SpeckleSense: Fast, Precise, Low-cost and Compact Motion Sensing using Laser Speckle

Jan Zizka [1,2] Alex Olwal [1] Ramesh Raskar [1]

[1] MIT Media Lab
77 Mass. Ave., E14
Cambridge, MA, USA

[2] Comenius University DAI
Mlynska dolina
Bratislava, Slovakia

{zizka, olwal, raskar}@media.mit.edu, http://specklesense.media.mit.edu/

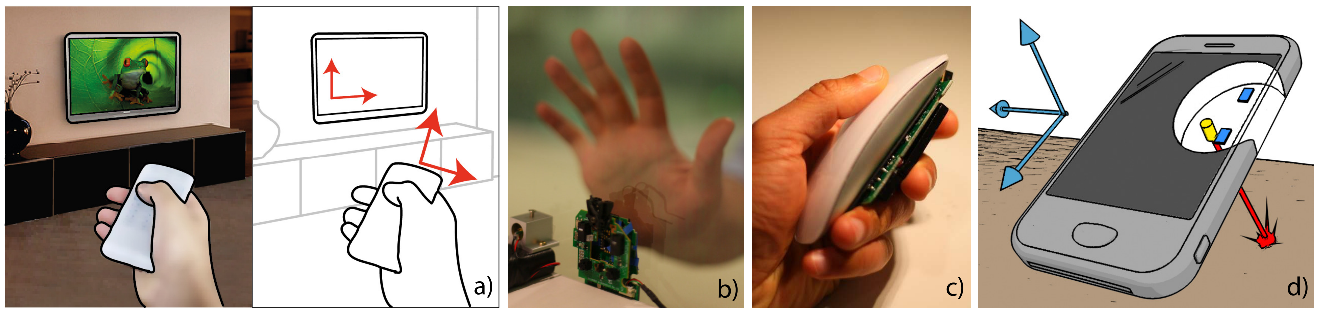


Figure 1: SpeckleSense uses laser speckle sensing to enable effective motion-based interaction that is applicable to many scenarios, such as a) motion-sensing remote controls, b) interaction with public displays, c) 3D input devices, and d) as the next-generation sensors for mobile devices.

ABSTRACT

Motion sensing is of fundamental importance for user interfaces and input devices. In applications, where optical sensing is preferred, traditional camera-based approaches can be prohibitive due to limited resolution, low frame rates and the required computational power for image processing.

We introduce a novel set of motion-sensing configurations based on laser speckle sensing that are particularly suitable for human-computer interaction. The underlying principles allow these configurations to be fast, precise, extremely compact and low cost.

We provide an overview and design guidelines for laser speckle sensing for user interaction and introduce four general speckle projector/sensor configurations. We describe a set of prototypes and applications that demonstrate the versatility of our laser speckle sensing techniques.

ACM Classification: H5.2 [Information interfaces and presentation]: User Interfaces. - Graphical user interfaces.

General terms: Design, Human Factors, Experimentation.

Keywords: Input devices, tracking, mouse, laser speckle

INTRODUCTION

Motion-based interaction has recently received widespread popularity thanks to low-cost camera-based technologies and an abundance of inertial sensors in consumer electronics. Today, all major video console platforms have the ability to sense human motion images using special-purpose cameras. Hundreds of millions of cell phone cameras have made image sensors extremely affordable, but their use for real-time motion tracking is limited by low pixel throughput (the product of resolution and frame rate). While special-purpose cameras can achieve higher frame rates, resolution is limited by image processing demands, which in turn affects the possible accuracy.

Many projects have explored creative usage of image sensors that avoid the implicit limitations when a camera is used to form an image of a scene in the classical sense. This has, for example, enabled compact imaging systems that can read extremely small patterns, such as Bokode [23].

Laser speckles are micro-patterns that are formed from the interference of scattered coherent light. Laser speckle tracking techniques have traditionally been used for applications like measurements and particle tracking in mechanical engineering and biology [12]. This paper explores the application of laser speckle to user interfaces and motion tracking, as their unique characteristics make them an interesting alternative or complement to classical camera- or sensor-based tracking techniques.

CONTRIBUTIONS

We exploit the high frequency interference patterns caused by lasers and sense direct or reflected light using a high framerate lensless 2D image sensor. Fast optical flow computation helps us track laser or sensor motion to enable new

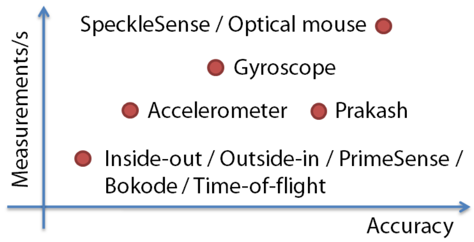


Figure 2: SpeckleSense's relative motion tracking combines high accuracy with fast update rates.

input devices and interaction techniques. Our contributions are:

- A novel laser speckle sensing technique that is particularly suitable for motion-based input. It is fast, precise, can be extremely compact, is low cost, and requires little computational power.
- An overview and design guidelines for using laser speckle sensing in human-computer interaction.
- Four general speckle projector/sensor tracking configurations for motion-based input.
- Two practical implementations of laser speckle tracking for human-computer interaction. The first type is based on a high-speed, low-resolution digital camera. The second type is based on readily available consumer electronics and optics, where we exploit the capabilities of optical sensors from computer mice.
- A set of prototypes and applications that demonstrate how laser speckle sensing can be used both standalone as well as in combination with other devices.

RELATED WORK

Surface interaction

Motion sensing is relevant to all classical pointing devices; mice, (multi-)touch displays, touchpads, pointing sticks, styli, joysticks, and others. The optical mouse is probably one of the most popular input devices. It uses a small camera that images and tracks the naturally occurring texture of the surface to establish its 2D motion on the surface.

