges. This can be done using multiple layers of different sensing material with different saturation levels, which we will not describe its details here. However, with the presence of uncertainty, there is a lower bound in being able to discriminate between objects of same shape and area. This lower bound gives a measure of the accuracy of the system.

The weight of an object can be calculated from equation 3:

$$W_x = \sum_i \sum_j a_{ij} \times w_{mij}, \quad a_{ij} = \begin{cases} 0 & \text{if location ij is } 0 \\ 1 & \text{if location ij is } 1 \end{cases} \tag{3}$$

In the above equation w_{mij} is the measured weight of location ij after the algorithm in Fig. 9 is applied and a_{ij} is determined by step 4 of the algorithm (thresholding). However there are multiple sources of uncertainties in the above formula. One source of uncertainty is sensor uncertainty (Δ) which is due to manufacturing and sensor degradation in time. The ADC resolution limitation is another kind of inherent systematic error. For example, the quantization error of 8-bit ADC is 0.39%. Another source of uncertainty is due to interference modeling. As described in part A, regression analysis is used to derive equations 1-2. Regression is not accurate and suffers from error. This error affects weight computation since the w_{mij}s used in equation 3 are computed by using the below formula. Here w_m is the measured weight read from the sensors.

$$w_{mij} = w_m - \sum_{i=0}^{8} N_i \tag{4}$$

And each of the N_is in this equation suffers from regression error. The R squared values derived in part A, show this error. Equations 5 and 6 formulate these uncertainties:

$$w_m = w_{mc} \pm \Delta\%.w_m \tag{5}$$

$$N_i = N_{ic} \pm (1 - R^2)\%.N_i \tag{6}$$

In the above formulas, w_{mc} and N_{ic} are the correct value of w_m and N_i respectively which is unknown to us. Therefore if we combine, equations 3-6 we will have the worst case uncertainty of:

$$\begin{aligned} W_x &= \sum_i \sum_j a_{ij} \times (w_{mij} - \sum_{i=0}^{8} N_i) \\ &= \sum_n w_{mc} + \sum_{8n} N_{ic} + \Delta\% \times \sum_n w_m + (1 - R^2)\% \times \sum_{8n} N_i \\ &= W_{mx} + \Delta\% \times \sum_n w_m + (1 - R^2)\% \times \sum_{8n} N_i \end{aligned} \tag{7}$$

Here W_{mx} is correspondingly the exact weight of object x. Therefore the difference in weights of two different objects of size n should be at least the value of $\Delta\% \times \sum_n w_m + (1 - R^2)\% \times \sum_{8n} N_i$ in order to be able to distinguish them in our system.

IV. Experimental Results

A. Experimental Set-up

The Smart Textile Surface we developed for our experiments is a 16x16 array of 1x1cm sensors with 50mm spacing. The E-textile used is NW170-PI-10E4 from EEONYX [6]. Conductive threads were chosen for the conductive part.

B. Results

We placed several objects of different weights and shapes on the developed Smart Textile Surface. Fig. 10 shows the output of the user interface GUI developed for the edge detection algorithm.

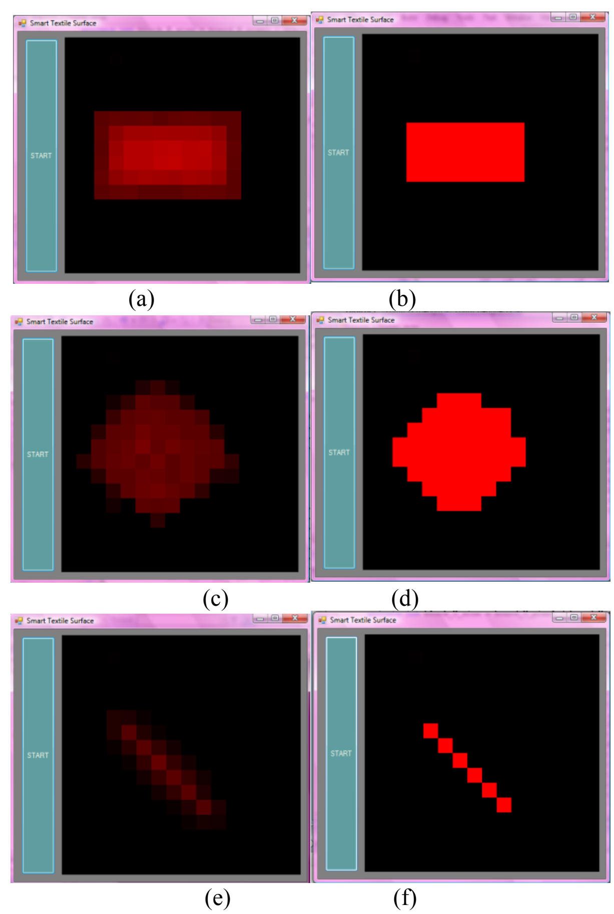

Fig. 10. Object's sensed data and result after thresholding in algorithm. (a) iphone sensed data. (b) iphone data after thresholding. (c) Cup sensed data.

(d) Cup data after thresholding. (e) Marker pen sensed data. (f) Marker pen data after thresholding.

In order to evaluate the effect of our proposed algorithm for eliminating the neighboring effect we present the details of our proposed algorithm, by showing each stage of our proposed algorithm on a wire roll object. Fig. 11.a shows the object under test.

