Dynamic Sensor Selection Based on Joint Data Quality in the Context of a Multi-Kinect Module inside the CAVE "Le SAS"

S. Salous, J. Newton, L. Leroy, and S. Chendeb

***Abstract*—In order to create interaction, immersive VR rooms can receive various types of inputs from a user. One of these input types are the gestures and movements of the user in the CAVE, which can be captured with sensors such as the Microsoft Kinect. However, the large scale of an immersive room implies the use of multiple Kinects to provide optimal coverage. This multi-Kinect set-up requires an effective selection method to determine the most accurate Kinects depending on the user's position. The example of VR environment described in this paper is a CAVE named "Le SAS" that currently supports 4 Kinects. This paper describes the SAS's features as well as the constraints the implemented solution had to take into account. It will justify the technical choices and provide experiment results.**

***Index Terms*—CAVE, Kinect selection algorithm, multi-Kinect.**

I. Introduction

The Microsoft Kinect is a sensor embedded with a color camera that returns an RGB video stream and an IR camera that measures the visible objects' depth. Since its commercial release in 2010, this device has attracted interest from the community due to being a low-cost VR sensor. Intended to be used as a game controller for the Microsoft XBOX360, the device's accessibility, compared to other similar sensors, led to its use in other kinds of applications, such as user-tracking in a CAVE. In this environment, gestures of the user can also be tracked as an input scheme.

In this paper we use multi-Kinects system in order to properly track the user and its gestures in real time inside the CAVE. Multi-Kinects have advantages such high coverage area and easy setup. Also, in our solution we need to determine in real time which of the four sensors have the most relevant data in every frame.

This paper is organized as follows. Section II highlights the related work and various camera selection methods. Section III describes virtual reality cave "Le SAS". Section IV describes the algorithm that is the most suited to the context and the environment. Section V describes experiment set-up for our CAVE. Section VI concludes on the algorithm choice and the obtained data.

II. Related Work

Selection and classification algorithms have already been discussed and used in previous research works. Thrun et al. [1] relied on a real-time version of the Expectation Maximization (EM) Algorithm to generate 3D representations of the inside of buildings from images and measurements taken by mobile robots. They present algorithm for recovering 3D models from camera and manipulate prior knowledge on the shape of basic building elements. Also they fit a probabilistic model that consists of large rectangular, flat surfaces to the data collected by a robot. They design generative probabilistic model that consists of four sections, world model, measurements, correspondences and measurement model. In work model they represent non-flat surfaces by small polygons and flat surfaces by rectangular surfaces for representing doors, walls and ceiling. In measurements are we using a laser range finder that each range is projected into 3D space. In correspondences they use an efficient algorithm for environment mapping to make explicit relation between individual measurements. The measurement model match between volumetric and the measurements where the measurement model is generative probabilistic model of the measurements. They propose EM for likelihood maximization which considers a popular method for hill climbing in likelihood space, EM starts with two steps which are E-step and M-step. E-Step applies Bays rule applied to the sensor model which permit to calculate the desired expectations. M-step passes through certain set of calculations to determine some parameters that's important for principal orientation and location of the rectangular surface without the surface boundary. Kushwaha *et al.* [2] also used EM algorithms for multi-target tracking in a multimodal sensor system. This system could track an object through audio and visual sensors. They use Markov Chain Monte Carlo Data Association (MCMCDA) algorithm for tracking that avoids enumeration of tracks. Also, (MCMCDA) can tracks unknown number of targets in noisy urban environment. They aims to track moving vehicles emitting engine noise which include system components including audio processing, video processing, WSN middleware services, multimodal sensor fusion, and target tracking based on sequential Bayesian estimation and MCMCDA. For audio they implement beam forming technique on audio sensors utilizing an FPGA-based sensor board and evaluated its performance as well its energy consumption. For video they use a standard motion detection algorithm on video sensors, we have implemented post-processing filters that represent the video data in a similar format as the audio data, which enables seamless

Manuscript received March 14, 2015; revised August 12, 2015.

