Computer Vision Application using the Kinect Sensor for the Identification and Tracking of Users Interacting with a Surface Computing Platform

Eric R. Harvey[a], Jean-Nicolas Ouellet[a], Juan Echevarria[a], Glenn Franck[b], Stacey D. Scott[c]
[a]CIMMI, 2750 rue Einstein, Québec, Québec, Canada G1P 4R1; [b]Defence R&D Canada Atlantic, 9 Grove Street, PO Box 1012, Dartmouth, Nova Scotia, Canada B2Y 3Z7; [c]Systems Design Engineering, University of Waterloo, 200 University Ave W, Waterloo, Ontario, Canada N2L 3G1

RÉSUMÉ

Les environnements collaboratifs de type tables interactives connaissent un intérêt grandissant ces dernières années. Dans ce type d'environnement, les usagers interagissent physiquement avec la surface de la table soit en utilisant leur main ou un stylet. Dans ce contexte, la gestion des identités de plusieurs personnes interagissant autour de la table et leurs droits d'accès aux informations affichées est problématique. En effet, bien que des méthodes de reconnaissance de visages existent pour reconnaitre l'usager, il est difficile d'établir le lien entre une main interagissant avec la table et un usager en particulier. Afin de résoudre cette problématique, nous proposons d'utiliser le capteur de mouvement Kinect de Microsoft afin de déterminer l'identité des usagers interagissant avec la table. Pour se faire, nous exploitons la carte de profondeur retournée par le capteur pour détecter et suivre chaque usager. Le capteur nous renseigne sur la posture de l'usager en déterminant, entre autres, la position des bras et de la tête. En appliquant un algorithme de reconnaissance faciale à l'image 2-D de la caméra visible de la Kinect, il est alors possible de déterminer l'identité de l'usager interagissant avec la table. Nous présenterons les résultats de ce travail de recherche financé par R&D pour la Défense Canada (DRDC) Atlantic, en collaboration avec le département d'ingénierie systèmes de l'Université de Waterloo.

Mots clés: Vision numérique, Kinect, reconnaissance faciale, table interactive, environnement collaboratif

ABSTRACT

Multi-user computing platforms to support collaborative interaction have been of growing interest for the past few years. An interactive tabletop computer provides one such platform, on which users interact directly with the physical tabletop using their hands or a digital pen. In this environment, the identity management of several people interacting around the table and their authorized access to displayed information is problematic. Facial recognition methods exist for performing this task, but it is difficult to link one hand interacting with the table with a particular user. To address this problem, we propose a design that uses the Microsoft Kinect motion capture sensor to identify users interacting with the tabletop. We use the depth map provided by the sensor to detect and track the users. The Kinect is used to determine the user position, along with the position of the user's arms and head. By applying a facial recognition algorithm to the 2-D image provided by the visible camera of the Kinect, it is possible to determine the identity of the user interacting with the tabletop. This work is presented within the framework of a research project funded by Defence R&D Canada (DRDC) Atlantic and in collaboration with the Department of Systems Design Engineering of University of Waterloo.

Keywords: Robotic vision, Kinect, face recognition, surface computing, tabletop computer, collaborative environment

1. INTRODUCTION

In high-value, emergency, or crisis situations, having intuitive and sharable access to complex, dynamic data is often essential to timely and effective decision-making. Large-scale, interactive displays that support multi-user collaborative interaction, such as interactive tabletop computers, are becoming increasingly common in such complex environments. Such systems enable groups of people to interact with and share dynamic, digital data using a direct, intuitive interface. However, the sharable nature of these displays introduces inherent user authentication and security concerns when dealing with sensitive data. This paper describes an initial investigation into user identification and tracking in a multi-

user tabletop environment. To provide context for the project, we first provide some background on tabletop computers and the overall project framework under which this work was performed. We then further describe the specific problem addressed by this research.

1.1 Surface Computing and Tabletop Computers

Surface computing is the term for the use of a specialized computer in which traditional graphical user interface elements like a keyboard and mouse are replaced by the user interacting directly with a touch-sensitive screen [1]. Optical sensing and computer vision techniques can also be used for interacting with a screen. When a system is composed of a computer and an interactive surface configured like a desk table, we call the device a tabletop computer. Since the early work of Pierre Wellner on his interactive DigitalDesk [2], many surface computing devices have been developed for operating in our working environment [3]. The applications in which tabletop computers could be used are numerous in the field of human-computer interaction, but the present research project is mainly related to time-critical contexts such as the military [4].

1.2 Military Application

Defense R&D Canada (DRDC) Atlantic has been investigating the use of surface computing for use in a naval environment, and has now developed a prototype application showing some application of this technology. In this military environment, user security levels are of great importance in determining access to information, application privileges, permission overrides, etc. For the prototype tabletop application in use at DRDC Atlantic [5], security is managed through the use of digital pens, but the users of these pens still need to be authenticated so that permissions can be tied to the pens.

