

Calibration and statistical techniques for building an interactive screen for learning of alphabets by children

Riby Abraham Bobby¹, Ravi Prakash², Subir Kumar Saha³,
Takafumi Matsumaru⁴, Pratyusha Sharma³ and Siddhartha Jaitly³

Abstract

This article focuses on the implementation details of a portable interactive device called Image-projective Desktop Varnamala Trainer. The device uses a projector to produce a virtual display on a flat surface. For enabling interaction, the information about a user's hand movement is obtained from a single two-dimensional scanning laser range finder in contrast with a camera sensor used in many earlier applications. A generalized calibration process to obtain exact transformation from projected screen coordinate system to sensor coordinate system is proposed in this article and implemented for enabling interaction. This permits production of large interactive displays with minimal cost. Additionally, it makes the entire system portable, that is, display can be produced on any planar surface like floor, tabletop, and so on. The calibration and its performance have been evaluated by varying screen sizes and the number of points used for calibration. The device was successfully calibrated for different screens. A novel learning-based methodology for predicting a user's behaviour was then realized to improve the system's performance. This has been experimentally evaluated, and the overall accuracy of prediction was about 96%. An application was then designed for this set-up to improve the learning of alphabets by the children through an interactive audiovisual feedback system. It uses a game-based methodology to help students learn in a fun way. Currently, it has bilingual (Hindi and English) user interface to enable learning of alphabets and elementary mathematics. A user survey was conducted after demonstrating it to school children. The survey results are very encouraging. Additionally, a study to ascertain the improvement in the learning outcome of the children was done. The results clearly indicate an improvement in the learning outcome of the children who used the device over those who did not.

Keywords

Image-projective Desktop, interactive display, calibration, statistical learning, education

Date received: 28 February 2016; accepted: 18 January 2017

Topic: Special Issue - Robotics in the developing world

Topic Editor: Ayorkor Korsah

Associate Editor: M Bernardine Dias

Introduction

In recent years, development of touch screen is getting high attention. Samsung's display tables known as SUR40,¹ Microsoft's PixelSense² and Surface Table³ are results of this interest. These devices detect fingers, hands and objects placed on the screen using a camera. Though many of the new interactive display systems deliver a high quality

¹ Programme for Autonomous Robotics, IIT Delhi, New Delhi, India.

² Kyushu Institute of Technology, Japan

³ IIT Delhi, India

⁴ Graduate School of IPS, Waseda University, Japan

Corresponding author:

Riby Abraham Bobby, Room 433c, Block 2, Programme for Autonomous Robotics, IIT Delhi, Hauz Khas, New Delhi, 110016, India.

Email: ribyab@gmail.com



Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 3.0 License

(<http://www.creativecommons.org/licenses/by/3.0/>) which permits any use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (<https://us.sagepub.com/en-us/nam/open-access-at-sage>).

of graphics with good interactive experience,⁴ they are expensive and not easy to be transported. These are important requirements for being used in fields like education in developing and underdeveloped countries.⁵

On the other hand, research was going on in order to design a projective screen as an interactive screen. The first instance of desktop projection system was in the study by Wellner,⁶ where a projector and a camera were used. By using suitable sensors, it was possible to provide interaction on a much larger screen. These days, effort is constantly being made in a way such that interaction is possible at a personal level. Handheld projectors⁷ are compact, portable and use laser or light-emitting diode (LED) as light source. SixthSense^{8,9} was developed using this technology. It is a wearable gestural interface that augments the physical world with digital information and enables the use of natural hand gestures to interact. Such devices use camera-based gesture recognition by finger-tracking through coloured marker worn on the finger.¹⁰ In the category of projective display, though head-mounted device was first developed long ago by Sutherland,¹¹ a lot of interest has been generated recently owing to the release of Google Glass.¹² All these devices require a user to wear a device or some kind of tracker. This will impede the natural movement of the human body to some extent, and especially it is not suitable for small children.

We have designed an interactive device using a two-dimensional (2D) scanning laser range finder instead of a camera. As a result, the system is portable and do not require the users to wear any markers. Similar system was used earlier as step-on-interface in which the projected screen was used as bidirectional interface.^{13,14} Through this, information was presented from a mobile robot to the user, and the user instructions were delivered to the robot. A similar device named Image-projective Desktop Arm Trainer (IDAT)^{15–17} was designed for upper limb rehabilitation. Image-projective Desktop Varnamala Trainer (IDVT; Varnamala is a Hindi word for alphabets) is the next generation of IDAT. However, there are many differences that make IDVT distinct from IDAT. To lower the total cost, a less expensive sensor has been used. To compensate for lesser accuracy, a new calibration methodology was introduced to improve the positioning accuracy of the visual feedback system (see ‘Calibration’ section for details). This also makes the system adaptable to different kinds of display systems. Earlier studies revealed that there are significant variations in the way people use a touch screen of a mobile device.¹⁸ The same is more pronounced while using an interactive device like IDVT or IDAT. For the prediction of user’s behaviour, a learning-based method has been proposed (see ‘Learning-based prediction’ section for details). Finally, the target group in both cases are different. The IDAT device, though it was meant for arm trainer, was found to captivate the children because of the gaming part. This was observed while it was demonstrated in many exhibitions like IREX 2013¹⁹ and H.C.R. 2014.²⁰

Therefore, we have adapted the system for educational purpose of children. In short, the IDVT is different from IDAT in the following ways:

- IDVT is used for educational purpose and as an application of interactive screen. For this, a game-based teaching software was developed.
- IDVT uses a less accurate sensor (2D scanning laser range finder) with accuracy of ± 30 mm as against ± 10 mm used earlier. New calibration and learning-based algorithm to accommodate measurement inaccuracies.
- Generalized calibration algorithm which may be used for conversion of ordinary screens to touch-sensitive screens.

Game-based learning has been found to be suitable for learning alphabets and basic arithmetic operations.^{21–23} A brute majority of the implementations of games for learning are targeted at higher and secondary school education.^{24,25} They are not suitable for younger students in primary schools.²⁶ Additionally, many of these are based on desktop environments and do not involve interaction beyond the standard keyboard and mouse. Also there has been no earlier attempt to create such content specifically in Indian languages. It can be noted that there are approximately 310 million native speakers of Hindi language around the world which accounts for 4.7% of the total world population (fourth largest spoken language).²⁷ Nobel peace prize for the year 2014 being awarded to two individuals from south Asia ‘for their struggle against the suppression of children and young people and for the right of all children to education’²⁸ reiterates the fact that a significant effort is required to improve the conditions of children’s education in this region. The proposed device has been accordingly named SAKSHAR, which in Hindi means ‘literacy’. We feel that the devices like SAKSHAR will give a boost to the learning of alphabets by the children and, thus, improve literacy.

The structure of the article is organized as follows. The second and third sections discuss the construction/calibration of the set-up. The fourth section discusses about the methodology for prediction of user’s behaviour. The results of the user survey are discussed in the fifth section. Finally, the conclusions are given in the sixth section.

SAKSHAR:IDVT

In this section, we will give an overview of the SAKSHAR device.

System configuration

The device is shown in Figure 1. It consists of a projector, scanning laser range finder and a computer. The graphics rendered by the computer is projected on a flat surface. The scanning laser range finder, which is mounted on the

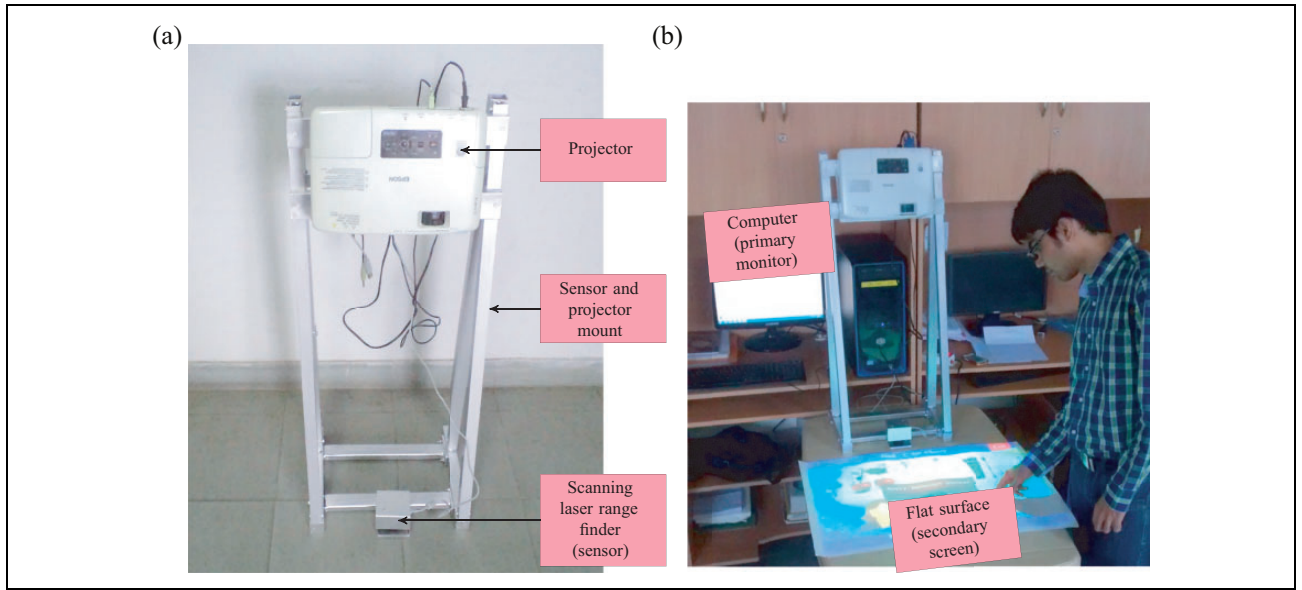


Figure 1. SAKSHAR: IDVT device. (a) System configuration and (b) person playing a game.

device, provides the information about the place which has been touched by the user. The URG-04LX-UG01^{29,30} sensor is a 2D scanning laser range finder (Hokuyo, Osaka, Japan) and has applications in mobile robotics. Detailed specifications of sensor is available in Appendix 1. Since the sensor is small and light weight, it is easily portable. It has detection range of approximately 5 m and makes measurement in polar coordinates. Note that many earlier applications for building big interactive displays used cameras for sensing.^{31,32} Camera is suitable for use in controlled environments, as small changes in illumination demand fine-tuning of the segmentation process. Using two cameras, three-dimensional data can be obtained. But to cover a large area, more number of cameras are required.³³ Other options for sensors are Leap motion³⁴ and Kinect.³⁵ Leap motion, which has the capability of accurate detection (within small volume forming a hemisphere of 1 m radius and field of view of 150°), cannot be used for large displays. Kinect is another device which has a very low resolution and may not distinguish between holding a hand slightly above the surface (say, 1 cm above surface) and touching the surface. The sensor we have used allowed detection over a large area (across approximately 270°, and a radius of 5 m) thus making it suitable for being used in large displays. The display may be 5 m in size according to the range of sensor. If the plane along which the scanning is done is kept parallel to the display surface and close to it, the user's action of touching the display can be easily identified. Note here that the current development has been primarily designed for touch detection and therefore other actions like click, move, push, pull, and so on, are not discussed in this article. However, camera may be suitable for implementing such functionality on a smaller sized screen owing to the limited field of view of the

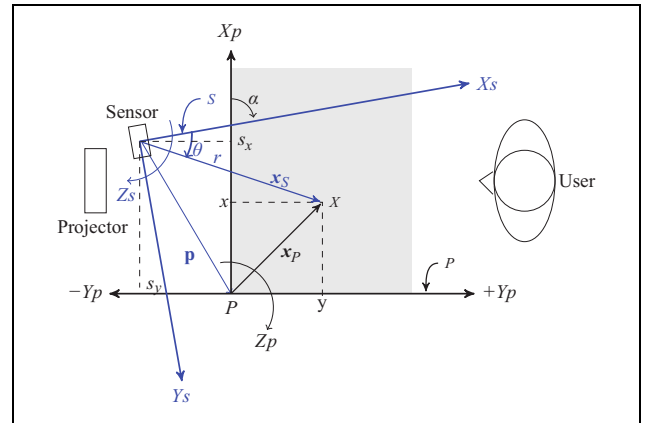


Figure 2. Transformation of coordinate axis.

cameras.^{36–38} The sensor can be easily calibrated by using the method proposed in ‘Methodology’ section. The configuration of the projector with respect to the sensor is measured using two translations and one angular rotation as shown in Figure 2 (details will be discussed in ‘Methodology’ sub section). The projector and sensor are mounted on a stand made of aluminium alloy that ensures strength with light weight. The sensor is placed at the bottom of the stand, almost touching the projection surface. This converts it to a screen plane. On the other hand, projector is placed inclined and focused on the ground plane.