The authors are with the Paragraph Laboratory, Universit éParis 8, France (e-mail: saleh.salous@citu.fr, soosaine@ece.fr, Safwan.chendeb@citu.fr, Laure.leroy@citu.fr).

audio-video data fusion. Levine *et al.* [3] detailed some applications of a variant of the EM algorithm called the Monte Carlo EM algorithm. They use Monte Carlo simulations to compute expectation in EM algorithm, take result of each iteration of Monte Carlo sample, then apply Monte Carlo EM algorithm through Markov chain Monte Carlo (MCMC) routines such as the Gibbs and Metropolis–Hastings samplers. They apply an automated rule for increasing the Monte Carlo sample size. EM provides a tool for getting maximum like hood equations under models that yield analytically formidable like hood equations. On the other hand, Gupta *et al.* [4] proposed another sensor shuffling technique that uses a stochastic sensor selection algorithm to select one sensor every time step among set of sensors because all sensors cannot operate simultaneously. Their algorithm differs than other algorithm, it based on the letting the sensors switch randomly according to certain optimal probability distribution to get the best expected steady-state performance. Also their algorithm can be applied to the problem of sender trajectory generation for optimal coverage of an area. This problem happens when are some specified numbers of mobile sensors that can reach sense over a limited region but together they must monitor a given area. Faion *et al.* [5] present a method to intelligently schedule a network of multiple RGBD sensors in a Bayesian object tracking scenario. The method also deals with multiple Kinects issues such as large amount of raw data generated by the sensors and interference caused by overlapping fields of view. They propose a new hardware that control IR-projector toggling and synchronize depth data stream with existing software as in Fig. 1.

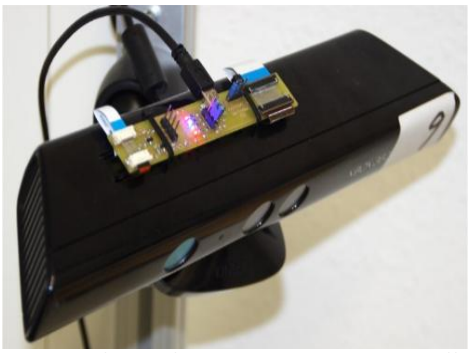

Fig. 1. Hardware modification for IR-projector-subsystem on/off-toggling of a Kinect device [5].

The proposed algorithm addresses these issues by selecting and exclusively activating the sensor. Other selection methods exist, based on specific criteria such as a relevancy level attributed to each sensor and is updated in real-time to determine the most accurate [6]. The Kinect's distance from the user is also a point of comparison for the selection.

The choice of camera selection algorithm depends on its efficiency in terms of accuracy but also of computational cost and algorithmic simplicity. For this reason, it is necessary to perform an analysis of the SAS to determine which method is more suited to the VR environment and the constraints that are related to the infrastructure.

In the context of a multi-Kinects system, this model can infer the Kinects which data is the most likely to be relevant by analyzing samples from previous user tracking data in the SAS.

III. Analysis of "Le SAS"

The "SAS" is an immersive room with a front screen and a floor screen. Both are 3 meters high and 4 meters wide as in Fig. 2. In our configuration, two Kinects are located on the top of the front screen and two other Kinects are located at the back of the floor screen. All four of them are pointed towards the middle of the floor screen, where is located the default position of the user which is supposed the most common position. This position is where the Kinect coverage is at its best [7].

Fig. 2. Representation of "Le SAS.

They propose an algorithm to overcome these problems by synthesizing the skeletons generated by duplex Kinects, which capture the human motion indifferent views. The algorithm is formulated under the constrained optimization framework by using the bone-lengths as hard constraints and the tradeoff between inconsistent joint positions as soft constraints. Extracted single view skeleton has problems such as self-occlusion, Bone-length variation and artificial vibration as in Fig. 3. Self-occlusion happens when some parts of skeleton are hidden from the camera, so, the depth value of a pixel will be missed. For bone-length-variation, they use segmentations algorithm to generate confidence-weighted proposals for the join positions. For solving single-view skeleton problems, they design a system with duplex kinect for motion capture. First camera faces the user which called principal camera and second camera. For reducing artificial variation can be reduced by y averaging the positions of joints reported by two kinects. But still the system has inconsistency due to overlapped regions on the human body. Another reason is the miss classification of regions in the 3D data obtained from a single-view. Another reason which inconsistent positions are estimated because it's not tracked. Artificial vibration is result of the acquisition error from camera which produces unwanted vibrations on the extracted joint-position and makes length of bones change during the motion [8]. They apply object recognition approach that produce designing an intermediate body parts representation that maps the difficult pose estimation problem into a simpler per-pixel classification problem. They use large set of highly varied training dataset allows the classifier to estimate body parts invariant to pose, body shape, clothing. Then generate confidence-scored 3D proposals of several body joints by reprojecting the classification result and finding local modes. They use consumer hardware which runs at 200 frames per second, this allow to show high accuracy on both synthetic and real test sets, and investigates the effect of several training parameters [8].