Considerable effort has been invested in supporting additional degrees-of-freedom (DOF), beyond just two, for computer mice. MacKenzie et al. use two balls to add one DOF [22], whereas VideoMouse [18] uses a special mouse pad, camera and computer vision. The Rockin' Mouse [5] is augmented by inertial sensors and Baudish et al. create a mid-air mouse using a flexible "skin" [7]. Villar et al.'s five prototypes [38] and Apple's Magic mouse [2] replace the mouse wheel with a multitouch surface that enables sensing of 2D multi-finger gestures. Integral 3D mice [1, 4] are intended for effective spatial manipulation.

Table 1: SpeckleSense tracks motion velocity using a projector-based approach.

	Camera	Inertia	Projector / Structured light
Acceleration		Accelerometer	
Velocity	Optical mouse	Gyroscope	**SpeckleSense**
Position	Inside-out (Wii)		Bokode
	Outside-in		Prakash
	Time-of-Flight		PrimeSense

Mid-air interaction

As discussed in a survey by Welch and Foxlin [40], several options for optical and non-optical motion tracking are available. Sony PS Move[1] and Xwand [41] demonstrate the advantages in combining optical and inertial tracking. Similarly, Nintendo Wii[2] tracks fixed markers with a moving camera and inertial sensors. Commercially available mid-air mice [21] and [13] use inertial sensors only. The PrimeSense [29] sensor used in Microsoft Kinect[3] recovers depth images with structured IR light to enable markerless and deviceless user interaction.

Motion-based interaction

Several projects investigate the use of sensing to create spatially aware displays [11, 25, 26, 34, 36, 43, 32] for new forms of human-computer interaction with handheld displays. Motion-based input for small displays has also been explored as a promising direction to expand the possible interaction [8, 14, 15, 17, 39, 24, 35, 42] for portable devices.

The SpeckleSense principle is related to other projector-based tracking techniques, where the environment is spatially divided and coded using special lightning (See Figure 2 and Table 1) . The Office of the Future [31], Bokode [23] and Prakash [30] projects use incoherent structured light, while PrimeSense [29] uses special diffraction gratings to form a desired pattern.

Speckle phenomena and HCI

After its theoretical development in the 1970's and 1980's, speckle-based techniques have begun to find their way from controlled laboratory conditions to practical applications and user interaction. Popov et al. [28] describe the design and simulation of optical computer mice, where five photodiodes track the speckle pattern, and Schroeder et al. present a similar solution [37]. DePue et al. [10] estimate the device's distance to the surface using two detectors and two lasers with different wavelengths, and Bathiche et al. [6] describe a technique to suspend tracking when the device is lifted. Liao at al. investigate designs with lenses and limiting apertures [20] and these principles are also used to track a mechanical finger-controlled pad [19]. Reilly and Hanson [33] use CCD linear arrays and collection optics to capture speckle pattern that is reflected off the skin. The Philips Laser Doppler [27] is different from speckle sensing, as it uses interferometry techniques to measure the Doppler shift in the frequency of laser light, which is proportional to velocity.

---

[1] http://us.playstation.com/ps3/playstation-move/
[2] http://www.nintendo.com/wii
[3] http://www.xbox.com/kinect

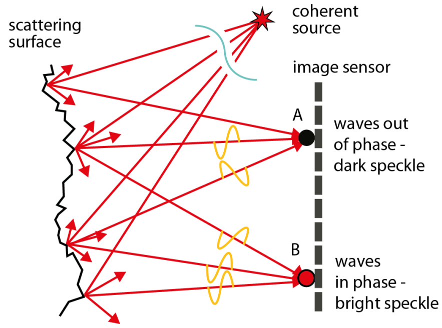


Figure 3: The laser speckle phenomena. Light from a coherent light source scatter on a surface's micro-structure and the resulting reflected waves reach an image sensor in different phases, due to differences in their travel paths. The interference from each wave contributes to the intensity at each pixel.

LASER SPECKLE: EXPLOITING COHERENT LIGHT FOR FAST, PRECISE TRACKING

When a diffuse object is illuminated with coherent light, the resulting interference forms a speckle pattern, as illustrated in Figure 3. In the ideal case, each point of the surface scatters the incoming light waves in all directions. Each pixel in an image sensor placed in the field would thus receive contributions from multiple reflected waves. More importantly, the measured intensity in each pixel is a result of the interference from each wave, as they all have traveled different path lengths from the surface to the sensor. These waves have different, theoretically random, phases. Most surfaces (except e.g., glass) are sufficiently rough to produce statistically independent phases of waves. The resulting images contain grainy, high-contrast structures that are referred to as "speckle" (See Figures 4–7).

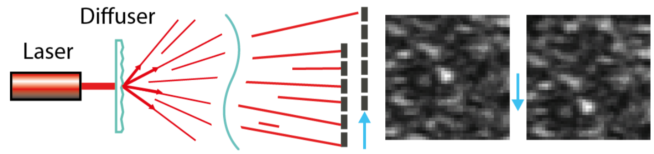


Figure 4: The speckle pattern motion can be tracked with image processing algorithms as the speckle sensor or projector move.

Speckle Projector: Generating speckle patterns with laser + diffuser

Speckle patterns can also be generated by integrating a laser diode with an optical diffuser, to create a *Speckle Projector*, as illustrated in Figure 4.

The light from such a speckle projector has similar characteristics to a point light source, if the diameter of the diffuser's illuminated area is significantly smaller than its distance to the image sensor (i.e., at least one order of magnitude smaller). This makes it possible to project a dense (but unknown) pattern into space, which can be sensed using a lensless image sensor.

The diffuser controls the projection angle, which also affects energy distribution and intensity.

Speckle tracking

While we have no control of the projected speckle pattern, we can use image processing and computer vision techniques to analyze its properties.