The concept behind the tabletop prototype is to have a basic map display system, capable of showing and editing ship tracks, and supporting data input from an arbitrary data source. Track histories can be shown, and reports can be queried to get more information to help establish the recognized maritime picture (RMP). It is designed to showcase the manner in which relevant maritime data can be accessed and shared in a collaborative environment [5]. Figure 1 shows pictures of the tabletop and its user interface similar to the prototype at DRDC Atlantic.

The application prototype is designed to enable collaborative exploration of a dynamic maritime tactical picture and of related information sources. The application software provides standard operator access to map and track data capabilities. It also provides an intuitive, direct-touch interface (via digital pens) that supports both individual and shared access to geospatial and other key mission-related information and media.

Figure 1: Tabletop system and its user interface (from [5]).

1.3 Challenge for Defense and Security

In the context of defense and security applications, the collaborative system must enable identification tracking/filtering of personnel and inputs – for example providing different functionality for different users who have various security levels or responsibilities. The identity management of several people interacting around the table and their authorized access to displayed information is provided through non-tethered digital pens, which use Bluetooth communications. With users of various security levels and responsibilities coming and going from the tabletop, picking up and putting down pens or walking away and then returning, manual authentication can become cumbersome. This is why a means of automatically linking a user's hand to a pen could secure the collaborative interaction process and would provide a preferred means of user privileges management.

The remainder of this paper describes the overall approach taken by the project, including the project requirements, background investigations, and an overview of the project technologies. We then present the developed system and its components, followed by a discussion of the preliminary results from performance testing conducted on the system.

2. APPROACH

To address the problem of identifying users interacting in a collaborative environment, we investigated the use of the Kinect motion capture sensor from Microsoft. The depth map provided by the sensor is used for detection and tracking of the users. With the Kinect, the user position is determined together with the arms and head position. By applying a facial recognition algorithm to the 2-D image provided by the visible camera of the Kinect, the identity of the user interacting with the tabletop is determined.

The first step of this approach consisted of defining the requirements for a tabletop computing system in terms of the interaction of the users with the system. The second step consisted of identifying various open source software development kits (SDKs) to provide Kinect sensor interfacing functions and facial recognition algorithms. The third step consisted of producing a software library of subroutines and developing a client application for demonstrating the capability and robustness of the recognition functionality. The application was developed using the C++ programming environment running on a Windows 7 operating system.

2.1 Analysis of Requirements

The initial requirements were defined together on the basis of the following assumptions:

- System detection of new users is continuous;
- Assumes a one-interaction device-per-user paradigm for a tabletop environment;
- Once a user is detected and identified, an interaction device is associated to the user;
- Once the device association process is complete, the system will continue to track the user's general location, and occasionally revalidate the device-user profile association based on nearest tracked user;
- Based on updated user location information, an interaction device is automatically re-associated from one user to another, whether to a new user or an existing user.

Figure 2 presents the list of requirements which have been specified for guiding the system definition and development. These initial set of requirements are specifically targeted to the prototype tabletop developed at DRDC Atlantic.

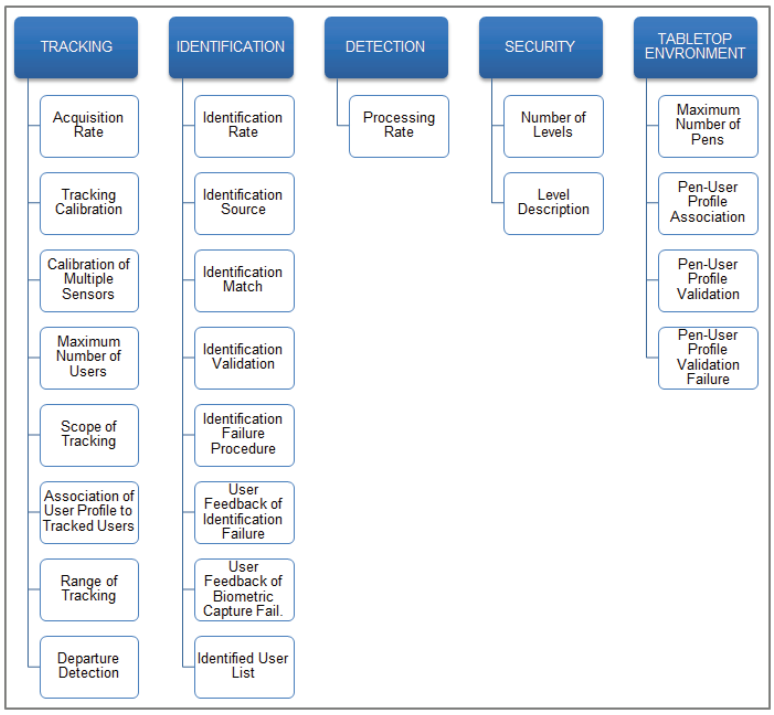

Figure 2: Set of initial system requirements.