Fig. 3. Problems of skeleton tracking by single Kinect (Top-left) Self-occlusion, (Top-right) Bone-length variation [7].

When the user is in the middle of the floor screen, all of the Kinects can detect the skeleton and send its data to the VR application server. However, as soon as the user moves away from the center, the coverage is altered. For instance, the Kinects' FOV are not long enough to properly track a user located on the diagonally opposite side of the SAS. As a result, the number of relevant Kinects changes depending on the user's location.

Therefore, the selection algorithm will have to take into account the differences between optimal and sub-optimal coverage areas and adapt to the user's behavior as in Fig. 4.One of the main constraints of the virtual reality environment is its interactive real-time nature. As a result, the selection algorithm must be efficient enough to compute the Kinect data at a steady Kinect rate 30FS, with 5 millisecond timestamp difference between frames due to synchronization between Kinects. . Optimization of data processing may be required in order to provide a lag-free interactive experience for the user, for that, the algorithm sends only head joint data with status = 2.

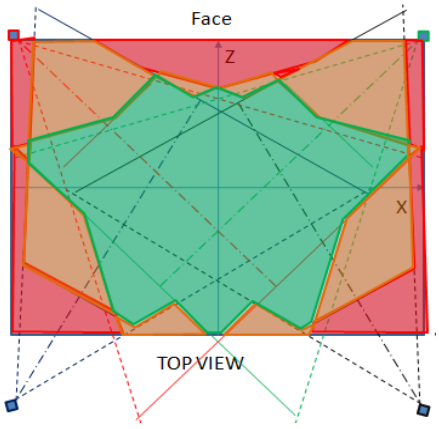

Fig. 4. Tracking level areas. [6]

IV. Sensor Selection Algorithm

After an overview of the various selection algorithms and an analysis of the constraints of our physical infrastructure and our technical environment, we decided to apply a simple algorithm that will extract the head joint data (position and orientation) from the Kinects that are the closest to the user. To achieve this, the depth coordinate of the joint returned by the Kinects are compared to each other. The selection algorithm collects the data from the 4 Kinects for the head and associates a status value to each for every skeleton joint head in every frame. This status value is 0 if the Kinect does not recognize a joint, 1 if it detects a joint but does not provide joint data (position and orientation), and 2 if the joint tracking's status is optimal. Based on this status values, the algorithm focuses on the joint data from the Kinects with the higher status value and sends it to the SAS's interaction server. When several Kinects track the user's joint data with excellent status on the same frame, one of the Kinects is randomly chosen.

The algorithm also provides a failsafe mechanism for frames where no Kinect returns an acceptable status value. In these situations, the data sent to the server is related to the previous frame where at least one Kinect's status was optimal. That means the user stands in dead area which is not detected by any Kinect. Fig. 5 shows overall kinect selection algorithm with failsafe mechanism.

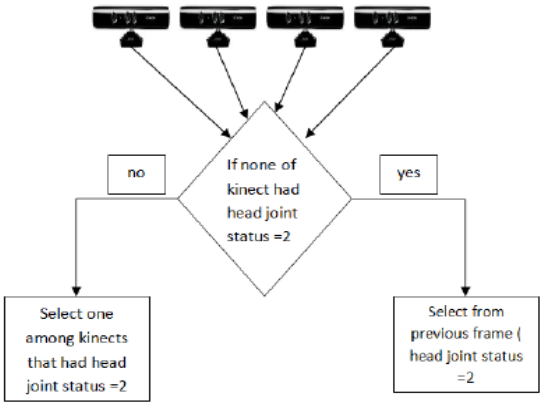

Fig. 5. Kinect selection algorithm.

V. Experiment

A. Experiment Setup

We installed the 4 kinects in the SAS as shown by Fig. 6 as explained in [6]. A user moved freely around the SAS without any equipment, guides or instruction for approximately 5 minutes.

B. Simulation

A user moved around the SAS, his skeleton was tracked by the sensors and his joint data was collected by our infrastructure. We chose to focus on head joint data for tracking purpose, and analyzed the data sent by the Kinects for this specific joint for every single frame.

Fig. 4 shows coverage areas inside the SAS, Green area related to optimal coverage area. Orange area related to sub-optimal coverage area. Red area or dead area which means all Kinects had a head joint status value of 0, due to IR interferences, noise between the sensors and dead areas as in Fig. 4.

Fig. 7 shows results of analyzed frames that collected during the simulation. On83.89% of analyzed frames, there was at least one of the four Kinects that returned a status head joint value = 2. On 52.56% of analyzed frames there was all Kinects that returned a status head joint value = 2. 15.09% of analyzed frames returned a status head joint value of 0 for every Kinect.