Relative motion tracking

A small translation v of the image sensor will translate the captured image by $-v$. Similarly, a translation v of the speckle projector in a plane parallel to the sensor would shift the image by v, as shown in Figure 6. On the other

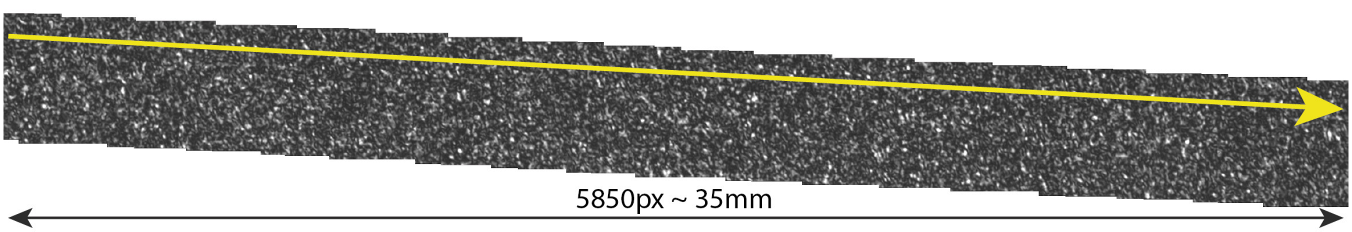


Figure 5: Frame-to-frame tracking of 22 speckle patterns, displayed in a stitched panorama image.

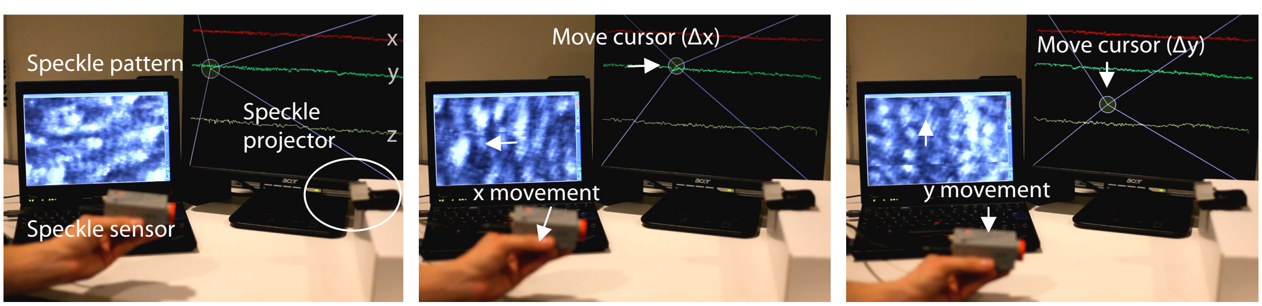


Figure 6: The speckle sensor is sensitive to small translations relative to the projected speckle pattern. A high frame-rate camera that is matched with the laser speckle characteristics can be used to track fine frame-to-frame movement. Here, the movement controls a cursor in our software.

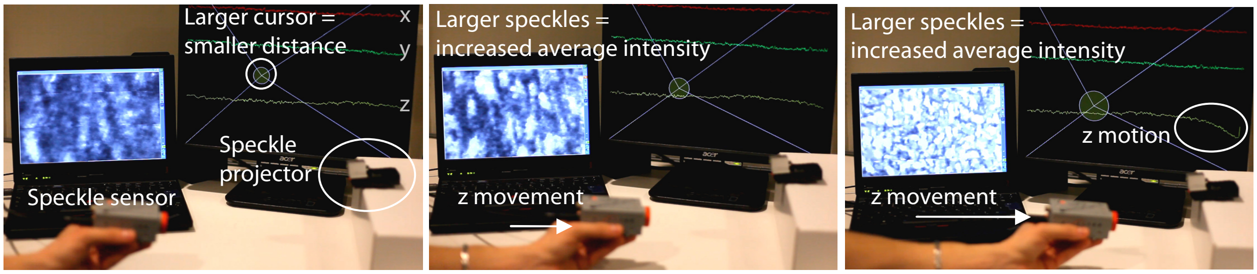


Figure 7: As the speckle sensor is moved closer to the speckle projector, speckles become larger in the image. While the speckle pattern is unknown, the statistically consistent distribution allows us to estimate distance with average intensity. Here, the distance controls the circle's radius in our software.

hand, a small change in the speckle projector's orientation will move the speckle image by θd, where θ is the angle, and d is the distance to the image sensor.

Distance approximation through speckle structure

As with any beam, the projected laser speckle will increase with size as the speckle projector gets closer to the sensor. The specific speckle pattern is, however, neither controlled nor known, as both the distribution of speckles and their sizes vary. The average sizes and distribution is, however, consistent, which allows us to approximate distance by analyzing the average image intensity. The size of speckle varies in space according to the following equation [9]:

$$s \sim \lambda \times d / a \qquad (1)$$

where d is distance to the illuminated surface of diameter a and λ is the laser's wavelength. We can thus estimate the absolute distance between the speckle projector and sensor if λ, a and pixel size is known. See Figure 7.

Design criteria for the speckle sensor apparatus

To track speckles we must ensure that they are larger than the sensor's pixel size, to avoid the averaging that would happen if several speckles (bright and dark) would be captured in the same pixel, as the needed contrast would be lost. On the other hand, the speckles must be smaller than the sensor, over the whole tracking range.

The image motion computations depend on how fast the camera can acquire subsequent images for frame-to-frame comparisons. The maximum speed (mm/s) at which a system can correctly track is given by the following equation:

$$Speed_{max} = PixelSize \times Overlap_{max} \times FPS_{max} \quad (2)$$

where $PixelSize$ is in mm, $Overlap_{max}$ is maximal frame-to-frame translation in pixels and FPS_{max} is the camera's highest frame rate.

Creating speckles on external surfaces

A collimated laser beam can be used to create a virtual speckle projector on an external surface, such that the reflected speckle pattern can be tracked with a speckle sensor that is rigidly attached to the laser, as shown in Figure 8.