2.2 Open Source SDKs

The main goal of the project was to investigate the feasibility of using the Kinect for automatic identification of users with the use of existing algorithms and functionality, whether contained within the Kinect hardware and SDKs or utilized from an existing third-party library. Thus, the availability of ready-to-use software code for the Kinect and facial recognition was explored in various open source libraries. Among the SDKs reviewed, the libraries listed in Table 1 for the Kinect and for facial recognition were chosen.

Table 1: Selected SDKs.

Library	Utilization	Internet Link
OpenCV and Haar	Face detection	http://opencv.willowgarage.com/wiki/ http://opencv.willowgarage.com/wiki/FaceDetection
FaceL	Eyes detection	http://sourceforge.net/apps/mediawiki/pyvision/index.php?title=FaceL:_Facile_Face_Labeling
Fisherface (Bytefish)	Facial recognition	https://github.com/bytefish/opencv/tree/master/lda
Kinect for Windows (version 1.0.3.190)	Kinect interfacing	http://www.microsoft.com/en-us/kinectforwindows

2.3 Kinect Technology

Kinect is a motion sensing input device manufactured by Microsoft for the Xbox 360 video game console. The device features one color camera and a depth sensor composed of an infrared laser projector and an infrared camera and also multi-array microphones running proprietary software, which provides full-body 3D motion capture, facial and voice recognition capabilities. The Kinect sensor, as shown in Figure 3, works at a frame rate of 30 Hz. The main technical characteristics of the Kinect sensor are listed in Table 2.

Figure 3: External (top left image) and internal (top right image) views of the Kinect sensor and its zoom lens kit (bottom image) [6].

Table 2: Characteristics of the Kinect sensor [7].

Parameter	Value
Operating range	Absolute max: 0.7 – 6m; effective: 1.2 – 3.5m
Field of view	43deg. Vertically 57deg. Horizontally
Motorized pivot tilt	+/- 27 deg.
Data streams	320x240 16-bit depth @ 30 frames/sec 640x480 32-bit colour @ 30 frames/sec

The basic configuration of the Microsoft Kinect provides the following features and functionalities:

- Skeletal tracking (can track up to two people and up to six people without skeletal tracking enabled)
  - Initialized automatically;
  - Continuous (at the camera frame rate).
- Advanced audio capabilities
  - Noise suppression;
  - Echo cancellation;
  - Beam formation to identify sound source direction.
- Access to raw data streams
  - Depth image;
  - Color image;
  - Four-element microphone array.

A Nyko Zoom Lens Kit Adapter (Figure 3) can be used for adapting the Kinect to the limited space of the tabletop. Nyko is a third party manufacturer [8] whose zoom lens kit adapter can expand the field of view (FOV) by 40% and shorten the working range. This device allows working at a shorter range so that the Kinect used in a smaller space, like the one we can expect in a limited volume of a tabletop computer.

However, the most recent version of the Kinect for Windows sensor, released after this work was complete, provides enhanced sensor capabilities and features a near mode, which enables the depth camera to see objects as close as 40 centimeters in front of the sensor [9]. Thus, the lens kit may not be required in the future.

2.4 Multi-Kinect Interfacing

A constraint on this initial investigation was to identify and track a maximum of four users. Ideally, the system would use more than one Kinect sensor to track multiple users simultaneously. In theory, up to six users in the field of view will be reported in the depth stream. With the current Microsoft SDK, of the six reported, only two users are actively tracked with skeletal data. The Windows SDK (official 1.0 release) supports up to four Kinect sensors plugged into the same computer, assuming the computer is powerful enough and they are plugged in to different USB controllers, providing sufficient bandwidth [10]. The skeletal tracking can only be used on one Kinect per process. If one wants more users to be tracked, it would require the networking of computers as shown in Figure 4, for synchronizing the tracking and identification tasks, which has not been done in the current library.