Though we have not found such a portable interactive educational device for comparison, we will compare SAKSHAR with other non-portable interactive displays for large screens. The major difference is that we have used a single scanning 2D laser range scanner sensor instead of a camera.^{36–38} The devices that use camera are difficult to calibrate and highly dependent on illumination and other

image processing-related factors. Issues related to illumination are also prominent while using other sensors-like Kinect as proposed in the study by Sharma et al.³⁹ Additionally, the conventional displays are fragile and thus difficult to transport. The proposed set-up is very cheap (about US\$2000 based on the prices of the components procured in India in April 2016) when compared to any of the commercially available interactive surfaces (e.g. Samsung SUR 40 costs about US\$17,000). The high cost of the commercial devices may be principally due to the price of the LED screen. SMARTBOARD is also expensive (more than US\$3600), and like other display systems it is also not portable. For the proposed SAKSHAR:IDVT, except the sensor (costing US\$1000), other components like computer, projector and mounting stand may be readily available in many schools and educational institutions. In such a situation, the cost of fabrication of the system is only about US\$1000. In other countries or in the future, prices may vary based on the technology, governmental levies, popularity and demand.

The game

The developed application produces output in two windows. One of them is a main window that opens in primary monitor and other one opens in a secondary screen, as depicted in Figure 1(b). The primary monitor is a part of the personal computer (PC) from where the game is invoked. This window has keys/drop-down menus for several configurations including the one to choose the language. The application runs in the main window. The secondary monitor is the flat surface on which projector projects or liquid crystal display (LCD) screen. Game was programmed in C-.05em C# using OpenTK⁴⁰ framework. The secondary window pops up in full-screen mode to generate the game. The game has provision for audio feedback, which was produced using Cgen.Audio,⁴¹ a Microsoft.NET⁴²-based audio library. The laser range finder was connected to the central processing unit through a universal serial bus cable. Serial communication was implemented using UrgCtrl library.⁴³ The details about the game are discussed in Appendix 2.

Calibration

Calibration of the laser range finder (URG sensor) is an integral part of the system. The methodology, results, and applications will be discussed in this section.

Methodology

The speciality of the sensor is that it can scan along a plane surface and allows detection over a large distance (5 m radial distance). It does not have high accuracy because of its use in mobile robots. But the current application requires better performance and thus a sensor calibration

process is to be done. The laser range finder provides polar coordinates, which are then converted into Cartesian coordinates. These coordinates are found with respect to the sensor coordinate frame. Figure 2 shows the configuration of the coordinate systems of the projective screen (P) and the sensor (S). There are methods for obtaining transformation from a camera coordinate system to a sensor coordinate system⁴⁴ when such a sensor is used along with a camera. It is not suitable for the current application since we have a projective screen instead of a camera. Therefore, we have implemented the following methodology.

The equation to obtain the nominal coordinates on the projected or the screen coordinate system P is given by

$$\mathbf{x}_P = \mathbf{T}\mathbf{x}_S \quad (1)$$

where $\mathbf{x}_S \equiv [r \cos\theta, r \sin\theta, 1]^T$ and $\mathbf{x}_P \equiv [\bar{x}_P^T, 1]^T$; $\bar{x}_P \equiv [x, y]^T$ are the nominal readings in sensor (S) and projected screen coordinate systems (P), respectively. Projected screen coordinates in 2D are given by \bar{x}_P , as defined above. Measurements in polar coordinates obtained from the sensor are given by r and θ (Figure 2), whereas \mathbf{T} is the 3×3 homogenous transformation matrix relating the sensor coordinate system (S) to the projective screen coordinate system (P). It is given by

$$\mathbf{T} \equiv \begin{bmatrix} \cos\alpha & -\sin\alpha & s_x \\ \sin\alpha & \cos\alpha & s_y \\ 0 & 0 & 1 \end{bmatrix}$$

The nominal value of the angle between the frames about Z axis is given by α and the translation is given by $[s_x, s_y]^T$. They are indicated in Figure 2. Let $\hat{\mathbf{x}}_P$ be the nominal vector of coordinates in P coordinate system, as measured by the laser range finder, and $\hat{\mathbf{x}}_a \equiv [x_a, y_a]^T$ be the actual coordinates. The differences represent the errors. It occurs due to the presence of inaccuracies in the nominal parameters caused by the measurements or other phenomena. The error vector between the two measurements is given by

$$\Delta\mathbf{x} = \hat{\mathbf{x}}_P - \hat{\mathbf{x}}_a \quad (2)$$

which can be theoretically estimated by partial derivative of \bar{x}_P with respect to the parameters, namely, $\zeta \equiv [r, \theta, \alpha, s_x, s_y]^T$. The error $\Delta\mathbf{x}$ can then be expressed in terms of the errors in ζ , that is, $\Delta\zeta$, as

$$\Delta\mathbf{x} = \mathbf{C}\Delta\zeta \quad (3)$$

where the 2×5 matrix \mathbf{C} is given by

$$\mathbf{C} \equiv \begin{bmatrix} \cos(\alpha + \theta) & -r \sin(\alpha + \theta) & -r \sin(\alpha + \theta) & 1 & 0 \\ \sin(\alpha + \theta) & r \cos(\alpha + \theta) & r \cos(\alpha + \theta) & 0 & 1 \end{bmatrix} \quad (4)$$

Matrix \mathbf{C} in equation (4) is nothing but the identification Jacobian and $\Delta\zeta \equiv [dr, d\theta, d\alpha, ds_x, ds_y]^T$ is the vector of parameter errors that needs to be found. Errors in r and θ may occur due to the shape of the contour which is detected (though the sensor gives multiple scans of only the outer

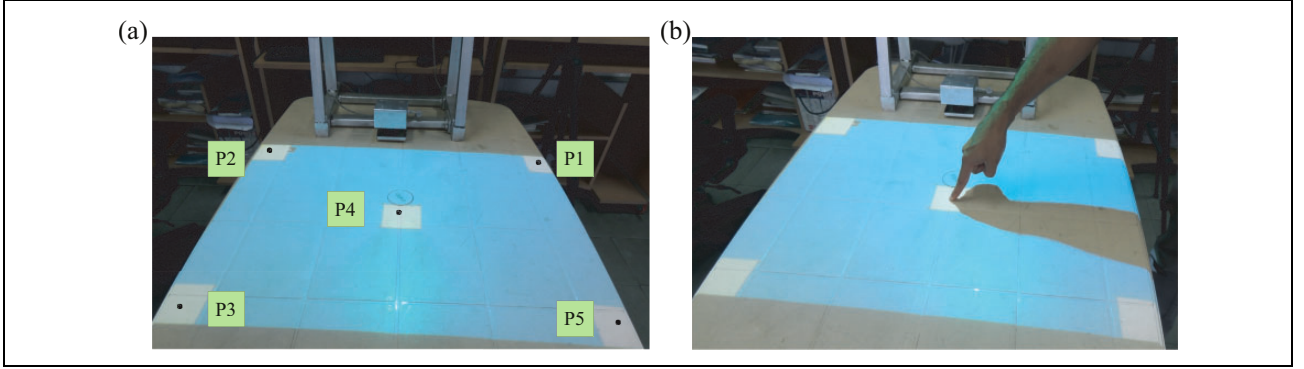


Figure 3. Measurement positions for calibration. (a) Measurement locations and (b) user action.

profile of an object that is scanned, the final point considered will be an average of all these points and this will be discussed in ‘Learning-based approach’ section) or due to some internal measurement errors in the sensor. Errors in s_x , s_y and α may also occur due to the manual inaccuracies of measurement owing to the sensor coordinate system

being not physically accessible. By concatenating multiple readings, the following expression is obtained

$$\Delta \mathbf{x}' = \mathbf{C}' \Delta \zeta \quad (5)$$

where

$$\Delta \mathbf{x}' \equiv \begin{bmatrix} dx_1 \\ dy_1 \\ \vdots \\ dx_m \\ dy_m \end{bmatrix} \quad \text{and} \quad \mathbf{C}' \equiv \begin{bmatrix} \cos(\alpha + \theta_1) & -r_1 \sin(\alpha + \theta_1) & -r_1 \sin(\alpha + \theta_1) & 1 & 0 \\ \sin(\alpha + \theta_1) & r_1 \cos(\alpha + \theta_1) & r_1 \cos(\alpha + \theta_1) & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \cos(\alpha + \theta_m) & -r_m \sin(\alpha + \theta_m) & -r_m \sin(\alpha + \theta_m) & 1 & 0 \\ \sin(\alpha + \theta_m) & r_m \cos(\alpha + \theta_m) & r_m \cos(\alpha + \theta_m) & 0 & 1 \end{bmatrix}$$

In equation (5), m is the number of readings that were taken by making measurements in different areas of the workspace. Note that r_j , θ_j , x_j and y_j , for $j = 1, \dots, m$, were the corresponding nominal values of the j th position in the projector coordinate frame P , where the measurement was done. During the calibration process, a user was asked to touch some marked positions (Figure 3) and readings were taken from the sensor. Though minimum of three points are sufficient for solving equation (5), more number of points would improve the estimation.

Moreover, the position errors would increase with the distance due to radial measurements. Hence, it will be suitable to make measurements at extreme points of the screen away from the sensor. Keeping the above points in mind, corrections required in the parameters are obtained from equation (5) as

$$\Delta \zeta = (\mathbf{C}'^T \mathbf{C}')^{-1} \mathbf{C}'^T \Delta \mathbf{x}' \quad (6)$$

where the components of $\Delta \zeta$, namely, dr and $d\theta$ values were used to update r and θ obtained from the sensor. The rest of the values can be used to update the nominal values of the parameters α , s_x and s_y used in the transformation matrix (\mathbf{T}). Note that one may also use the damped least squares optimization like Levenberg–Marquardt⁴⁵ to

obtain $\Delta \zeta$. This works better when the number of points for calibration is large. This was used in the experiments which were carried out in ‘Varying number of points for calibration’ and ‘Recalibration’ sections.