As the laser beam hits the surface, reflected waves reach the sensor's pixels, and their interferences generate the measured per-pixel intensities. Let us assume that this device is translated by p, the sensor's pixel size. Then, the projected laser beams P_1 (before movement) and P_2 (after movement), will have an overlapping area Q. In-between frames Q will generate a static speckle pattern, which can be tracked with motion estimation algorithms. The speckle pattern variation due to the difference between P_1 and P_2 will have a negligible influence on the tracking as long as Q is significantly larger than p. Thus, for detection of one-pixel speckle motion, the sensor frame rate must be sufficiently high.

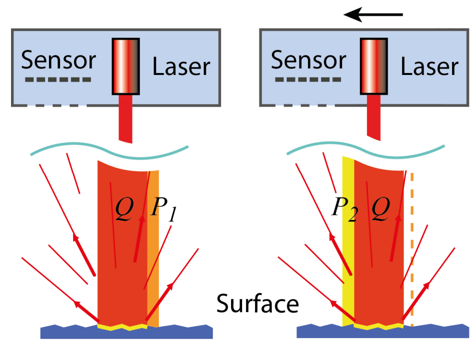


Figure 8: A device with an embedded speckle sensor and laser can detect its relative motion along a surface if the sensed inter-frame motion (P_1-P_2) is significantly smaller than the laser beam's width. The speckles generated by the area Q will thus dominate the speckle pattern with frame-to-frame trackable patches.

SPECKLE TRACKING CONFIGURATIONS

Our speckle tracking relies on active illumination (speckle projector) combined with an optical sensor (speckle sensor), but differs from traditional camera-based motion tracking systems, due to the flexible configurations possible with simple hardware. Before we describe a number of prototypes where we use multiple speckle projectors and sensors as building blocks for novel input devices, we will explore four general tracking configurations using a single speckle projector/sensor pair. Speckle tracking provides the following characteristics in all of our configurations:

- *High sampling rate.* The optical sensor has a very high sampling rate (approximately 10,000 Hz) compared to traditional digital cameras (30–100 Hz).
- *High sensitivity.* The dense speckle pattern allows tracking down to 50 µm.

- *Extremely small.* The active components of the speckle projector and sensor are only a few millimeters in diameter, making them suitable for integration in embedded systems, like mobile devices. Displays that are too small to fit touch-screens or other input controls may especially benefit from the ability of using the space in front and around them for user interaction [8, 15].
- *Flexible choices for projection angles — ultra-wide or narrow.* In our experiments we use optical diffusers that project the speckles ultra-wide and uniformly over 80–120 degrees. Narrower diffusers can be chosen if it is desirable to limit the field-of-view or to increase the range (as the energy will be less diffused).
- *Flexible choices for sensor field-of-view — ultra-wide or narrow.* Light falls directly on the surface of the lensless image sensor and from all directions (180°). Narrower field-of-view can be achieved with small apertures or masks.
- *Compatible with legacy applications.* Laser speckle can be sensed using modified optical mouse sensors, for direct compatibility with mouse-controlled applications (avoiding drivers or intermediate software).

Handheld Speckle Projector (A)

Digital Pointing Device, Purely Optical Communication

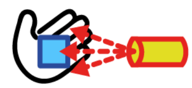
A fixed optical sensor can be used for precise motion tracking of a handheld speckle projector to create a digital pointer for relative 2D input. This configuration does not require any other communication between the handheld input device and the system, besides the motion data that is optically transmitted with the projected speckle pattern. Several characteristics distinguishes it from the on-board sensors (e.g., accelerometers or gyroscopes) that are typically used in current digital pointing devices.

- *Relative tracking solely through optical tracking.* Lack of radio and active electronics greatly simplifies hardware design, making the design low-cost and power efficient.
- *Fast transmission.* Motion data is transmitted at the speed of light, which eliminates the limitations and lag associated with radio-based communication.
- *Directional.* The optical communication allows a single speckle projector to be used for control of multiple devices without the need for radio-based pairing or connections. The speckle projector could thus be used for casual control of multiple simple devices around one’s home, e.g., communicating directly with a light bulb with an embedded speckle sensor.

We could also augment the optical communication with additional data, to enable actuation and mode switching, while preserving the minimalistic design. For this purpose, one could simply use the light modulation techniques that are found in standard IR remote controls and encode commands by pulsing the laser at high frequencies. We, for example, envision a low-cost approach for adding continuous motion to traditional remote controls. A user could thus turn on a TV using standard remote control buttons, and use relative motion with its embedded speckle projector to increase/decrease volume.

Similarly, unique device IDs could be optically encoded to enable device-specific functionality, e.g., when switching between input devices. A limitation of the minimalistic design is that *simultaneous* multi-device input is a bit more complex. One approach for a small set of speckle projectors at different wavelengths could be to separate the input using matching infrared filters for a corresponding set of speckle sensors. With different infrared filters, this approach scales well under the assumption that speckle projectors and sensors are available at low cost and with negligible additional computational requirements.

Handheld Speckle Sensor (B)



Speckle Sensor in Handheld Device → Long-range, Mid-air Tracking

An input device with a speckle sensor can be used to track its relative movement in a projected speckle pattern. In this configuration, we fix a speckle projector in the environment, directed towards the user, who interacts by moving an input device with an embedded speckle sensor. A portable device (e.g., a mobile phone) can use the motion data independently, whereas an input device that controls a remote system would rely on radio, cable or optical communication to report the sensor data. While this introduces more complexity, it adds the inherent support for unique identification of multiple devices and the use of additional physical controls (e.g., buttons, scroll wheels, and touch pads).

- *Rotation-invariant translation.* The motion estimation is only sensitive to global translations since the sensor tracks the speckle pattern's vector field. Rotation of the sensor will results in a skewed image with negligible motion, as parts of the pattern will either move in opposite directions, or not move at all. This enables precise translational movements while avoiding influences from unintentional rotational motion.

