

## Gesture-Based Control of Scanning Electron Microscopes Using Leap Sensors

S Cater<sup>1</sup>, BC Breton<sup>2</sup>, DM Holburn<sup>2</sup>, and NHM Caldwell<sup>3</sup>

<sup>1</sup>. BT plc, Adastral Park, Martlesham Heath, Ipswich, United Kingdom

<sup>2</sup>. Department of Engineering, University of Cambridge, Cambridge, United Kingdom

<sup>3</sup>. Department of Science and Technology, University of Suffolk, Ipswich, United Kingdom

This work forms part of our ongoing research into enhancing and improving SEM technologies, and was one of two projects initiated to explore the potential of gesture-based interface control. This project focused on gesture-based manipulation of the SEM.

The instrumental setup was a Carl Zeiss 1430VP SEM and a Leap Motion Controller (LMC). Carl Zeiss' Application Programming Interface to their SmartSEM software enables external programs to directly access and control instrument parameters. The Leap Motion is a consumer-grade motion capture device, which utilizes three infrared lights and two infrared cameras in a parallel configuration to create a stereoscopic capability. The sensors are directed in the Y-axis relative to the surface upon which the device is laid and have a hemispherical view of approximately 150 degrees from the top of the controller with a range of 25-600mm above the unit. The cameras can capture and process images at rates between 20-200 frames per second, transferring the resulting three dimensional point reference to a linked computer where they are analyzed for variances between the two camera perspectives. A three-dimensional image is created and a mathematical model applied by the Leap Engine to identify fingers, hands, and gestures, with additional mathematical analysis applied between Leap image frames to derive movement and direction. The Leap API exposes raw frame data as well as more refined information such as hand and finger positions, active movements and gestures. The project therefore involved building a bridge application to convert the gesture input (processed through the LMC and Leap Engine and obtained from the Leap API) into requests for instrument operations and parameter adjustments to be executed through the SmartSEM API.

Studies into using Leap Motion in other domains have raised concerns of inconsistency of sampling rate of the LMC, inaccuracies in gesture identification by the Leap Engine, and unanticipated and undesirable behaviors in the Leap API [1]. Whilst the device is able to track reasonably well in the absence of occlusion of hands and/or fingers, occlusion causes its mathematical model to overreach in its identification, generating incorrect frame data (and subsequent triggering of undesired actions) with the middle and little fingers being particularly problematic [2]. More generally, most three-dimensional gestures are actually two-dimensional gestures conducted in three-dimensional space and novice users struggle to learn three-dimensional semaphoric gestures [3].

The Leap Engine detects a relatively small set of predefined movements, consisting of "motions" and "gestures", which can be detected across both hands. Motions include easily observed behaviors such as scaling, rotating, and translating, while gestures include "swipe", "circle", "tap", "pinch" and "grab". This represents a relatively modest list in number, but the three-dimensional size and number of digits employed in making a gesture increase the pool of potential identifiable interactions. Gestures can also be combined either in sequence or performed simultaneously to trigger software events.

The key effort in the project was to determine an appropriate set of gestures which can be mapped to

specific instrument operations. This was complicated by the “best efforts” mathematical model of the Leap Engine, which can misinterpret user motions and gestures, and the non-discrete nature of many gestures, where overlap in movements or positioning of individual gestures in a serial or parallelized sequence causes gesture interference and misidentification. All these forms of gesture collision had to be avoided. This led to a highly defensive and occasionally baroque coding style utilizing multiple granular qualifiers and careful ordering of high-level conditional cases. For example, an action might be triggered with the gesture “index finger circling”, but this was qualified with conditions on whether the finger was the only one currently extended in the frame, whether a particular hand was being used, whether the gesture was in progress or recently stopped. Table 1 provides a mapping between SEM function and Leap behaviors that were successfully implemented within the project. Locking gestures to particular hands, i.e. SEM movement actions to the left hand and SEM trigger actions to the right hand was a simple strategy which significantly reduced the risk of gesture collision. To toggle between zooming and panning, it was necessary to make a gesture with the right hand and then resume moving the left hand.

Task	Associated Gesture
Run up the beam	Spin index finger of right hand clockwise for two seconds
Turn off beam	Spin index finger of right hand anti-clockwise for two seconds
Activate autofocus	Pinch index finger and thumb of the right hand for two seconds
Zoom out	Move the left hand to the far left of the device in the X axis
Zoom in	Move the left hand to the far right of the device in the X axis
Panning	Move the left hand (in both the X and Y axes) to the far left of the device to pan left, the far right to pan right, ahead of the device to pan upwards and behind the device to pan downwards.

**Table 1:** Mapping of SEM Tasks to Gestures

In conclusion, the project demonstrated that it was feasible to use Leap sensors to control a SEM. However the Leap Motion is insufficiently accurate in its recognition of hand and finger positions and of gestures for reliable and precise control of a microscope. The default selection of recognized gestures is extremely limiting; compiling multiple gestures and recognition qualifiers in a stack is only practicable in scenarios where there are limited gestures needed and those gestures can have symbolic meaning attributed to their mapped function. Modern microscopes have too many useful functions relative to the Leap gesture pool – with so many functions relying on a core capability (such as zooming and panning), the gesture pool for other functions is emptied rapidly [4].

#### References:

- [1] A Bracegirdle, “Investigating the Usability of the Leap Motion Controller: Gesture-Based Interaction with a 3D Virtual Environment”, 2014 Dissertation, University of Canterbury, New Zealand.
- [2] L Shao, “Hand movement and gesture recognition using Leap Motion Controller”, 2016 Dissertation, Stanford University, USA.
- [3] W Delamare *et al*, “Designing 3D Gesture Guidance: Visual Feedback and Feedforward Design Options”, (Université Grenoble Alpes, Grenoble).
- [4] Leap Motion is a trademark of Leap Motion, Inc. This research was supported by funding from Carl Zeiss Microscopy and in-kind support of the first author by BT plc. The authors gratefully acknowledge the assistance of Carl Zeiss personnel, especially Daniel Aldridge, David Hubbard and Hidde Wallaart.