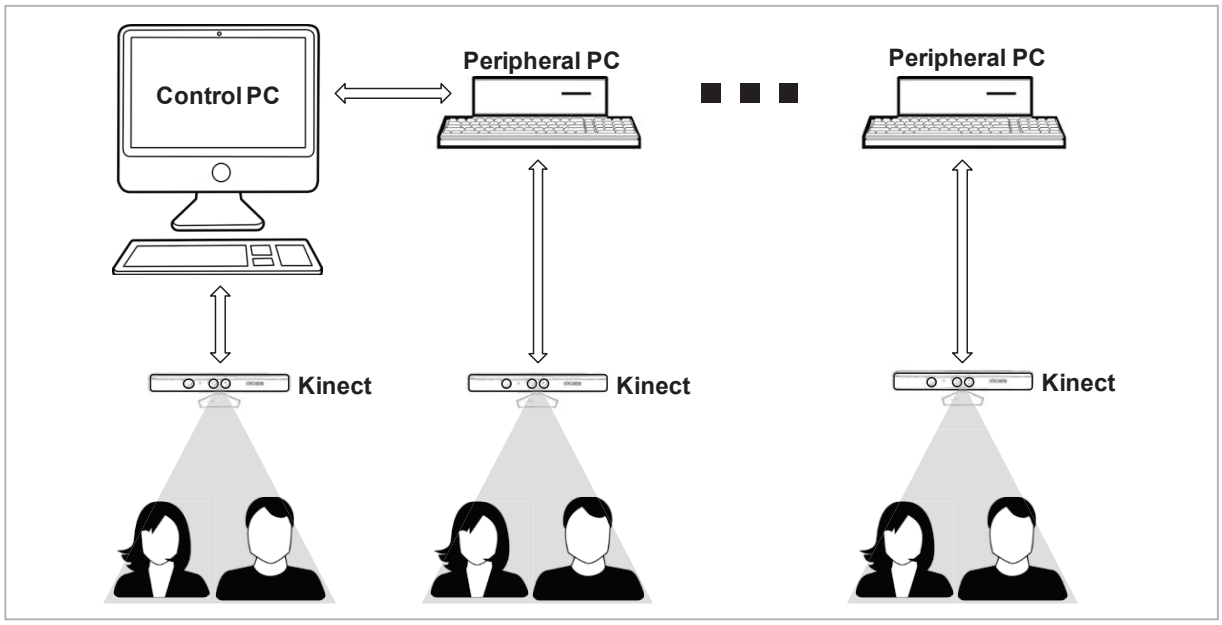

Figure 4: Possible configuration to enable skeletal tracking of more than two users.

2.5 Skeletal Tracking

For gesture-driven applications, the Kinect offers the capability to track the skeletal image of one or two people within its field of view. The motion sensor tracking is based on a recognition method, which quickly and accurately determines 3-D positions of body joints from a single depth image [11]. The Kinect automatically provides the body joint positions in space. The positions can be used to link for instance the head with the hand of a particular user. This tracking capability is integrated and operational in the client application developed for this project.

The tracking function does not need to be developed and is directly provided by the Microsoft Kinect SDK. Figure 5 shows the skeleton tracking capability in action.

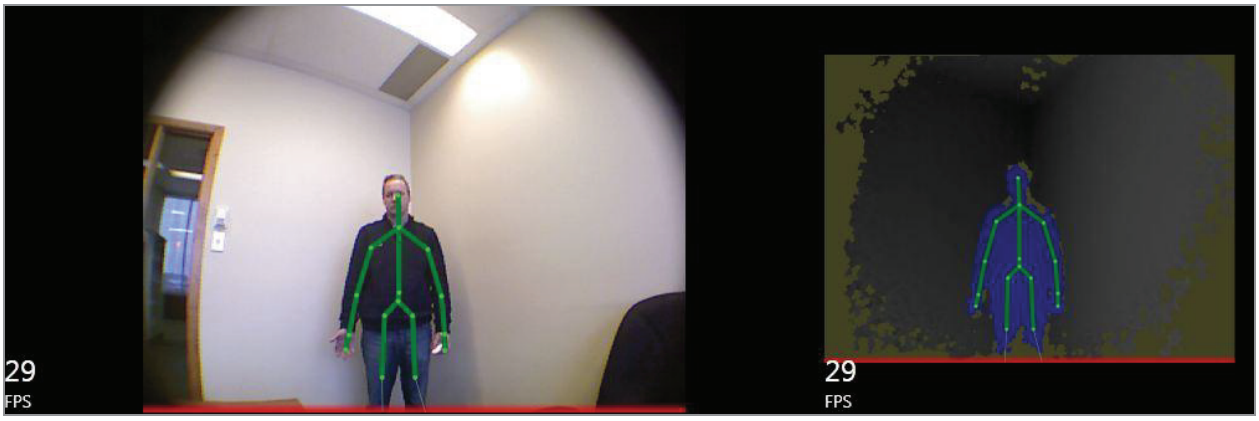

Figure 5: Skeleton tracking with the Kinect using the Nyko zoom lens adapter. Left: RGB camera; Right: depth map.

2.6 Face Detection and Recognition

To design the proper algorithmic library, it is necessary to understand the specific process of user identification using facial recognition. Figure 6 presents the steps: camera acquisition, face and eyes detection, face image rectifying, image normalization and recognition. The last step involves the use of a pattern recognition algorithm identified in the survey of open source SDKs. The pattern recognition method selected is called Fisherface. It is a set of mathematical linear discriminants, which is defined in a subspace representing a large set of images depicting different human faces. These vectors are included in a training database containing a large number of datasets used for discrimination analysis of many pictures of faces.