Self-tracked Device (C)

Embedded Laser + Speckle Sensor → Device Tracked On & Above Surface

The speckle phenomena can also be exploited in a standalone configuration where both sensor and light source are embedded in the input device, similar to the configuration of a traditional optical mouse. In our configuration, however, the speckle sensor tracks the resulting speckle pattern produced by an embedded laser's coherent light when it reflects off the surface. Such a mouse works both on and above the surface, depending on the effect of the laser and the aperture of the optical sensor.

- *Standalone configuration.* Only needs surface for operation and works both on and above the surface.

Speckle Tracking of Hands (D)

Laser + Sensor in Environment → Tracking Hand Motion

For gestural interaction, we can exploit the body as a speckle generating surface when illuminated with a laser. In this configuration, the laser and speckle sensor are integrated in the device to support hand interaction in front of it. It allows unencumbered interaction without special-purpose input devices and is suitable for integration in small form factors or in public displays where it may not be desirable or feasible to use physical controls.

HYBRID TRACKING CONFIGURATIONS

The four general configurations consisting of a single speckle projector/sensor, described above, could be extended for additional functionality using multiple lasers/diffusers and optical sensors. Here, we describe a few interesting examples that we have started to explore.

Tracking Both on Surface and for Mid-air Gestures (E)

On-surface + Longe-range, Mid-air Input

The handheld speckle sensor in the long-range configuration (B) can be reused to also support traditional on-surface input, by mounting the sensor at a 45° angle and adding a downward-pointing low-effect laser (See Figure 9). When moved on the surface, the embedded laser will hit the surface and create a speckle pattern that will be seen by the speckle sensor. When the device is lifted from the surface, the low effect of the embedded laser will not create a sufficiently bright pattern, and instead, a speckle projector in the environment (e.g., integrated with a large display) will provide lateral motion tracking.

It may be desirable to avoid simultaneous sensing of speckle from the embedded laser and the source in the environment. This could be addressed with a proximity sensor at the bottom of the input device to control the projections, or by arranging the laser and speckle sensor such that lift-off would limit the visibility of the embedded laser light [6].

Two Sensors for Relative Tracking in 3D Space (F)

Standalone Relative 3D Tracking

By adding a second optical sensor to the Speckle-sensing mouse configuration at an angle (e.g., 45°), we can also recover relative motion above the surface.

When the device moves perpendicular to the surface, the sensor aligned with the laser is only observing speckle scaling, whereas the second off-axis sensor senses vertical motion, which corresponding to change in depth. See Figure 10a. The use of two sensors also works in other configuration, e.g., hand tracking, as shown in Figure 10b.

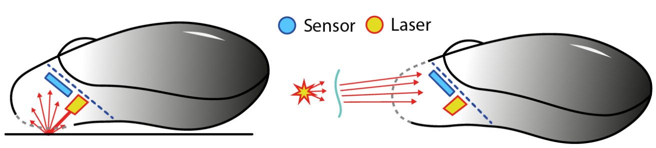


Figure 9: *Hybrid tracking configuration E* uses a speckle sensor at 45° to allow tracking both on the surface and in mid-air. An embedded laser creates speckle that can be sensed when the mouse is on the surface, while a speckle projector installed in the environment allows tracking in mid-air.

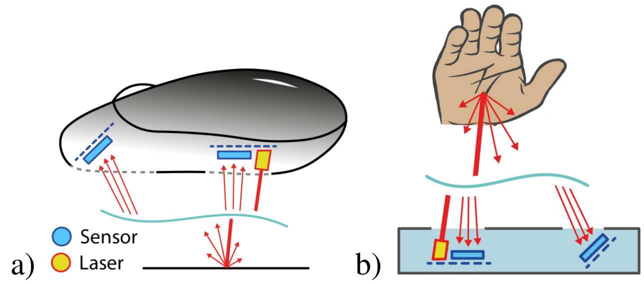


Figure 10: *Hybrid tracking configuration F* uses two speckle sensors to estimate relative motion orthogonal to a surface. a) Distance estimation with two embedded sensors in a mouse. b) The user's hand is used as a speckle generating surface when illuminated with colliminated light, allowing its 3D motion to be tracked with two fixed speckle sensors.

IMPLEMENTATION

Our initial experiments, shown in Figures 6 and 7 were conducted using a PixeLINK PLA741 firewire camera (http://www.pixelink.com), set to an image resolution of 160×120 pixels at 300 fps. The configuration makes it possible to demonstrate and experiment with speckle tracking, even if it the amount of motion velocity that can be tracked is limited by hardware.

The limited frame rate (30–100 fps) of typical digital cameras prohibits speckle motion tracking due to small sensor sizes. While it is impractical to dramatically increase the sensor size, we can use a fast camera (compared to motion velocity) to estimate the relative 2D motion of either the sensor or speckle projector.

Optical sensors in modern mice have small, but fast sensors with dedicated optical flow calculations in hardware. The gaming grade sensor Avago ADNS-9500 [3] captures images at 10,000 fps, and computes and reports the frame-to-frame motion vector at 1,000 Hz. In several of our prototypes, we repurpose the components from wireless (Logitech M505) or wired (Logitech G500) optical mice, and use the optical sensors with lenses removed as speckle sensors..

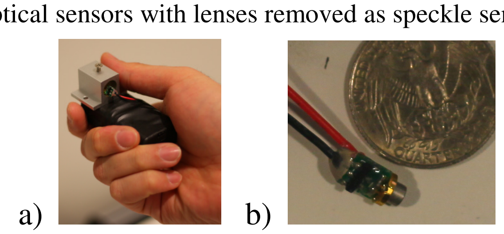


Figure 11. a) Our handheld Speckle Projector consists of a 50 mW laser (780 nm), diffuser (80°) and battery. With our speckle sensors it has a 3 m range. b) The laser diode is only a few millimeters wide and could be integrated into various hardware configurations.