Once the calibration is done, it is possible to track the position of the user’s hand/finger accurately through appropriate feedback. This makes the system interactive, that is, the system can determine the position of the user’s finger or palm that touches the screen and give suitable feedback either through visual or audio signal. One advantage of this calibration methodology is that any type of screen can be easily calibrated and then be used as a touch sensitive screen at a very low cost. This will be discussed in ‘Recalibration’ sub section.

An illustration

This sub section explains how the calibration was performed using five points. In practice, one should use more number of points for improved accuracy. First, the measurements (Figure 3(a)) of the five points on the surface where the projection was to be made were taken. The measurements are given in Table 1. The corrections obtained after applying the calibration formula

Table 1. Measurements of calibration points.

Points	Actual ^a			Sensor readings ^b		
	x_a (mm)	y_a (mm)	r (mm)	θ (°)	x (mm)	y (mm)
P ₁	1520	0	1440.6	21.8	1463	-386
P ₂	0	0	1330.2	-44.2994	0	0
P ₃	0	1160	651.7	-98.2	284	1045
P ₄	400	580	669.2	-38.3	514	427
P ₅	1610	1130	851.9	71.7	1738	685

^aMeasurements were made using a ruler of resolution 1 mm.

^bCalculations for x and y were made using equation (1).

Table 2. Calibration results.

dr (mm)	66.8
$d\theta$ (°)	0
$d\alpha$ (°)	13.1
ds_x (mm)	-242.1
ds_y (mm)	277.6

Table 3. Parameters used during calibration.

Screen width (mm)	1550
Screen height (mm)	1140
s_y (mm)	952
s_x (mm)	929
α (°)	-90

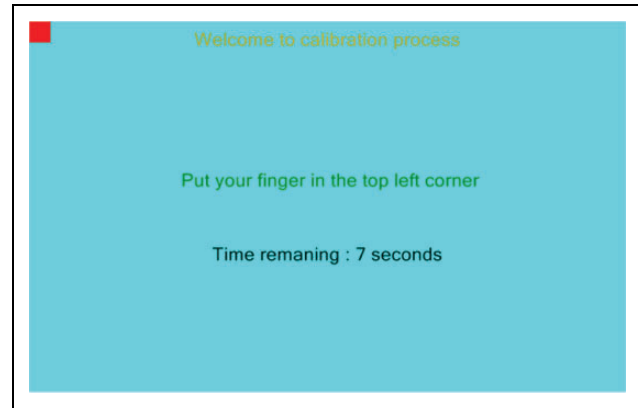
(equation 6) are given in Table 2. Table 3 shows the nominal parameters and other projection specific parameters that were used for the computation. Once the corrections are implemented, it is possible to accurately determine the point on the screen that is being touched. Note that the correction in θ , that is, $d\theta$ of Table 2 was found to be zero. It was incorporated in the value of α as these are along the same axis.

Varying number of points for calibration

To perform the calibration and to evaluate the performance of the algorithm with varying number of points, the following experiment was done. Screen size of 1550×1140 mm was considered with a total of 25 different points on it. Positions were chosen on a 5×5 grid, and measurements were taken using ruler of 1 mm resolution on the projector coordinate system P . Calibration was done multiple times by varying the number of points as 5, 10, 15 and 20. The remaining points, that is, 20, 15, 10 and 5, respectively, were used for evaluating the accuracy of the algorithm. The final errors obtained are shown in Table 4. Best results were obtained when using 20 points. The norm of the differences between the estimated and actual, averaged over the 5 points, gave 22.4 mm as the error of estimation. It is clear from Table 4 that the more the number of points used for calibration, the more is the accuracy.

Table 4. Calibration considering different number of points.

Number of points	Errors (mm)
5	44.1
10	33.4
15	24.9
20	22.4

**Figure 4.** Screen for calibration.

Recalibration

Calibration is normally done as a separate step at the beginning of construction of a system. For a change in the orientation of the projected screen and the sensor, the calibration data need to be changed. Also, if a different screen was used, the calibration has to be redone too. The calibration methodology explained in 'Methodology' sub section was designed in such a way that it can be easily integrated with the game, and the whole system will be adaptable to different kinds of displays. For this, a user interface was designed, which is to be displayed on the projected screen to guide a user. Before starting the calibration, the user will have to enter approximate maximum dimension of the projective screen, distance and rotation between the two coordinate systems (Figure 2). Later, the user will have to touch some marked positions at the corners of the screen (similar to those shown in Figure 3), which is indicated in Figure 4. The laser range finder, which is running on a different thread of the program, will be able to record reading for position of the hand on the projective screen within 10 s. Once this process is finished for all the points, the required calibration values can be calculated and saved in a file for future use. To test the applicability of the recalibration algorithm, it was tested on screens of several types and sizes. Besides the projected surface of Figure 1, LCD screen of Figure 5 was also calibrated. A video showing a normal LCD screen working as a touch-sensitive screen can be seen at <https://www.youtube.com/watch?v=GNEbfTbyrmM>.

The algorithm has been tested on three different screen sizes. First, a projector was used to produce displays of different sizes, which are shown in Table 5. The corresponding errors were calculated using the methodology discussed

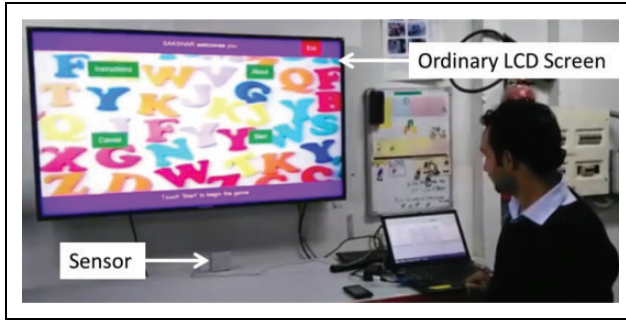


Figure 5. Ordinary LCD screen converted to touch sensitive screen. LCD: liquid crystal display.

Table 5. Calibration for different screen sizes

Screen size (mm)	Errors (mm)
710 × 540	10.4
1135 × 930	16.7
1550 × 1140	22.4

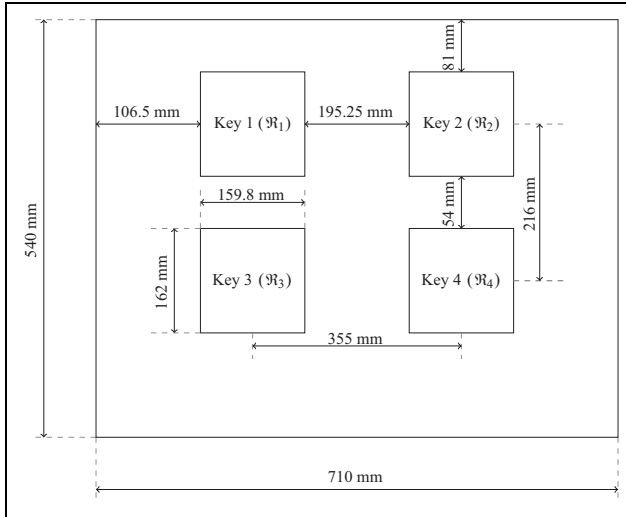


Figure 6. Arrangement of four keys on the displayed screen.

in ‘Varying number of points for calibration’ sub section with 20 points for calibration and 5 points for the measurement of errors. It can be noted that the calibration performed well for all the different screen sizes. The size of keys in the game is in the order of 150 mm for 710 × 540 mm screen, which is shown in Figure 6. For this, the error is only 10.4 mm which is quite acceptable for the detection of touch. For larger sized displays, the errors obtained are higher as expected but acceptable. After the calibration, the game was also run on all the screens without any issues.

Learning-based prediction

As briefly mentioned in ‘Introduction’ section, the usage styles of different people may affect the performance of the

interactive system. This issue and the possible solution are discussed in this section.

Problem

To have an understanding about the issue arising due to varied usage styles, some graphical explanation will be provided. Figure 7(a) shows the user touching the planar surface on which the game is to be displayed. Figure 7(b) shows the points detected by the sensor and the corresponding mean points that were obtained. It can be easily inferred that the sensor detects only a contour of the palm which is different from the centre of the palm. Added to it, response to a projected graphical user interface key varies from one person to another. This has been observed while demonstrating the IDAT in IREX 2013.¹⁹ The reason for this variation can be due to the way a person holds his hand while responding to the interface, that is, either as a fist, single finger or using palm surface. Such a variation also might be attributed to the perception of the centre of the key by the user.

Other variation includes the size of the palm. There is a significant variation in palm sizes depending on genders and ages. So, there is a need of customization for each person. The kinematics of human arm makes it convenient to make planar movements within a circular or elliptical boundary. However, the boundary of the projected key was defined by straight lines. The errors due to these variations can be compensated using an online learning-based approach, which is explained below.

Learning-based approach

What we want to know is the key which a user intends to select. To accommodate the variation in the user’s habits, a learning-based methodology is proposed here to predict the user’s intention. Any user who starts a game has to initially make some choices. Using these data, his or her hand movements can be learnt. A Bayesian learning method⁴⁶ was used to learn the pattern of hand placement. For this, consider that the person was supposed to touch the point with coordinates \mathbf{x}_a as per the projected display. The user finally touched the point with coordinates \mathbf{x}_a' , which introduces a variation, $\mathbf{x}_a - \mathbf{x}_a'$. As stated earlier and visible in Figure 7(b), assume that the user has touched the screen and multiple radial coordinates were obtained. For the k th point on the q th scan, the radial coordinates obtained from the sensor be $[r_{k,q}, \theta_{k,q}]$. Using the identified parameters (assumed that the calibration was done as discussed in ‘Calibration’ section), these coordinates were transformed into projector coordinate system (P) as $\mathbf{x}_{k,q}$ with the following specifications

$$\mathbf{x}_{k,q} = \{[x_{k,q}, y_{k,q}]^T \in \mathbb{R}_2 | [x_{k,q}, y_{k,q}]^T \neq [0, 0]^T, \\ \text{where } x_{k,q} < x_{\max}, y_{k,q} < y_{\max}, k \in \mathbb{Z}, q \in \mathbb{Z}, \\ 1 \leq k \leq n_1, 1 \leq q \leq n_2\}$$