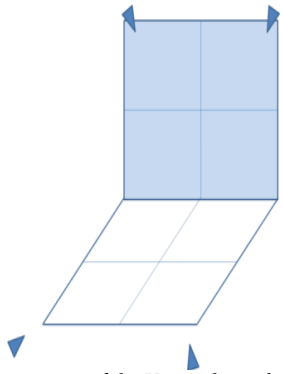

Fig. 6. Simple representation of the Kinect dispatching on the SAS [6].

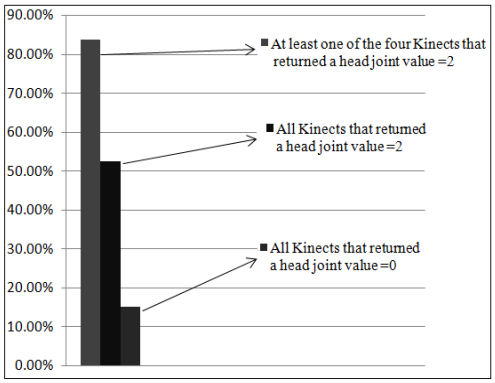

Fig. 7. Percentages for each area.

VI. Conclusion

In this experiment, we proposed a selection method for the multi-Kinect system in our CAVE. Our solution is easy to implement as it alleviates the issues related to the heaviness of multi-sensor calibration. The selection process bypasses this need to calibrate the entire system and chooses the optimal Kinect for user tracking in real time.

Acknowledgment

This research and its results are made possible thanks to the members of the CiTU-Paragraphe lab. Thanks to Taha RIDENE in particular for his previous work on the subject and his support and advice.

References

[1] S. Thrun, C. Martin, Y. Liu, and H. Dirk, "A real-time expectation maximization algorithm for acquiring multi-planar maps of indoor environments with mobile robots," *IEEE Trans. on Robitics and Automation*, vol. 20, pp. 433-442, June 2004.

[2] M. Kushwaha, S. Oh, I. Amundson, and X. Koutsoukos, *Handbook of Ambient Intelligence and Smart Environments*, 1st ed. Springer, US, 2010, ch. 2, pp. 117-147.

[3] R. A.Levine and G. Casella, "Implementations of the Monte Carlo EM algorithm," *Journal of Computational and Graphical Statistics*, vol. 10, no. 3, pp. 422-439, 2001.

[4] V. Gupta, T. H. Chung, B. Hassibi, and R. M. Murray, "On a stochastic sensor selection algorithm with applications in sensor scheduling and sensor coverage," *Automatica*, vol. 42, pp. 251-260, February 2006.

[5] F. Faion, S. Friedberger, A. Zea, and U. D. Hanebeck, "Intelligent sensor-scheduling for multi-kinect-tracking," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 3993-3999.

[6] S. Salous, T. Ridene, J. Newton, and S. Chendeb, "Study of geometric dispatching of four-kinect tracking module inside a Cave," in *Proc. 10th International Conference on Disability, Virtual Reality and Associated Technologies*, 2014, pp. 369-372.

[7] K. Y. Yeung, T. H. Kwok, and C. C. L. Wang, "Improved skeleton tracking by duplex kinects: A practical approach for real-time applications," *Journal of Computing and Information Science in Engineering*, vol. 13, no. 4, pp. 1-10, 2013.

[8] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *Proc. the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 1297-1304.

Saleh Salous was born in Kuwait in 1982. He received a MS degree from University of Jordan, Amman, Jordan in 2008. Now he is pursuing a PhD degree in computer science in Universit é Paris 8 and he has gained the bachelor degree in An-Najah National University in Plaestine. His interests are multi sensors control and data fusion.

Safwan Chendeb was born in Lebanon in 1977. He received a PhD degree from National School of Mines in Paris, 2007. Now he is the head of Partnership Service and Research Exploitation at Universit é Paris 8, Vincennes, Saint-Denis since November 2014. His interests are augmented reality, robitcs and CAVES.

Laure Leroy was born in Belgium, 1981. She received PhD degree from National School of Mines in Paris in 2008. Now she is a lecturer at University Paris VIII Vincennes - Saint-Denis since 2012. Also she is a member of AFRV from October 2011. Her interests are stereoscopic devices, virtual reality, immersion and interaction.

Julien Newton was born in Paris. He received the bachelor degree in litt érature and laungaes ET civilisation étrang ère anglais in 2012. Now he is a student in third year in informatique/computer. His interests are mobile and web applications.