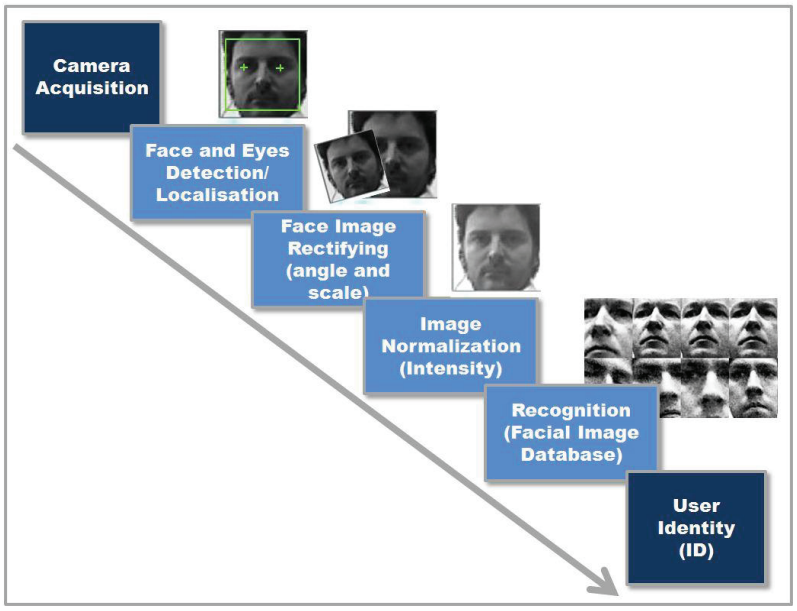

Figure 6: Process of identification using facial recognition.

Based on the SDK review, the OpenCV library was selected for this project. This open source library provides most of the functionality for implementing facial recognition. The facial detection and recognition process was implemented with Visual Studio 2010 in the C++ programming language using the OpenCV library (v2.3.1).

3. DEVELOPING A COMPUTER VISION SYSTEM FOR THE IDENTIFICATION AND TRACKING OF USERS

3.1 System

The system designed for automatic user recognition system is named SITU (System of Identification and Tracking of Users), which consists of using four Kinect sensors fixed on the four sides of an interactive tabletop device (Figure 7). They are oriented with an upward angle and observe the opposite side of the table. Under this configuration, it is assumed that each sensor will have at most four users simultaneously in its field of view.

SITU comprises the following components: the user tracking and recognition modules, the multi-user environment application and the user database module (Figure 8).

Within the framework of the present project, the SITU demo application has been designed specifically to be operated in a standalone mode, i.e. it does not need external applications to be used. The SITU library could be integrated to other types of applications (other than a collaborative environment).

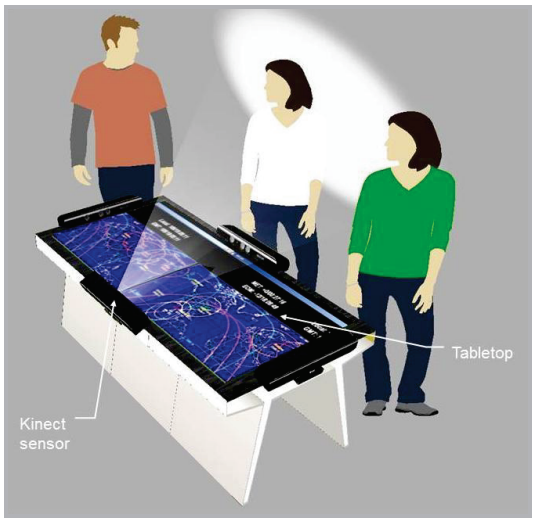

Figure 7: Conceptual view of SITU system.

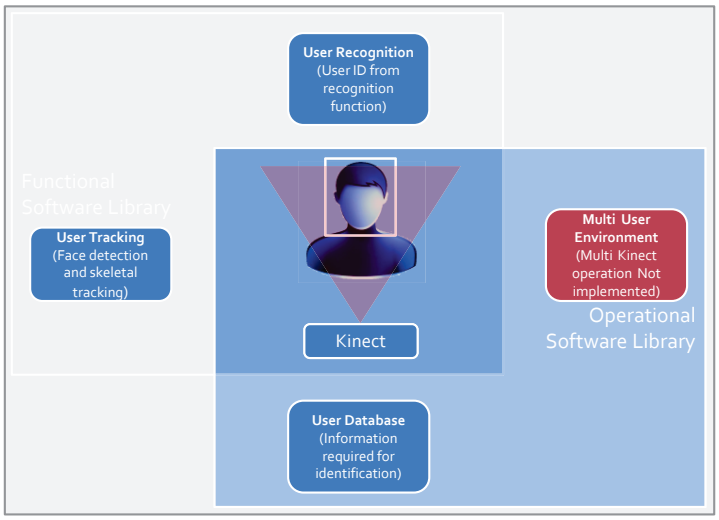

Figure 8: High level system diagram.

3.2 Software

A client application of SITU has been developed to demonstrate how the library can eventually be integrated to another application such as the tabletop computing prototype at DRDC Atlantic. This application has been used also for demonstrating the capability and robustness of the recognition functionality. It has been developed using the C++ programming environment running on a Windows 7 operating system.

The main interface window of the application (shown in Figure 9) displays the image seen by the RGB camera of the Kinect (converted to grey scale to reduce bandwidth). Information data are superimposed on the image, presenting special features associated with detection, identification and skeleton data. Figure 10 shows a diagram which describes the information displayed on screen.