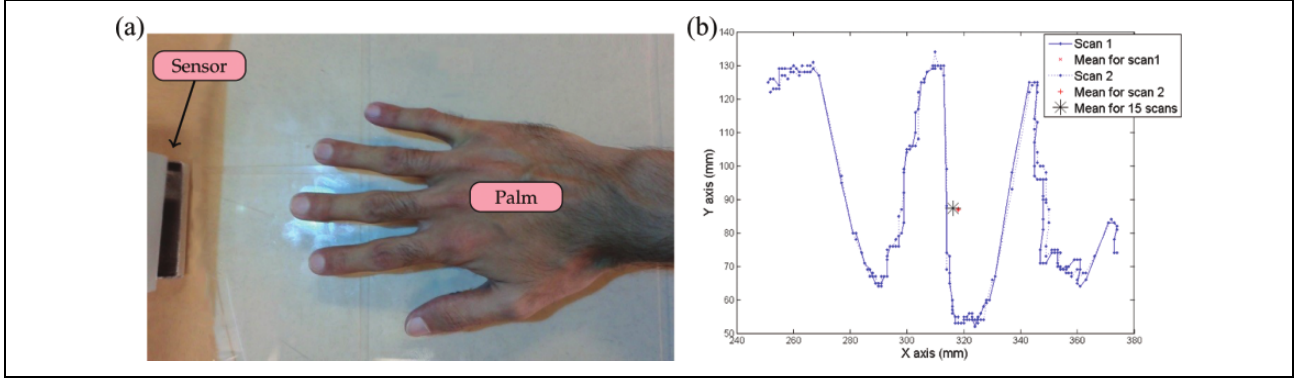


Figure 7. Coordinate data captured by sensor. (a) User making a choice by placing the hand and (b) multiple readings from sensor.

where n_1 is the number of readings obtained from the sensor in one out of n_2 cycles of measurements. Note that the sensor works at 1 Hz speed and therefore produces multiple scans when the arm is placed on the screen. Multiple scan profiles are shown in Figure 7(b) whose mean values are very close to one another (not distinguishable in the figure). Hence, one may use few scan data instead of multiple scans (e.g. 15 scans as shown in the figure). The values of x_{\max} and y_{\max} are defined by considering an approximate maximum size of the screen. Based on the sensor data, the calculated coordinates can be represented by

$$\mathbf{x}_a' = E[\mathbf{x}_{k,q}] = \frac{\sum_{k,q} \mathbf{x}_{k,q}}{n_1 n_2} \quad (7)$$

where $E[\mathbf{x}_{k,q}]$ is the expectation of $\mathbf{x}_{k,q}$. In all further calculations, \mathbf{x}_a' is considered as representing the point on the screen that was touched by the user. This is easier than considering all the points on the profile of scans. If the user makes multiple attempts to touch specific region or key i , then the values of the coordinates obtained can be assumed to become a bivariate normal distribution with mean $\boldsymbol{\mu}_i$ and covariance matrix $\boldsymbol{\Sigma}_i$. The keys and their respective orientations for the display screen are shown in Figure 6. Given the measurement \mathbf{x}_a' from the sensor, the log likelihood⁴⁶ of the user having chosen a region \mathcal{R}_i is given below

$$\begin{aligned} f\left(\frac{\mathbf{x}_a'}{\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i}\right) &= \ln \left[P\left(\frac{\mathcal{R}_i}{\mathbf{x}_{k,q}}\right) \right] = \frac{-d \ln(2\pi)}{2} - \ln(|\boldsymbol{\Sigma}_i|) \\ &\quad - \frac{(\mathbf{x}_a' - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1} (\mathbf{x}_a' - \boldsymbol{\mu}_i)}{2} \\ &\quad + \ln[P(\mathcal{R}_i)] \end{aligned} \quad (8)$$

where d is the dimension of the vector \mathbf{x}_a' . In this case, $d = 2$, because we are considering a two-dimensional plane. The term $P(\mathcal{R}_i)$ is the prior probability that the region \mathcal{R}_i corresponding to key i was chosen by the user. Note, $i \in \mathbb{Z}$, $i \in (1, n_b)$, n_b being the number of keys that is displayed on the screen. In this case, we project four different keys on the screen and thus $n_b = 4$. The value of $P(\mathcal{R}_i)$ may be constant for all i if all the events are

equiprobable, which is the case for the current application. Thus, $P(\mathcal{R}_i) = \frac{1}{n_b} = \frac{1}{4} = 0.25$. The value of log likelihood can be utilized to make choices of the key that was touched. The final choice is region \mathcal{R}_c corresponding to key c , where

$$c = \arg \max_i f\left(\frac{\mathbf{x}_a'}{\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i}\right) \quad (9)$$

With this formulation, errors due to varying usage patterns can be corrected. Boundaries of the decision regions for each key will be defined by a quadric curve, which fits best the movement of the human arm. The training for model can be done using a few slides at the beginning of the game where the user may be asked to touch particular regions of the projective screen or the data can be gathered as the user keeps interacting with the device. Once the training of the model is complete, the estimated parameters can be used to decide whether a particular region has been selected or not based on the value of the log likelihood given by equation (8). At the initial stage of the game, the values of mean and covariance can be initialized as the coordinates of the centre of the key and the half size of length and breadth of the key, respectively. As the game progresses, these values can be updated.

Experimental results

As discussed earlier, parameters corresponding to the hand positions of a user while selecting a particular key were to be learnt. This implies that the mean $\boldsymbol{\mu}_i$ and covariance matrix $\boldsymbol{\Sigma}_i$ corresponding to the i th key are to be determined. They are obtained as

$$\boldsymbol{\mu}_i = \frac{\sum_v \mathbf{x}_{i,v}'}{n_i} \quad (10)$$

where

$$\boldsymbol{\Sigma}_i \equiv \begin{bmatrix} \sum_v (x_{i,v}' - \mu_{i,1})^2 & \sum_v (x_{i,v}' - \mu_{i,1})(y_{i,v}' - \mu_{i,2}) \\ \sum_v (x_{i,v}' - \mu_{i,1})(y_{i,v}' - \mu_{i,2}) & \sum_v (y_{i,v}' - \mu_{i,2})^2 \end{bmatrix} \quad (11)$$

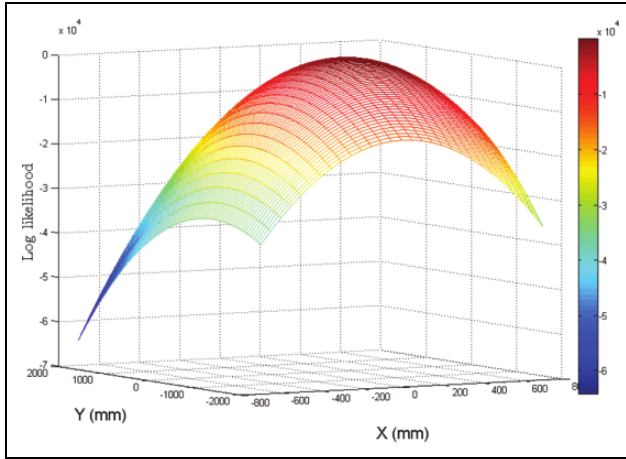


Figure 8. Discriminant function corresponding to key on top left.

in which $\mathbf{x}_{i,v}' \equiv [x_{i,v}', y_{i,v}']^T$ and $\boldsymbol{\mu}_i \equiv [\mu_{i,1}, \mu_{i,2}]^T$. The user is asked to choose the i th key n_i times. The vector $\mathbf{x}_{i,v}'$ represents the coordinates obtained from the sensor, that is, the scanning laser range finder at the v th attempt of i th key obtained using equation (7). The data corresponding to the user's hand movement were calculated and stored. The number of trials and the methodology to be followed are also to be decided, which will be discussed below.

Training the model. Since the performance measurement for human arm positioning in response to visual stimulus was not available in literature, we had to design an experiment to calculate these parameters. We relied on the methodology by which performance measurement for industrial manipulators was made. This is based on the notion that the kinematics of human arm can be compared to that of an industrial manipulator which is modelled to mimic human arm anthropometry. The quantification of performance measurement for such an industrial robot is based on computing mean and standard deviation as per ISO 9283.⁴⁷ Hence, the training was done using the conditions given in ISO 9283 for measuring the repeatability of an industrial robot. The user is made to choose four different options in a sequence. First, the user touches the key on top left corner, then the top right corner, followed by that on bottom right corner and finally on bottom left corner. This was repeated 30 times for a particular user. Once the experiment was done, the mean ($\boldsymbol{\mu}_i$) and covariance ($\boldsymbol{\Sigma}_i$) were calculated. Figure 8 shows $f\left(\frac{\mathbf{x}_{i,v}'}{\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i}\right)$ for the workspace corresponding to a particular key or region. By using multiple discriminant functions for the whole workspace, we can deduce the boundaries for each key as per the user's style of usage. Therefore, the decision that a particular key or region has been touched can be made using the boundaries shown in Figure 9. It can be noted that the boundaries of the decision surfaces are not made of straight lines but quadrics as discussed in 'Learning-based approach' section. Also the decision regions are slightly shifted from the place where the original images were projected. The star marks

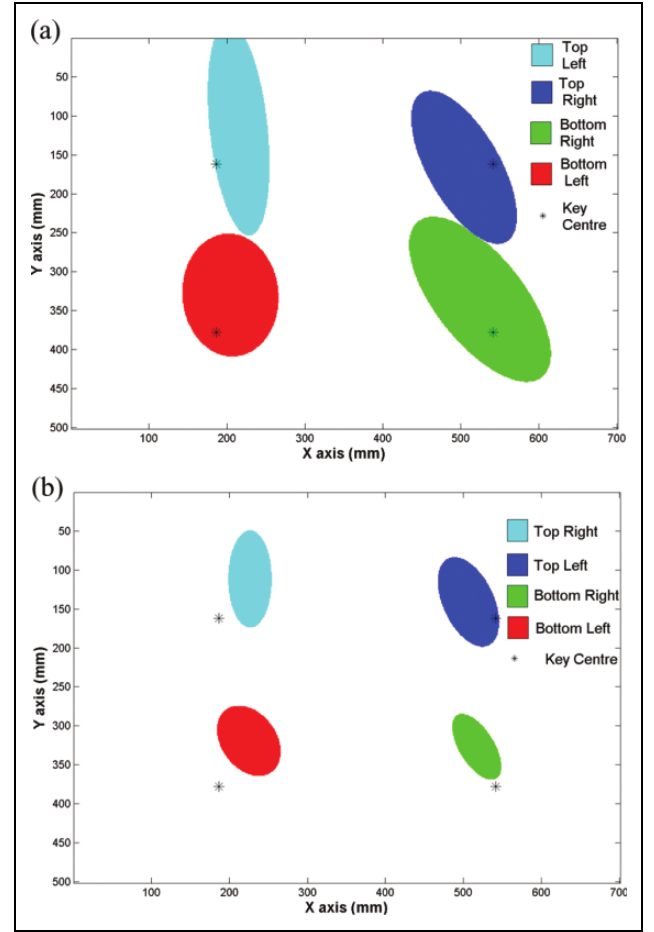


Figure 9. Decision regions regarding a key is being pressed or not. (a) User 1 and (b) user 2.

show the centre of the key that is displayed. Another interesting observation is that the usage style for two different people (users 1 and 2, shown in Figure 9) varies. This points to the possibility that the response of a system for a particular person can be customized to suit his or her usage style.

Accuracy. To ascertain the accuracy of the proposed method, the device was used by 10 different users (users were males of ages 20–35 years) and the coordinates of the points where they touched were recorded while playing the game. As discussed earlier, the users touched the keys in a sequence. More than 700 points corresponding to four different keys were recorded. The data were divided into 70% training data and the rest as testing data. The parameters estimated using the training data were used to make predictions. The confusion matrix that represents the predicted classification of the keys according to the proposed method is shown in Table 6. The accuracy of the prediction of each key is shown in Table 7. The overall accuracy of prediction is 96.17%. The points used in conducting the tests are shown in Figure 10.