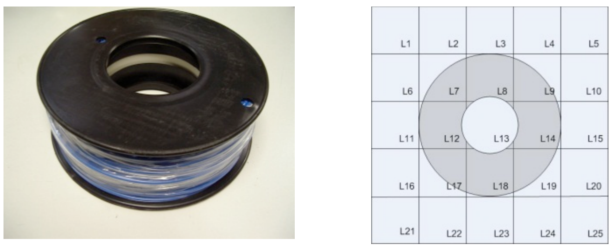

Fig. 11. (a) Object under test (b) Object view from above on textile

Fig. 11.b shows this object from top view and how it is located on the surface. The reason for choosing this object is the middle location on the surface which suffers most from the neighboring effect. Fig. 12.a shows the sensed data obtained from the Smart Textile Surface.

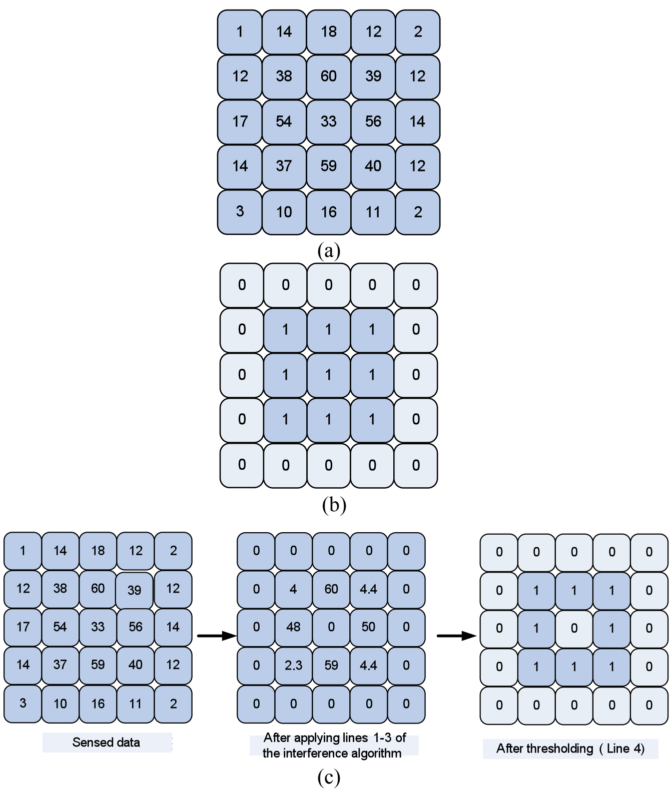

Fig. 12. (a) Sensed data. (b) Result of a trivial thresholding (c) The result and process of applying the algorithm proposed in Fig. 9.

Applying a trivial thresholding and edge detection algorithm on the data in Fig 12.a will result in a wrong result (12.b). A trivial algorithm cannot distinguish any differences between value of locations 13 (L_{13}) and L_7, L_9, L_{17}, L_{19}. But our proposed algorithm applies the Interference rules and from that distinguishes correct area locations. As we see one sensor value has a great effect on the output of the object recognition phase. Please note that although the resulting shape is a hollow square instead of a hollow circle, we are able to detect the circular shape of the object by simply using smaller sensor sizes. The current sensor size and objects are used as a proof of concept.

V. Conclusion and Future Work

In this work we developed a textile-based object recognition platform called the Smart Textile System. We modified a thresholding algorithm for this system to account for uncertainties such as sensor uncertainty, neighboring effect and modeling error. We modeled and formulated the uncertainties present in this system through an experimental approach. We also presented a precision level for system accuracy when comparing objects with the same object area.

Our future work consists of extending this work on developing methods and algorithms to identify multiple overlapping objects. We will design algorithms for both dynamic and batch modes of object addition to our system. We also aim to reduce the amount of interference through system design optimization.

References

[1] Tatsuya Harada, Akiko Sakata, Taketoshi Mori, and Tomomasa Sato. (2000) Sensor pillow system: monitoring respiration and body movement in sleep. Proc. of IEEE IRSO. 1:351-356

[2] "Managing the prevention of retained surgical instruments: What is the value of counting?" by Natalia N. Egorova, Ph.D., Alan Moskowitz, M.D., Annetine Gelijns, Ph.D., and others, in the January 2008 Annals of Surgery 247(1), pp. 13-18.

[3] http://www.ti.com/corp/docs/landing/simpliciTI/index.htm

[4] Mehdi Tafti, "Genetic aspects of normal and disturbed sleep", Proceedings of 6th International Sleep Disorders forum

[5] Eric Bachmann, Robert McGhee, Xiaoping Yun, Michael Zyda "Intertial and magnetic posture tracking for inserting humans into networked virtual environments", Proceedings of the ACM symposium on Virtual reality software and technology.

[6] http://www.eeonyx.com

[7] Robert T. Wilkinson, Ken B. Campbell "Effects of traffic noise on quality of sleep", Journal of Acoust Soc

[8] Sally Ann Ness "Body, Movement and culture: kinesthetic and visual symbolism"

[9] K. W. Bollhoefer, K. Meyer, and R. Witzsche. Microsoft surface und das Natural User Interface (NUI). Technical report, Pixelpark, Feb. 2009.

[10] A. Schmidt, M. Strohbach, K. V. Laerhoven, A. Friday, and H.-W. Gellersen. Context acquisition based on load sensing. In Proc. of UbiComp'02, volume 2498 of LNCS, pages 333–350. Springer, 2002.

[11] http://inqri.blogspot.com/2009/10/elderly-fall-injuries-cost-20-billion.html

[12] http://www.smartcarpet.com/

[13] William Dement, Nathaniel Kleitman, "Cyclic variations in EEG during sleep and their relation to eye movements, body motility and dreaming", Electroencephalography and Clinical Neurophysiology.