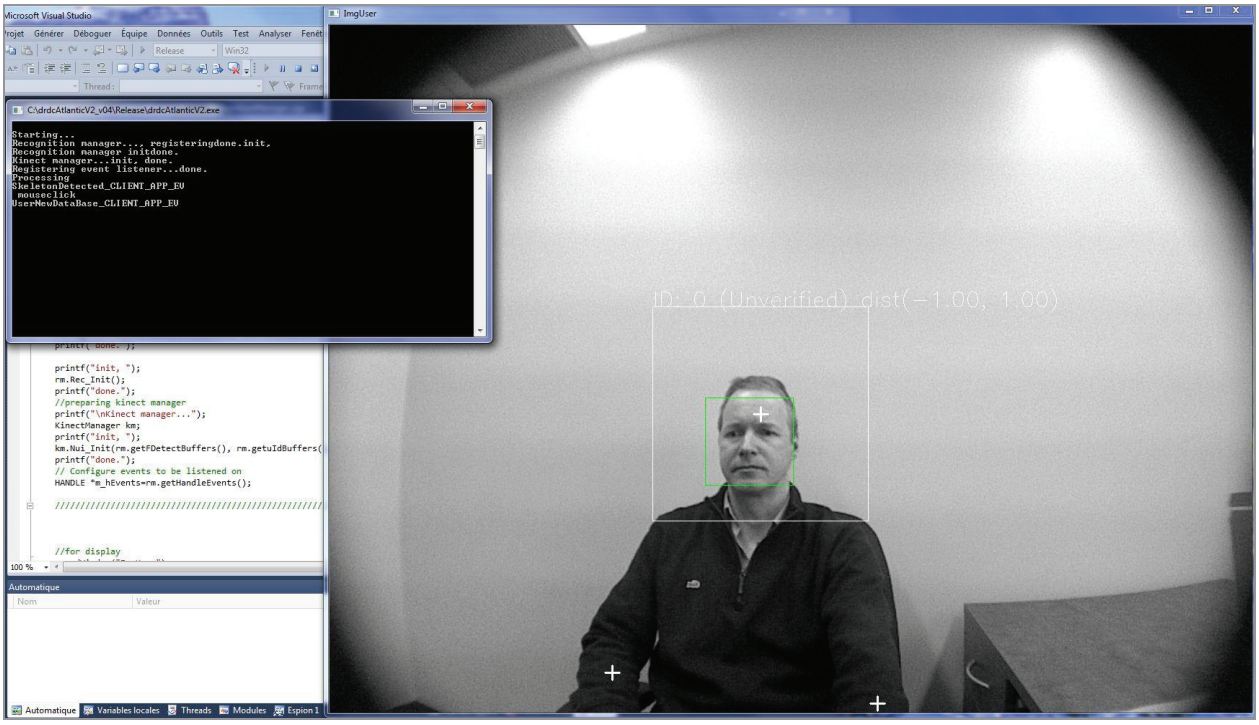

Figure 9: Client application interface display.

A static library has been written in C++, based on the SDK chosen from the reviewed open source libraries. The library is structured for enabling the following functions:

- User information management: manages the information of the user (identification number-ID, image, security level);
- Eye detection: performs the detection of the center of each eye;
- Face database management: manages and stores the face images in the database;
- Face detection: performs the extraction of the face inside an adapting window;
- Face training and recognition: performs the preparation of a training set of images of normalized and scaled faces;
- Kinect interfacing: communication with the Kinect hardware;
- Skeleton ID management: manages the skeleton information provided automatically by the Kinect sensor;
- OpenCV interfacing: interaction with the OpenCV programming environment.

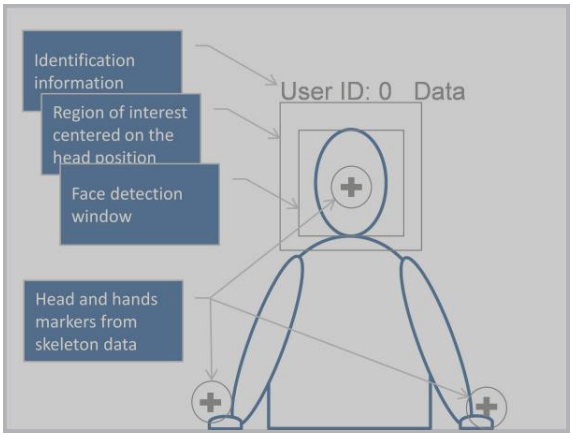

Figure 10: Displayed information.