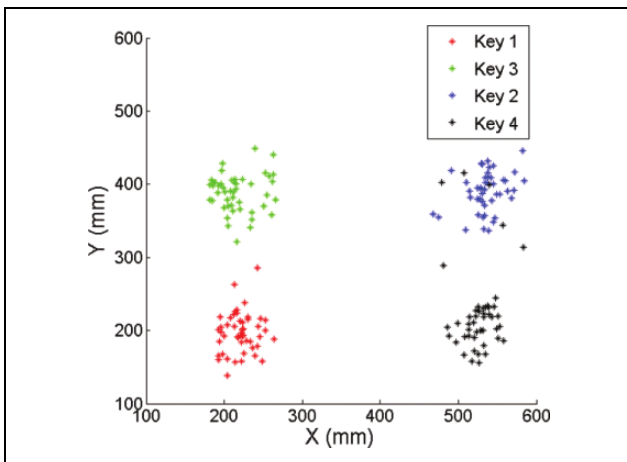
Advantage. An advantage of the above statistical analysis is that it can be used to decide the suitable size of the keys to

Table 6. Confusion matrix.

Input key no./Output key no.	1	2	3	4
1	48	1	0	0
2	0	47	0	0
3	0	0	44	3
4	0	0	5	41

Table 7. Prediction accuracy.

Key no.	Accuracy (%)
1	97.96
2	100
3	93.62
4	89.13
Overall	96.17

**Figure 10.** Data points used for testing.

be displayed at the interface. Statistically, this should be of the order of six times the standard deviation in each axis. The maximum standard deviations in X and Y directions are 25.3 and 27.5 mm as per the data shown in Figure 10. Therefore, the size of the keys can be of the order of 151.6 and 165.1 mm along X and Y directions, respectively. It also represents the minimum possible distance between the centres of two keys. This inference can also be substantiated by visually inspecting Figure 10. An average square-shaped key of size 160 mm and the centre-to-centre distance of 165 mm or more between any two keys along X or Y directions would be sufficient. As shown in Figure 6, the key size is 159.8 mm and the centre-to-centre distances between two keys are 355 and 216 mm along X and Y directions, respectively.

Note that SAKSHAR was designed to teach alphabets (teaching mode) and test the learning outcome (details are explained in Appendix 2). This requires a person to use the device multiple times which implies that the user will touch the screen multiple times. Therefore, customization of the system for the personal use is viable. Such method can also be extended to similar interactive devices.

Studies regarding adaptive learning of a user's intention were reported in the studies by Bi and Zhai and Yin et al.^{48,49} But, no information about the experimental conditions to standardize the measurements was reported. To our knowledge, this is the first time that the variation in usage style for a large display is experimentally established.

Field trials

A suitable way in which the performance of such systems can be evaluated is through demonstrating it to multiple users and then evaluating their personal feedback/learning outcome. This section presents the field trials conducted to evaluate the suitability of the device in primary school education.

User survey

User survey was conducted by taking the device to a school to demonstrate to the students and teachers. Thereafter, personal feedbacks were obtained. The aspects for which feedback was obtained are Usability/Operability/Accessibility, Interest/Fun/Pleasure/Fulfilment and Usefulness/Availability/Effectiveness. The Feedback regarding Usability/Operability/Accessibility represents the ease with which people could operate the system. The feedback corresponding to Interest/Fun/Pleasure/Fulfilment represents how much the people are interested to play the game. Finally, the feedback corresponding to Usefulness/Availability/Effectiveness shows the extent to which the goal of SAKSHAR system, that is, helping children to learn alphabets, has been attained. For each category, a nine-point scale was provided to choose in which points 1, 5, and 9 represent bad, average and good, respectively.

A survey was conducted in Kendriya Vidyalaya School⁵⁰ of IIT Delhi Campus. Four students each from classes 1 to 5 were selected. Their age-wise distribution is shown in Figure 11(a). Equal number of boys and girls were ensured as indicated in Figure 11(b). Additionally, from each class, two students with best academic performance index and the other two with relatively lesser performance index were selected. The following procedure was adopted while collecting the user survey details. Initially, the students were given introduction to the device and then allowed to play the game a few times on his/her own allowing them to get acquainted with it. This allowed each student to familiarize with the device. Later, a student was asked to play the game from beginning till the end on his/her own without any assistance. After this, the student was verbally asked questions regarding the game. The student was asked to give grades to the specific feature of the device. Using superlatives and with reference to the school grading system, the student was made to understand the feedback system employed in our study. The feedback was then recorded. All communication with the students was made in Hindi since it is the mother tongue of all the students involved in the study. All the participating students

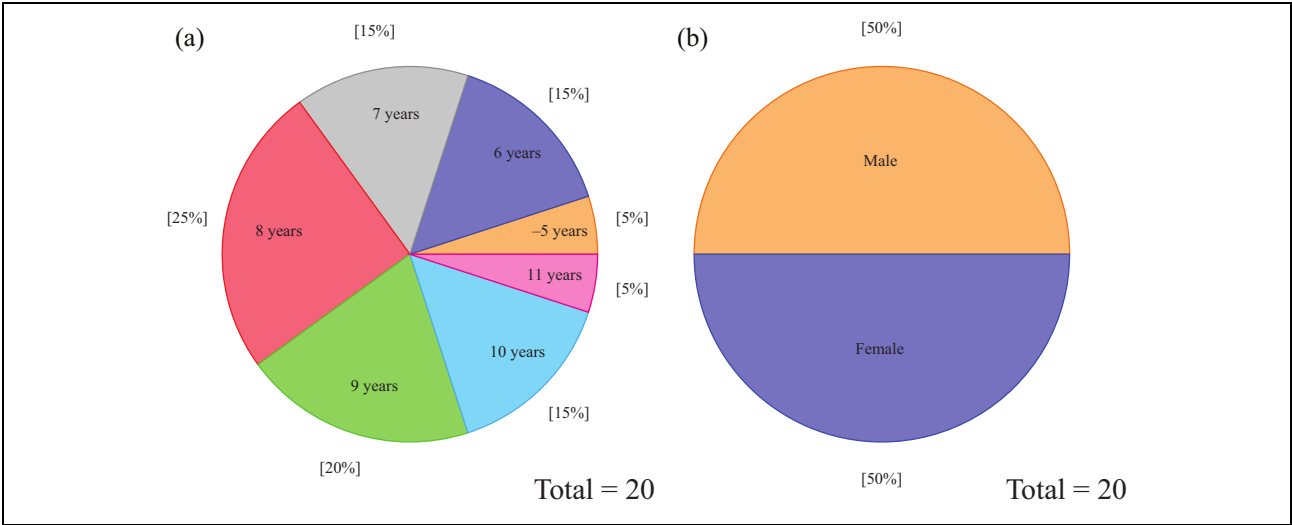


Figure 11. User's age and gender. (a) Age group distribution and (b) gender distribution.

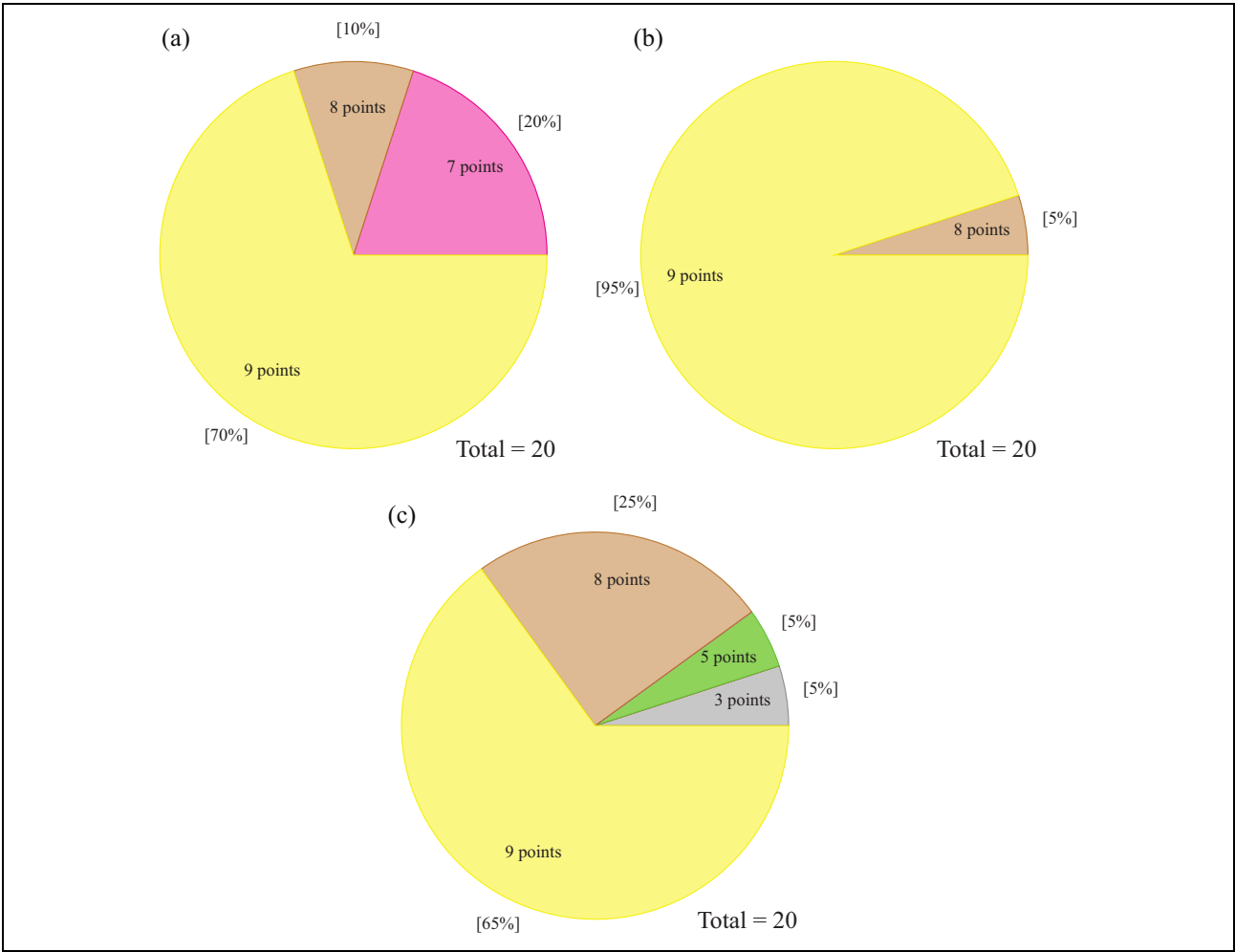


Figure 12. Survey results from trials in school. (a) Usability/Operability/Accessibility, (b) Interest/Fun/Pleasure/Fulfilment and (c) Usefulness/Availability/Effectiveness.