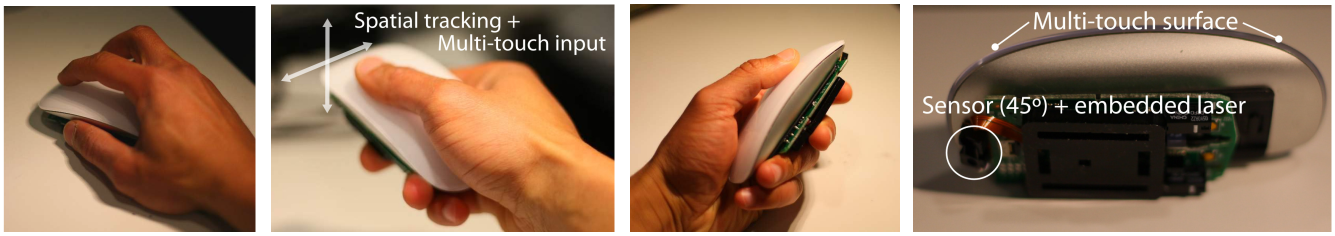


Figure 12: The TouchController works both as a regular mouse and in mid-air, tracked through a speckle projector in the environment. Its multi-touch surface enables a combination of spatial gestures and multi-touch interaction.

To optimize our results according to Equation 1 and 2, we use two types of lasers in our prototypes. For short-range interactions we use a 5 mW, 650 nm laser without diffuser. For long-range interactions (3 m tracking range), we use a 50 mW, 780 nm laser combined with a diffuser from Luminit (http://www.luminitco.com/) that provides a uniform 80° distribution (See Figure 11).

PROTOTYPES AND APPLICATIONS

Using our toolbox of tracking configurations we developed a set of prototypes and applications to explore the possibilities of laser speckle sensing, combining it with different physical controls and form factors.

TouchController: Remote Translation + Multi-touch

Numerous input devices exist where accelerometers and/or gyroscopes map device orientation to mouse translation on large displays, for example, during presentations.

Our TouchController, shown in Figure 12, instead allows the use of small, rotation-invariant translations. It uses a sensor from an optical mouse with the lens removed, and pointed at 45°. This allows both ordinary surface tracking of the speckle pattern created by the embedded laser, and tracking when lifted up and pointed in the direction of a speckle projector (See Hybrid Configuration E, above).

We augmented the device with a multi-touch surface (from an Apple Magic Mouse), such that mid-air motion could be combined with multi-touch gestures on its surface.

To explore interactions with the prototype, we implemented a 3D viewer using Java and the Processing environment (http://www.processing.org) and used Python on Mac OS X and its native multi-touch libraries to track finger position, orientation and contact size on the multi-touch surface. The speckle tracking was implemented with C# on Windows XP using the Raw Input libraries. Both tracking applications stream their data to the 3D viewer over UDP and control rotation, scale and translation of a 3D model:

- *Single-finger drag:* Rotates the model.
- *Pinch-zoom:* Scales the model.
- *Moving the device in air while touching surface:* Designed to simulate "grabbing" the model and translating it in the screen plane.

Mobile Viewport: Spatially Tracked Multi-touch Display

Inspired by previous work [11, 16, 25, 26, 34, 36, 43] on spatially aware handheld displays, we combined our 3D tracking (Configuration F, above) with a mobile phone. See Figure 13.

Relative tracking in the space above a surface is performed using two optical sensors, a small laser (5 mW) and a mirror for compactness, which were integrated with the bottom part of an Android OS phone (Motorola Droid X).

We implemented two applications. In a medical imaging viewer, we allow exploration of the slices in a CT stack by moving the device in 3D space. In an image viewer application, distance from the surface instead controls zoom level. The software was implemented in Java for the Android platform. Currently, it receives tracking events over UDP from a PC that interfaces with the wireless mouse sensors and calculates relative 3D motion. In an integrated solution, we expect the speckle tracking components to be embedded directly on the device, similarly to currently popular onboard sensors (accelerometers, gyroscopes, camera, etc.).

The graphics can be manipulated both with multi-touch gestures and spatial motion:

- *Move the device orthogonally to the surface:* Pan in current slice or image.

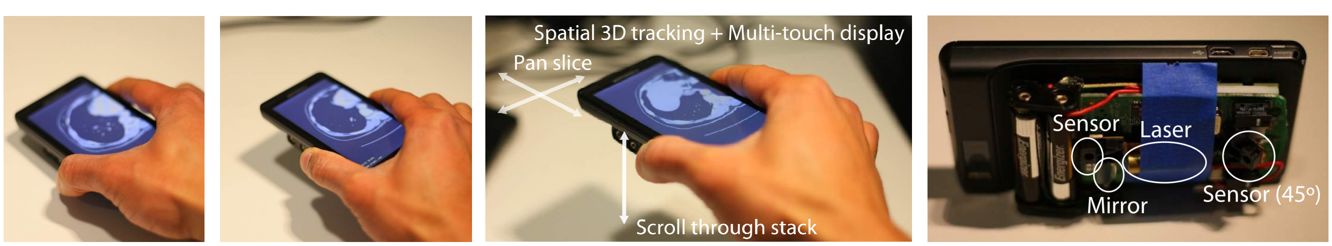


Figure 13: The Mobile Viewport adds relative 3D tracking to a mobile phone. Interaction is supported both by moving the device in 3D space and through multi-touch interaction on the display. A user can pan around in a medical image viewer and scroll through a CT stack by moving the device in 3D, or pan around in the view of a web page and control zoom level with distance.

Figure 14. Low-cost motion tracking for public displays using an embedded Speckle projector and sensor (configuration D). The user's 2D hand movement selects the fullscreen image from a matrix of thumbnails in the lower left corner.

- *Move the device closer/further from surface:* Go up/down in medical image stack, or zoom in image viewer.
- *Pinch-zoom:* Zoom in the medical slice, and alternative method to zoom in image viewer.
- *Two-finger drag:* Alternative way to pan current slice.
- *Single-finger drag:* Add annotations (draw).

Motion input for public displays

To enable natural interaction in public displays without requiring physical input devices or touch-sensitive surfaces, we integrated a speckle sensor (optical mouse sensor) and laser (Configuration D) behind a glass window.