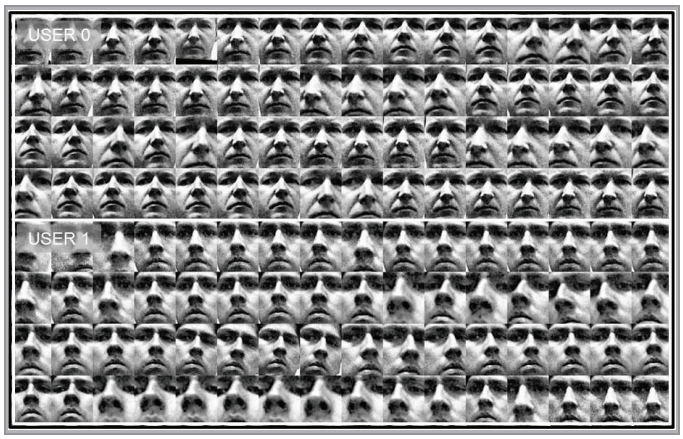

Figure 11: Display of faces used in the database.

The library acquires a user's face in the database when a body is detected and its skeleton is actively tracked. Once 64 images are stored in memory, the client application is notified that enough images are accumulated and it waits for a confirmation. Figure 11 shows the accumulated images of the faces of two users in the database. The first set of 64 images corresponds to user 0, the second set to user 1 and so on.

3.3 Current Set-up Prototype

A MacBook Pro computer (processor: Intel Core i7-620M; graphics card: NVIDIA GeForce GT 330M) operating under Windows 7 was used for running the software application. The Kinect was connected to the computer via one of the USB ports. The Kinect device was attached to a table tripod for a better stability and flexibility in positioning and orientation. The Nyko zoom lens adapter was installed on the Kinect. Figure 12 shows a photograph of the set-up. The image on the main window has a fish-eye lens effect due to the lens adapter.

Figure 12: Set-up composed of a portable computer and the Kinect.

4. PRELIMINARY RESULTS

Within the framework of this research project with DRDC Atlantic, one of the main goals was to develop a demonstration prototype application showing how to use the code library, which has the ability to recognize the identity of multiple users and track them simultaneously, using only the Kinect sensors as input. The focus of the experimentation effort was on the assessment of the recognition capability and robustness of SITU system. Due to project constraints, the current system was not integrated to the existing tabletop application in use at DRDC Atlantic.

A basic test plan was defined and emphasized the following aspects: head and face detection and user recognition.

Head and Face Detection

This aspect of the test plan tried to verify the detection under various lighting conditions and for different translation and rotation of the head.

Under directional lighting conditions, the face detection performance degrades substantially when the illumination is from above the head, producing deep shadow in the eyes area and decreasing the performance. It even prevents the detection of the center of the eye. This shows that a proper uniform illumination of the face is an important performance factor.

The system has been tested at different ranges representative of user distances in the tabletop environment, i.e. when someone is not interacting and far (more than one meter) from the tabletop or actively interacting and close to it. Tests show that the detection performance of the face was not degraded inside a selected range (55 to 95 centimeters from the Kinect device).

The limits of the angular orientation of the face were also tested with the system. Detection of the head moving in planes parallel and perpendicular to the sensor plane was performed. The head was tilted until the system was not able to track the face. Results show that if the head is tilted in a plane parallel to the Kinect sensor plane, the range extends from -10 deg. to +15 deg. (Figure 13). These results show the flexibility and the limitations of the system when submitted to expected user behaviour in a collaborative environment.

Figure 13: Face detection limits at distance d from the Kinect sensor and at head angle α parallel to the sensor plane.

User Recognition

A specific procedure was defined for this test to verify the recognition function in terms of consistency of the results (Table 3). Tests #1 to #3 consisted of static test conditions, in which the user did not move at all in the field of view. For tests #4 to #6, the user was free to move slowly or quickly in the field of view, simulating a representative activity in a collaborating environment. Figure 14 shows two users successfully identified at the same time in the field of view of the Kinect.

Table 3: Test procedure for recognition (Pre-condition: 2 users in the database (ID0, ID1).

Test #	Description
1	1 static user recognized (ID0)
	1 static user recognized (ID1)
3	2 static users recognized
	1 dynamic user recognized (ID0)
5	1 dynamic user recognized (ID1)
	2 dynamic users recognized