SAKSHAR, PAR LAB IIT DELHI

NAME: _____ CLASS: _____

	a) Apple b) Aplle c) Apple d) Aapple		a) Boll b) Bal c) Ball d) Bol
	a) Gril b) Girl c) Gerl d) Gerri		a) Ornge b) Orange c) Oraange d) Orangeie

Figure 13. Questions for test.

were given sweets as gift at the end of the whole process. The feedback obtained by the students is represented in Figures 12(a) to (c). A majority of the students (70%) have reported that the device is easy to use by giving maximum, that is, 9 points. It is interesting to note that 95% of the students enjoyed using the device, whereas 65% of the people stated that the device is beneficial for them in learning alphabets by giving 9 points in the respective categories.

In order to ascertain the reliability of the user survey, the same was conducted twice separated by a period of 5 months. The first user survey results are shown in Figure 12. For the second survey, the variation in mean score was 0.6, 0.5 and 0.8, respectively, for Usability/Operability/Accessibility, Interest/Fun/Fulfilment and Usefulness/Availability/Effectiveness. For each metric, mean was calculated after adding the points given by all the children and dividing by the total number of children. The results

obtained during both the surveys are therefore quite similar and thus user survey data can be considered reliable.

Improvement in learning outcome

To understand the real impact of the system, we conducted another trial to check the learning outcome of the students. We picked up a class of 19 students and conducted an assessment test (test 1) to check the comprehension of the students. The test consisted of questions where the picture of an object was followed by four different spellings of the name of the object as shown in Figure 13. The students had to identify the right spelling and had to mark their choice on article by ticking on it. We chose the questions such that the overall comprehension of the children would be tested. This was done after incorporating the opinion of the school teachers and few educationists. Points were given for the correct answer and no negative marks were given for the wrong answer. Adequate time was given to the students so that the test would check their understanding and not merely their speed. Keeping this in mind, a suitable upper limit was set on the time to attempt the paper. Based on the results obtained from the test, we divided the class into two groups of 12 and 7 students with similar performance indices. It was also ensured that both the groups had equal ratios of boys to girls. These groups were called group A and group B.

Group A was then allowed to use the device and group B was taught using the regular teaching method similar to the one already used in the school. While teaching group B, we showed a video with the images of objects and their names. For each alphabet, the object was shown and the object name was pronounced. Additionally, spelling was explained. After teaching the students using these methods,

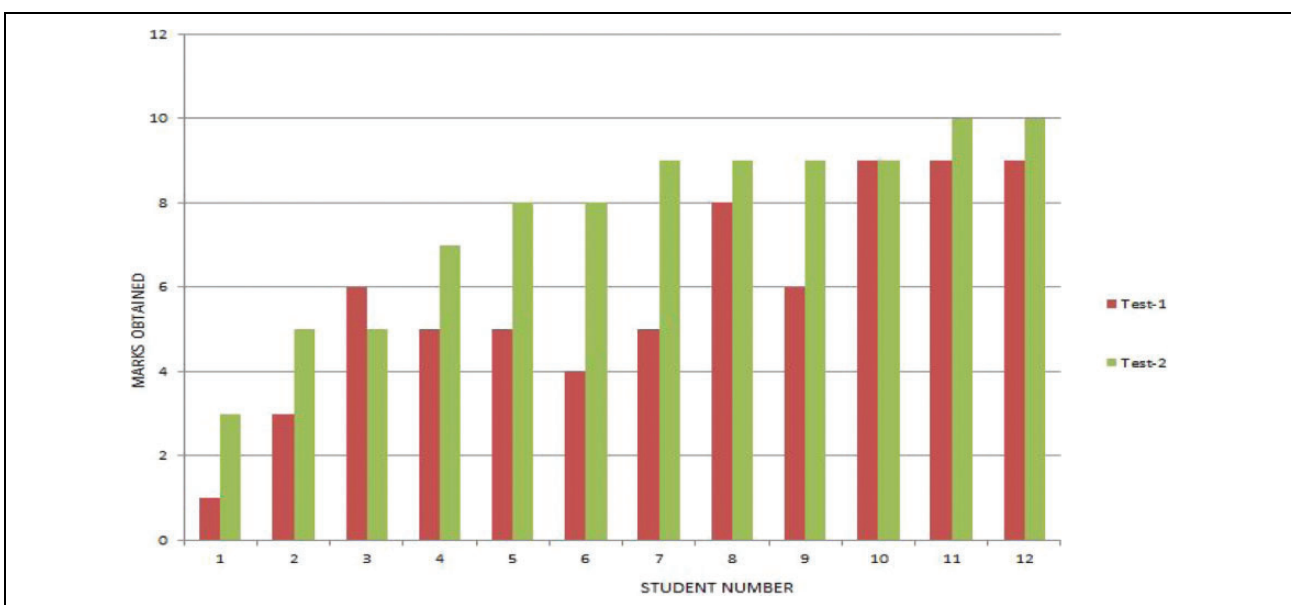


Figure 14. Group A tests.

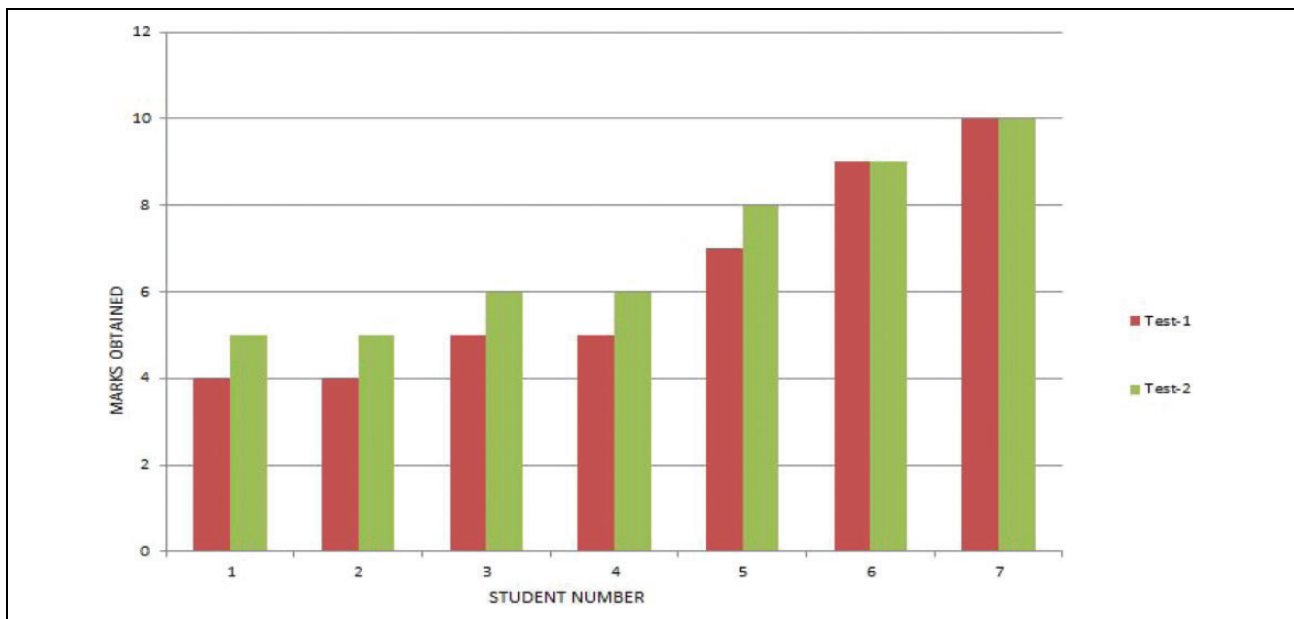


Figure 15. Group B tests.

another test (this is referred to as test 2) with the same design as the first one was conducted for both the groups. Improvement in score of group A was noticed. On an average, there was 12% increase in the marks of the students of group B, while there was a 33% increase in the marks of group A students who were taught using the SAKSHAR. The improvement in the scores of group A in the test are represented in Figure 14. The improvement in scores after and before being taught using regular teaching method for group B is shown in Figure 15. The improvement in the marks and understanding of the students in group A was 21% more than that of group B. This is mainly attributed to the involvement of the students with the interactive learning process rather than merely memorizing.

The results discussed here established that the device could prove to be a very useful learning aid especially in primary schools. The overall opinion about the SAKSHAR set-up was encouraging. There was also avenue for the people to give suggestions for improvement. Implementation of the system for teaching other subjects like geography, further lowering of costs using cheaper sensors, getting the system quickly available for purchase by users, and so on, were some of them. Efforts are now being made to implement some of the suggestions. For example, as discussed earlier, the learning-based prediction was made after evaluating the performance of the system during a feedback session.

Conclusions

A new device called SAKSHAR:IDVT was built for interactive learning of alphabets. Details of the development, particularly, the calibration of sensor, that is, the 2D

sensor to project, to projector screen is an important contribution of this article. Even though the fundamental mathematics is well known, but using it for sensor to projector calibration based on the observed data made the device reliable. The generalized formulation can be used for implementation of interactive screen for different types of displays using the same sensor. The calibration algorithm enabled us to get very good results with regards to providing suitable audio/visual feedback by detecting the point of touch with respect to the projected display. A learning-based methodology to estimate user's intentions was also proposed, which is another contribution of this article. This helps us to make suitable decisions about which key is being touched. This might have application in other touch-sensitive displays also. As an application, a game was developed using this platform with audiovisual feedback to encourage learning by primary school students. A video of the device being used can be seen at <https://www.youtube.com/watch?v=zDDbn8U4El4>. The game features multiple language-based user interfaces. In summary, the article has the following three main contributions:

1. Development and performance measurement of a generalized calibration process which can be used to convert a surface or a large screen to an interactive one using 2D scanning laser range finder.
2. Development and field test of IDVT which is a portable interactive screen that can be used in teaching alphabets to children.
3. A new learning-based prediction of a user's intention.

Acknowledgements

The authors would like to thank Suzuki Foundation, Kayamori Foundation of Informational Science Advancement and Waseda University for supporting this collaborative endeavour and sponsoring our research visits to Japan. The authors would like to present their sincere thanks to BRNS/BARC for their support regarding the setting up of Programme for Autonomous Robotics Laboratory at IIT Delhi where most of the development work took place. Mr Arun Dayal Udai and Ms Bharti helped in the sensor integration and Hindi content generation, respectively. Mrs Nidhi and Mrs Pinkoo Chawla from Kendriya Vidyalaya (KV) School, IIT Delhi helped in conducting the trials. Several KV students are also thanked for their participation.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by Suzuki Foundation (26-zyo-i29), Kayamori Foundation of Informational Science Advancement (K26kenXIX-453) and Waseda University (2014K-6191, 2014B-352, 2015B-346). This work was also supported by BRNS/BARC.