When hit by the laser, the user's hand creates speckles that are reflected to the speckle sensor. This allows non-contact interaction with the display, which may be desirable, both for sanitary reasons and to protect the input technology from users and environment.

We developed an image browser application in C# that interfaces with the speckle sensor, allowing it to be controlled with the hand, as shown in Figure 14. The user's 2D hand movement controls the selection in a matrix of thumbnails in the lower left corner. The currently selected photo is shown in full-screen, and to keep the current selection, the user pulls back the hand. Currently, we do not provide actuation in this configuration, but it would be possible to combine the sensing with swiping gestures, for example.

PERFORMANCE AND LIMITATIONS

Accuracy. The mouse sensor and our algorithm integrate huge amount of small shifts each second (10 kHz). This results in an accumulative error which increases over time. We used a linear actuator controlled by a stepper motor as a ground truth reference in an experiment to recover the accumulative error. The trajectory was 35 cm, movement took 1 s and we ran 10 experiments. The average deviation was 0.0217% (0.076 mm).

Effective range. The effective range for the mid-air configurations (A, B) was ~3 m (diffused 50 mw laser). In reflective mode, effective range depends on the reflectivity of the surface. With our 5 mw laser, we are able to precisely track 20 cm above a reference sheet of white paper. When the reflectivity of the surface is k times lower than our reference, the effective range is $sqrt(k)$ times smaller, as the incoming energy decreases with the square of the distance. As a comparison, we tracked ~6 cm above black plastic.

Ambient light. The combination of a lensless image sensor and coherent light ensures robustness to ambient light. Since incoherent light (e.g., light bulbs and LEDs) will not be focused in a lensless configuration, it will be blurred, as each light source produces an equal contribution to each image sensor element, and the sum of all such sources produces an average intensity signal. The laser speckle is added to this average intensity and creates a lower contrast image, but even with a high average intensity component in our experiments, we would typically have sufficient information to compute frame-to-frame motion.

PRELIMINARY USER FEEDBACK

For initial informal qualitative feedback on our prototypes, we invited five colleagues (25–35 years, all male) from our department to test our hardware prototypes and applications. All participants had used motion controllers and regularly used mobile phones with multi-touch displays.

For each interface we demonstrated how to use the input device to control an application and then allowed them to get familiar with it for a few minutes.

For each of the prototypes, we asked them to perform a set

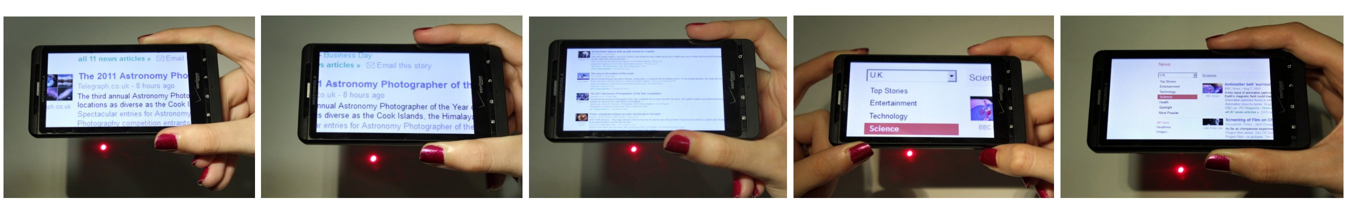


Figure 15: A participant in our informal feedback sessions zooms into a web page with the Mobile Viewport interface by varying the device's distance to the surface.

Figure 16: A participant exploring the TouchController for translation, rotation and scale of a model in our 3D viewer.

of tasks based on navigating, positioning, orienting or selecting objects, depending on the application. We asked the participants to think aloud and encouraged them to provide both positive and negative comments.

Participants liked the concept but found the public display prototype the hardest to use. The lack of actuation made it difficult to get in/out of tracking mode or reposition the hand. It also required a bit of practice for browsing the images, as the interaction area is quite small. All participants were able to control the imagery in the end, but had difficulties moving out of the tracking range without activating another image. One participant said that the continuous "Midas touch"-like tracking would not be a problem if the hand could be held still in a comfortable location. He also suggested the use of multiple sensors to expand the tracking area. Another participant initially tried swiping gestures, which we had not adapted the user interface for. For the next steps with this prototype, we plan to use multiple speckle sensors to both expand the tracking area and provide depth sensing, which could allow better visual feedback for entering/exiting the tracking area, and support actuation through proximity. A fast speckle-sensing image sensor where we have access to the image buffer (as in the experiments shown in Figures 6 and 7) would also allow us to recover absolute depth in a calibrated installation.

Participants appreciated the Mobile Viewport (See Figure 15), where they panned around and zoomed in a 1920×1200 screenshot of a news website on the 480×854 pixel mobile display. All participants were able to pan and zoom into different areas when asked. Several participants especially liked the single-handed zoom when moving the mobile phone closer or further from the surface. Based on the feedback we plan to also add smoothing to the tracking as it became clear how critical this was in tracking mode, as small pixel shifts disturbed the reading experience. We plan to create a more compact version of the sensors, as the current version affected the possibilities for grip.

The TouchController for controlling a 3D model on a projected wall was the most appreciated prototype (shown in Figure 16). One participant had initial difficulties in moving the model as he was swiping very fast and with a rotational motion, which resulting in the handheld device pointing away from the speckle projector. After additional instruction, he was able to successfully control the model. The combination of multi-touch input and spatial manipulation worked well for the participants, and two of them suggested that we also added depth control.

CONCLUSIONS AND FUTURE WORK

In this paper we introduced laser speckle sensing as a powerful and effective technique for motion sensing in human-computer interaction. It overcomes many of the limitations of traditional camera based systems and allows fast and precise tracking with a compact, low-cost technology that can perform the computations in embedded hardware.