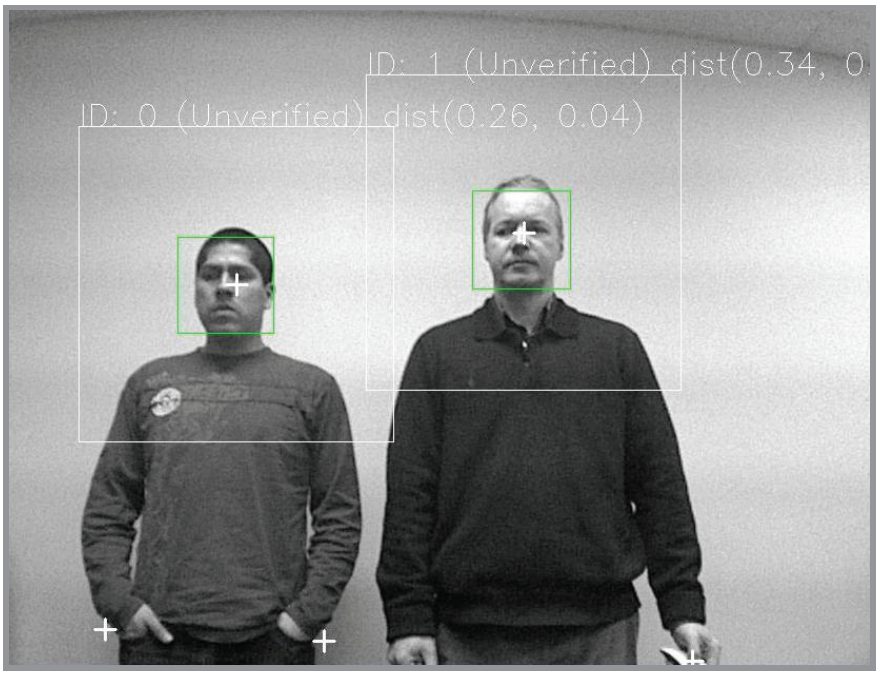

Figure 14: Test of the recognition function with two users in the field of view.

The test results consist of a series of identification parameters generated by the SITU software. Parameters such as time, user ID (presented just as unique numbers: 0, 1, etc.), recognition status and a distance parameter were extracted. This distance parameter is related to the classifying functions in the linear subspace of the face images (a value corresponding to a match with a user in the database).

The performance evaluation of recognition methods is a very difficult one and should be based on criteria which are generally accepted in the biometric science community. The following definitions are considered for the present discussion:

Detection rate: defined as the ratio between the number of faces correctly detected and the number of faces determined by a human;
False positives: in which an image region is declared to be face, but it is not;
False negatives: in which faces are missed resulting in low detection rates.

As expected, the recognition performances are very dependent on the tuning of the classification parameters in the code. Each test was performed for a short period of time (30 sec.). Preliminary results show a fairly good detection rate and low false positive errors, when the database is limited to two users. If the number of users in the database is more than two, then the false positive errors increase as the result of imperfect parameter tuning, leading to confusion in the recognition process. The performance of the face recognition method used is obviously dependent on the number of users saved in the database.

5. CONCLUSION

The work presented here consisted of a preliminary investigation using the Kinect sensor capabilities for user detection and identification within the context of collaborative surface computing. Its capability of providing depth/2-D imaging and skeletal extraction allows it to perform automatic identification and tracking functions for users interacting in a collaborative tabletop environment. This research work demonstrated that the Kinect can be used and integrated to a system dedicated to automatic user identification and tracking.

However, as a first step, the project contained constraints that limit its generalizability. Further work is warranted to fully understand the potential of using the Kinect sensor platform for user identification and tracking in tabletop computing and other interaction environments. Possible future directions to the project include the following:

- Identification and testing of other facial recognition methods to improve system performance;
- Optimization of the system software and of the face illumination conditions to produce more reliable results. The illumination environment around the tabletop can be easily controlled by adding diffusing light projectors;
- Expanding the library to support multiple Kinect over a common work space;
- Integrating the SITU system into the existing tabletop application at DRDC Atlantic and validating the system in a real collaborative environment.

This initial work produced very promising results with the use of a new, captivating and very affordable imaging technology originally developed for entertainment, which shows great potential for the development of high level biometric security applications.

REFERENCES

[1] Wikipedia: http://en.wikipedia.org/wiki/Surface_Computing.

[2] Willner, P., "The DigitalDesk calculator: tangible manipulation on a desk top display", UIST '91 Proceedings of the 4th annual ACM symposium on User interface software and technology, New York, NY (1991).

[3] Benko, H., "Beyond flat surface computing: challenges of depth-aware and curved interfaces", MM '09 Proceedings of the 17th ACM international conference on Multimedia, New York, NY (2009).

[4] Gouin, D., Lavigne, V., "Trends in human-computer interaction to support future intelligence analysis capabilities", 16th Command and Control Research and Technology Symposium International "Collective C2 in Multinational Civil-Military Operations", Quebec City June 21-23 (2010).

[5] Scott, S.D., Allavena, A., Cerar, K., Franck, G., Hazen, M., Shuter, T., Colliver, C., "Investigating Tabletop interfaces to support collaborative decision-making in maritime operations", Proceedings of ICCRTS 2010: International Command and Control Research and Technology Symposium, Santa Monica, CA, USA, June 22-24 (2010).

[6] ROS (Robot Operating System): http://www.ros.org/wiki/kinect_calibration/technical.

[7] Wikipedia: http://en.wikipedia.org/wiki/Kinect.

[8] Nyko Technologies: http://nyko.com/products/product-detail/?name=Zoom.

[9] Kinect for Windows: http://www.microsoft.com/en-us/kinectforwindows/discover/features.aspx.

[10] Kinect for Windows: http://www.microsoft.com/en-us/kinectforwindows/develop/release-notes.aspx.

[11] Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., Blake, A., "Real-Time human pose recognition in parts from single depth images", IEEE Computer Vision and Pattern Recognition (CVPR), June 21-23 (2011).