References

1. SUR40 Specialized Display. <http://www.samsung.com/uk/business/business-products/large-format-display/specialised-display/LH40SFWTGC/EN> (accessed 28 January 2016).
2. Microsoft Pixsense. https://en.wikipedia.org/wiki/Microsoft_PixSense (accessed 28 January 2016).
3. Microsoft Surface Table. <http://money.cnn.com/2012/06/19/technology/microsoft-surface-table-pixsense> (accessed 28 January 2016).
4. Mignonneau L and Sommerer C. Designing emotional, metaphorical, natural and intuitive interfaces for interactive art, edutainment and mobile communications. *Comput Graphics* 2005; 29: 837–851.
5. Smith CE. *Design for the other 90%*. New York: Editions Assouline, 2007.
6. Wellner P. Interacting with paper on the digital desk. *Commun ACM* 1993; 36(7): 87–96.
7. Handheld projector. http://en.wikipedia.org/wiki/Handheld_projector (accessed 28 January 2016).
8. Mistry P and Maes P. SixthSense – a wearable gestural interface. In: Yuko Oda and Mariko Tanaka (eds) *Proceedings of SIGGRAPH Asia, emerging technologies*, Yokohama, Japan, 16–19 December 2009, p. 11, New York, NY: ACM.
9. Mistry P. The thrilling potential of SixthSense technology. http://www.ted.com/talks/pranav_mistry_the_thrilling_potential_of_sixthsense_technology.html (accessed 8 April 2017).
10. SixthSense. <http://en.wikipedia.org/wiki/SixthSense> (accessed 28 January 2016).
11. Sutherland I. The ultimate display. In: Wayne A. Kalenich (ed) *Proceedings of IFIP 65*, New York City, 24–29 May 1965, pp. 506–508. London, UK: Macmillan and Co.
12. Google Glass Tech specs. https://support.google.com/glass/answer/3064128?hl=en&ref_topic=3063354 (accessed 28 January 2016).
13. Matsumaru T and Akai K. Step-on interface on mobile robot to operate by stepping on projected button. *Open Autom Control Syst J* 2009; 2: 85–95.
14. Matsumaru T. A characteristics measurement of two-dimensional range scanner and its application. *Open Autom Control Syst J* 2009; 2: 21–30.
15. Liu Y, Jiang Y and Matsumaru T. Development of image-projective desktop arm trainer, IDAT. In: *IEEE/SICE international symposium on system integration (SII)*, pp. 355–360, 2012.
16. Matsumaru T, Jian Y and Liu Y. Image-projective desktop arm trainer IDAT for therapy. In: Hyun-Taek Choi and Hyun Myung (eds) *The 22nd IEEE international symposium on robot and human interactive communication (IEEE RO-MAN 2013)*, Gyeongju, Korea, 29 August 2013, pp. 501–506. New Jersey: IEEE.
17. Matsumaru T, Liu Y, Jiang Y, et al. Image-projecting desktop arm trainer for hand-eye coordination training. *J Robot Mech* 2014; 26(6): 704–717.
18. Holz C and Baudisch P. Understanding touch. In: Geraldine Fitzpatrick, Carl Gutwin, Bo Begole and Wendy A. Kellogg (eds) *Proceedings of the SIGCHI conference on human factors in computing systems*, CHI '11, Vancouver, BC, Canada, 07–12 May 2011, pp. 2501–2510. New York, NY: ACM.
19. IREX 2013 international robot exhibition. http://biz.nikkan.co.jp/eve/irex/english/_files/irex2013report_en.pdf (accessed 28 January 2016).
20. H.C.R.2014. <http://www.hcrjapan.org/english/hcr2014/> (accessed 5 February 2016).
21. Ron O, Herb W, Sinitskaya RN, et al. Computer game development as a literacy activity. *Comput Educ* 2009; 53(3): 977–989.
22. Bottino RM, Ferlino L, Ott M, et al. Developing strategic and reasoning abilities with computer games at primary level. *Comput Educ* 2007; 49: 1272–1286.
23. Tuzun H, Yilmaz-Soylu M, Karakus T, et al. The effects of computer games on primary school student's achievement and motivation in geography learning. *Comput Educ* 2009; 52: 68–77.
24. Chen JX. Learning abstract concepts through interactive playing. *Comput Graphics* 2006; 30: 10–19.
25. Collier BD and Scott MJ. Effectiveness of using a video game to teach a course in mechanical engineering. *Comput Educ* 2009; 53: 900–912.
26. Cheung M, Li KR and Zapart T. Computer-based edutainment for children aged 3 to 5 years old. In: *Emerging trends and challenges in information technology management*. Calgary: Idea Group Inc., pp. 293–297, 2006.
27. List of languages by total number of speakers. http://en.wikipedia.org/wiki/List_of_languages_by_number_of_native_speakers (accessed 28 January 2016).
28. The Nobel Peace Prize 2014. http://www.nobelprize.org/nobel_prizes/peace/laureates/2014/ (accessed 28 January 2016).
29. Hokuyo automatic co., ltd. https://www.hokuyo-aut.jp/02sensor/07scanner/urg_04lx_ug01.html (accessed 28 January 2016).

30. Kawata H, Ohya A, Yuta S, et al. Development of ultra-small lightweight optical range sensor system. In: Max QH and Meng Hong Zhang (eds) *2005 IEEE/RSJ international conference on intelligent robots and systems (IROS 2005)*, Edmonton, Canada, 2–6 August 2005, pp. 1078–1083. New Jersey: IEEE.
31. SMART Technologies Inc. invents new technology for touch-sensitive displays. <http://www.bobsguide.com/guide/news/2003/May/23/smart-technologies-inc-invents-new-technology-for-touch-sensitive-displays.html> (accessed 28 January 2016).
32. Bell M, Gleckman P, Zide J, et al. Computer vision based touch screen, June 26 2008. Patent application 11/929,947, USA, 2008.
33. Stødle D, Ha PH, Bjørndalen JM, et al. Lessons learned using a camera cluster to detect and locate objects. In: Bischof C, Martin B, Paul G, Gerhard RJ, Thomas L, Bernd M and Frans P (eds) *Parallel computing: architectures, algorithms and applications*, 4–7 September 2007, pp. 71–78. IOS Press.
34. Leap Motion. <https://www.leapmotion.com/> (accessed 28 January 2016).
35. Kinect for Windows. <http://www.microsoft.com/en-us/kinectforwindows/> (accessed 28 January 2016).
36. Agarwal A, Izadi S, Chandraker M, et al. High precision multi-touch sensing on surfaces using overhead cameras. In: Andy W and Clifton F (eds) *Second Annual IEEE international workshop on horizontal interactive human-computer systems (TABLETOP'07)*, Newport, Rhode Island, 10–12 October 2007, pp. 197–200. New Jersey: IEEE.
37. Dai J and Chung R. Making any planar surface into a touch-sensitive display by a mere projector and camera. In: *2012 IEEE computer society conference on computer vision and pattern recognition workshops (CVPRW)*, Providence, Rhode Island, 16–21 June 2012, pp. 35–42. New jersey: IEEE.
38. Hu J, Li G, Xie X, et al. Bare-fingers touch detection by the button's distortion in a projector–camera system. *IEEE Trans Circuits Syst Video Technol* 2014; 24(4): 566–575.
39. Sharma P, Joshi RP, Boby RA, et al. Projectable interactive surface using Microsoft Kinect v2: recovering information from coarse data to detect touch. In: Kyoichi Tatsuno (ed) *Proceedings of IEEE/SICE SII 2015*, Nagoya, Japan, 12–13 December 2015, New Jersey: IEEE.
40. OpenTK. <http://www.opentk.com> (accessed 28 January 2016).
41. Cgen-Audio. <http://www.opentk.com/project/Cgaudio> (accessed 28 January 2016).
42. NET Framework and .NET SDKs. <http://msdn.microsoft.com/en-us/vstudio/aa496123> (accessed 28 January 2016).
43. UrgCtrl C++. http://hyakuren-soft.sakura.ne.jp/dame_rbo_html/urgCtrl_cpp (accessed 28 January 2016).
44. Vasconcelos F, Barreto JP and Nunes U. A minimal solution for the extrinsic calibration of a camera and a laser-rangefinder. *IEEE Trans Pattern Anal Mach Intell* 2012; 34(11): 2097–2107.
45. Gill PR, Murray W and Wright MH. The Levenberg–Marquardt method. In *Practical optimization*. London: Academic Press, pp. 136–137, 1981.
46. Duda RO, Hart PE and Stork DG. *Pattern classification*. New York: Wiley, 2012.
47. ISO 9283, *Manipulating industrial robots performance criteria and related test methods*. Geneva, Switzerland: ISO, 1998.
48. Bi X and Zhai S. Bayesian touch: a statistical criterion of target selection with finger touch. In: Shahram Izadi and Aaron Quigley (eds) *Proceedings of the 26th Annual ACM symposium on user interface software and technology*, UIST'13, New York, NY, 8–11 October 2013, pp. 51–60. New York, NY: ACM.
49. Yin Y, Ouyang TY, Partridge K, et al. Making touchscreen keyboards adaptive to keys, hand postures, and individuals: a hierarchical spatial backoff model approach. In: Stephen Brewster and Susanne Bødker (eds) *Proceedings of the SIGCHI conference on human factors in computing systems*, CHI'13, New York, NY, 27 April–2 May 2013, pp. 2775–2784. New York, NY: ACM.
50. Kendriya Vidyalaya – NMR, JNU campus. <http://www.kvjnu.org/english/> (accessed 26 January 2016).
51. Joshi RP, Boby RA, Saha SK, et al. SAKSHAR: an image-projective desktop Varnamala trainer (IDVT) for interactive learning of alphabets. In: M. Bernardine Dias, Ayorkor Korsah (eds) *ICRA 2015 developing countries forum*, Seattle, 19–21 May 2015, pp. 1–3.

Appendix I

Specification of laser range finder

The technical specification of laser range finder/sensor is given in Table 1A. The overall size of the sensor is $50 \times 50 \times 70$ mm.

Table 1A. Specification of Hokuyo URG-04LX-UG01 sensor.

Specification	Value
Model no.	URG-04LX-UG01
Power source	5VDC $\pm 5\%$ (USB power)
Light source	Semiconductor laser diode ($\lambda=785$ nm), laser safety class I
Measuring area	20–5600 mm (white paper with 70mm \times 70 mm), 240°
Accuracy	60–1000 mm: ± 30 mm, 1000–4095 mm: $\pm 3\%$ of measurement
Angular resolution	Step angle: approximately 0.36° (360°/1024 steps)
Scanning time	100 ms/scan
Noise	25 dB or less
Interface	USB2.0/1.1 [Mini B] (full speed)
Command system	SCIP Ver.2.0
Ambient illuminance	Halogen/mercury lamp: 10,000 Lux or less, Fluorescence: 6000 Lux (max)
Ambient temperature/humidity	–10 to +50°C, 85% or less (not condensing, not icing)
Vibration resistance	10–55 Hz, double amplitude 1.5 mm each 2 h in X, Y and Z directions
Impact resistance	196 m/s ² , each 10 times in X, Y and Z directions
Weight	Approximately 160 g

USB: universal serial bus.

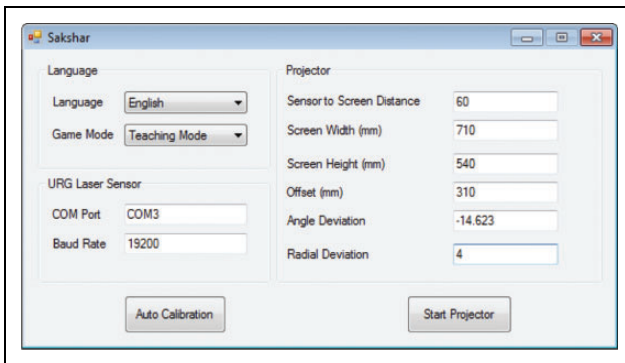


Figure 1A. Main GUI window in primary monitor. GUI: graphical user interface.

Appendix 2

Interactive game

Along with the hardware, we developed a game with audiovisual feedback to enable effective use of the interactive projective display system. This was designed to help children learn alphabets/mathematics, and so on. A game begins with a welcome screen where some control buttons are displayed on the projected screen. However, the main screen is displayed on the PC and can be used for device-level configurations (Figure 1A). The main screen can be used to set the language of the game displayed by the projector. In the projected screen, there are several control buttons, which are

‘Instructions’, ‘About’ and ‘Start’ (Figure 1B(a)), and so on. The game displays four different images, each corresponding to an alphabet. Out of the four available options, the player has to select one correct answer (Figure 1B(b)). Total points of the game are displayed at the bottom of the screen. The game also displays the time taken in seconds. Finally, the result is shown to the user at the end of the game (Figure 1B(d)). The game has two operating modes.

1. Teaching mode, which was designed for the children who are starting to learn alphabets. Corresponding to each alphabet, images of basic and well-known objects are displayed. Also, the user is given unlimited number of attempts for each alphabet. In this way, a user learns even though he/she makes mistakes.
2. Testing mode, which may be used once the user gets familiar with the alphabets and gains enough knowledge about them. This mode is used to test the knowledge. Testing mode contains multiple images for each alphabet, which exposes the user to more words corresponding to an alphabet. This mode does not provide unlimited attempts; instead, it has a countdown timer. Four points are given for the correct choice, whereas each wrong choice has a penalty of one point. In this way, learning outcomes can be measured. The ‘testing mode’ is explained in more detail in following the section.

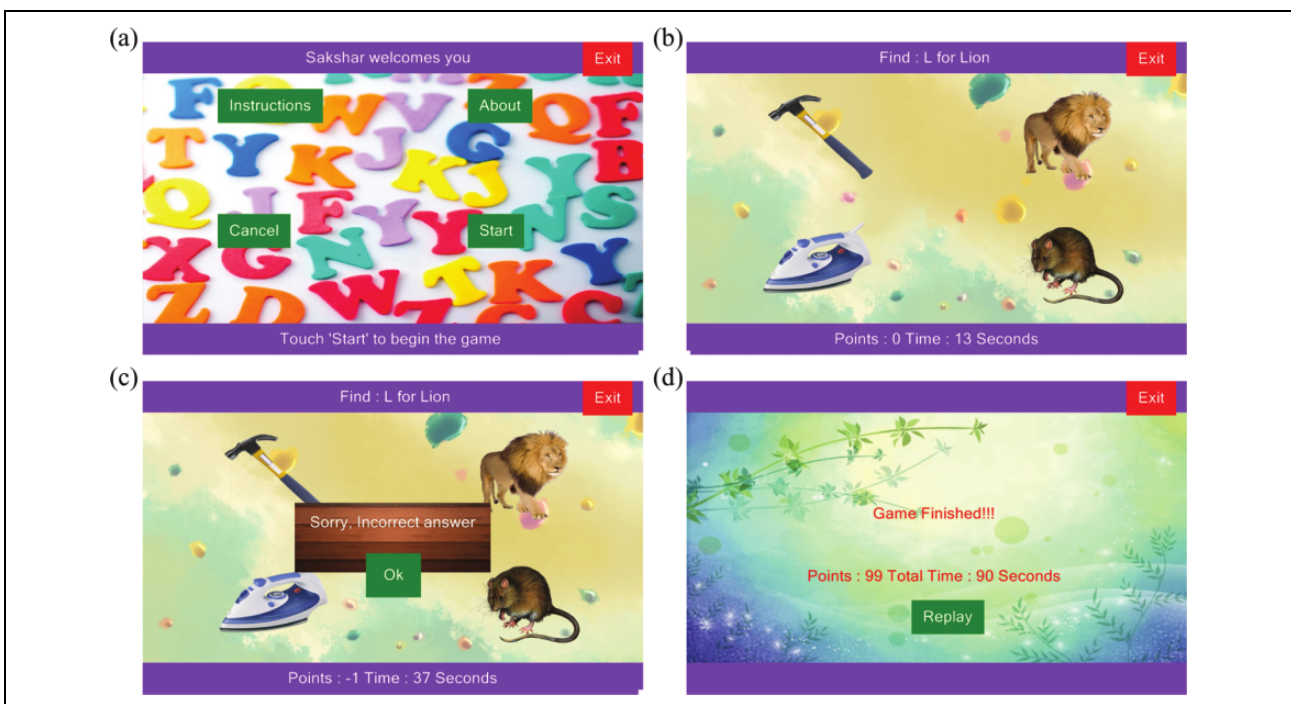


Figure 1B. Game screens: (a) Welcome screen, (b) alphabet considered, (c) feedback for incorrect answer and (d) final scores.

Algorithm 1. Generate random slides.

```

1: procedure SLIDEGENERATOR(array_alph, items_per_slide)
2:   Let array_alph be an array of characters to be used and items_per_slide is the number of choices to be displayed per slide
3:   Let array_temp be a new list of length  $\leftarrow$  array_alph.Length
4:   Let array_slides be a new list of length  $\leftarrow$  array_alph.Length
5:   for  $i = 1 \rightarrow$  array_alph.Length do
6:     first_char_per_slide  $\leftarrow$  getUniqueChar(array_temp) ▷ Get unique character from array_alph which is not present in array_temp
7:     unique_alph.Add(first_char_per_slide)
8:     while unique_alph.Length < items_per_slide do
9:       unique_alph.Add(getUniqueChar(unique_alph)) ▷ Get unique character from unique_alph and add it to unique_alph
10:    array_slides[i]  $\leftarrow$  current_slide(unique_alph)
11:    array_temp.Add(first_char_per_slide)
12:    unique_alph  $\leftarrow$  unique_alph.Empty
13:   return array_slides

```

Testing mode. The flow chart of the game is shown in Figure 2A. Once the game started, it displays a question to find out the correct image of an object. As an example, it asks to find 'L' for Lion (Figure 1B(b)). The game displays four available options and the player has to select the correct one. Four points are given for correct answer and one negative point for incorrect answer. The game is also having audio feedback which allows better interaction with the system. When a person selects an incorrect answer, a suitable message is shown on the display (Figure 1B(c)). This is reinforced by a sound. Same is the case for correct choice, which is rewarded with an appreciatory message and a suitable sound of applause.

The game algorithm takes care of generating random slides without any repetition. Slide starts from a random alphabet and covers all the alphabets. The slide generation algorithm takes an array of characters as input and returns a random sequence of slides, which has single correct answer among four available options. A function by name *getUniqueChar(list_of_chars)* was written, which recursively finds a single character from the main list of characters, keeping in mind that it should not be available in the *list_of_chars*. By following this procedure for each alphabet, we finally have all the slides ready. The pseudo code for generation of all slides is given in algorithm 1.

Game in hindi. While developing the game in Hindi, it was found that there is no suitable library to render Hindi fonts in OpenTK. Therefore, a library for rendering Hindi fonts in OpenTK was made. This library uses system. Drawing component from .NET framework to generate a graphics object, in which the text was written. Finally, this graphics object was converted to bitmap image and was rendered by OpenTK as a texture. This library enables us to render Hindi fonts in good quality. This library can be extended to various languages. We have tested it to render some of the Indian languages like

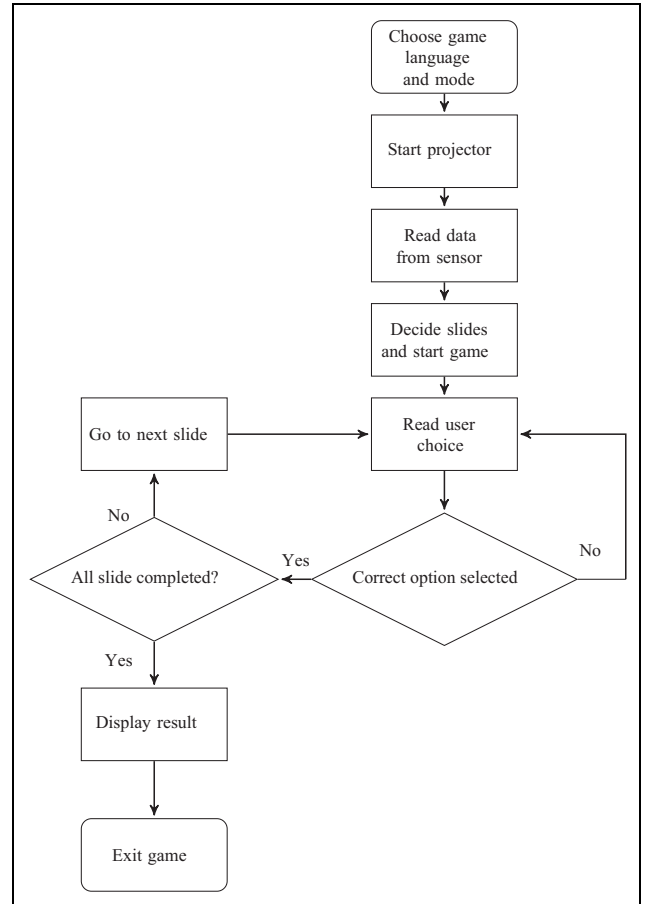


Figure 2A. Flow chart of the game.

Bengali (seventh largest group of speakers in the world), Malayalam and Urdu.²⁷ It can be designed for teaching various subjects. For example, we have developed a module for teaching mathematics also.⁵¹ Screenshots for Hindi and Mathematics version are shown in Figure 2B.

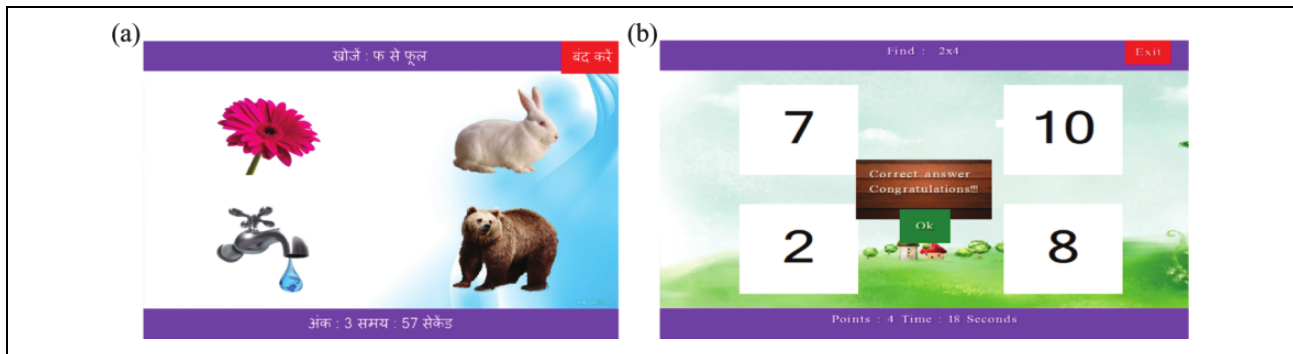


Figure 2B. Multiple versions of the game. (a) Game screen with Hindi alphabets and (b) game screen with mathematics.