We have also presented a set of prototypes and applications that demonstrate the potential of laser speckle sensing for different human-computer interaction scenarios. In future work, we plan to refine our prototypes and evaluate them in comparison with established technology and devices based on accelerometers, gyroscopes and other approaches to motion sensing. We also plan to further develop our speckle analysis methods, by switching to developer hardware, where we can access raw image data from fast (10,000 fps) image sensors.

A formal user study that evaluates the performance of our input devices and techniques will also provide additional insight into the benefit of our sensing technology's high precision to user interaction in a variety of applications.

ACKNOWLEDGMENTS

Jan Zizka was supported by Comenius University and ME-Inspection, Alex Olwal was supported by a Swedish Research Council Fellowship and a Blanceflor Foundation Scholarship, and Ramesh Raskar was supported by an Alfred P. Sloan Research Fellowship. We thank Samsung Electronics for support, the members of the MIT Media Lab Camera Culture group, the participants in our preliminary evaluation, and our reviewers for their insightful feedback.

REFERENCES

1. 3Dconnexion. Space 3d mice. http://www.3dconnexion.com/. July 2011.
2. Apple Magic Mouse. http://www.apple.com/magicmouse/. July 2011.
3. Avago Technologies. Adns-9500 sensor. http://www.avagotech.com/docs/AV02-1726EN. July 2011.
4. Axsotic. 3d-spheric-mouse. http://www.axsotic.com/. July 2011.
5. Balakrishnan, R., Baudel, T., Kurtenbach, G., and Fitzmaurice, G.. The Rockin' Mouse: Integral 3D Manipulation on a Plane. *Proc. CHI '97* (1997), 311–318.
6. Bathiche, S. N., and Starkweather, G. K., Data input device for tracking and detecting lift-off from a tracking surface by a reflected laser speckle pattern, U.S. Patent 7,161,582., 2007
7. Baudisch, P., Sinclair, M., and Wilson, A., Soap: a pointing device that works in mid-air. *Proc. UIST '06* (2006), 43–46.

8. Butler, A., Izadi, S., and Hodges, S. SideSight: multi-"touch" interaction around small devices. *Proc. UIST '08* (2008), 201-204.
9. Dainty, J. C. Laser Speckle and Related Phenomena. Springer-Verlag, Berlin (1984).
10. DePue, M. T., Schroeder, D. W., and Xie, T., Optical device that measures distance between the device and a surface, U.S. Patent 7,427,981, 2008.
11. Fitzmaurice, G. W. Situated information spaces and spatially aware palmtop computers. *Commun. ACM 36, 7* (Jul. 1993), 39–49.
12. Goodman, J. W.. Speckle phenomena in Optics: Theory and Applications. Roberts and Company, Englewood, 2007.
13. Gyration. Air mouse. http://www.gyration.com/. July 2011.
14. Hachet, M., Pouderoux, J., and Guitton, P. A camera-based interface for interaction with mobile handheld computers. *Proc. I3D '05* (2005), 65–72.
15. Harrison, C. and Hudson, S. E., Abracadabra: wireless, high-precision, and unpowered finger input for very small mobile devices, *Proc. UIST '09* (2009), 121–124.
16. Harrison, C. and Hudson, S. E., Minput: enabling interaction on small mobile devices with high-precision, low-cost, multipoint optical tracking, *Proc. CHI '10* (2010), 1661–1664.
17. Hinckley, K., Pierce, J., Sinclair, M., and Horvitz, E. Sensing techniques for mobile interaction. *Proc. UIST '00* (2000), 91–100.
18. Hinckley, K., Sinclair, M., Hanson, E., Szeliski, R., and Conway, M. The Videomouse: A Camera-based Multi-degree-of-freedom Input Device. *Proc. UIST '99* (1999), 103–112.
19. Liao, C.-M., Huang, P. S., Chiu, C.-C., Hwang, Y.-Y., Ma, S.-I. Real-time finger-controlled navigation system using laser speckle patterns, Optical Engineering, Volume 49, Issue 5 (2010).
20. Liao, C.-M., Huang, P. S., Hwang, Y.-Y., Chen, M., Chiu, C.-C. Robust technique of analyzing and locating laser speckle patterns for optical computer mice, Optics and Lasers in Engineering, 47, 7-8 (2009), 875–883.
21. Logitech. Mx air mouse. http://www.logitech.com/en-us/mice-pointers/mice/devices/3443/. July 2011.
22. MacKenzie, S., Soukoreff, R., and Pal, C. A Two-ball Mouse Affords Three Degrees of Freedom. *CHI EA '97* (1997) 303–304.
23. Mohan, A., Woo, G., Hiura, S., Smithwick, Q., Raskar, R. Bokode: imperceptible visual tags for camera based interaction from a distance. *Proc. SIGGRAPH '09* (2009).
24. Ni, T. and Baudisch, P., Disappearing mobile devices. *Proc. UIST '09* (2009), 101–110.
25. Olwal, A. LightSense: Enabling spatially aware handheld interaction devices. *Proc. ISMAR '06* (2006). 119–122.
26. Olwal, A., and Feiner, S. Spatially Aware Handhelds for High-Precision Tangible Interaction with Large Displays. *Proc. TEI '09* (2009), 181–188.
27. Philips. Laser Doppler technology. http://lasersensors.philips.com/. July 2011.
28. Popov, P., Pulov, S., and Pulov, V. A laser speckle pattern technique for designing an optical computer mouse, Optics and Lasers in Engineering, 42, 1 (2004), 21–26.
29. PrimeSense. http://www.primesense.com/. July 2011.
30. Raskar, R., Nii, H., deDecker, B., Hashimoto, Y., Summet, J., Moore, D., Zhao, Y., Westhues, J., Dietz, P., Barnwell, J., Nayar, S., Inami, M., Bekaert, P., Noland, M., Branzoi, V., and Bruns, E. 2007. Prakash: lighting aware motion capture using photosensing markers and multiplexed illuminators. *Proc. SIGGRAPH '07* (2007), Article 36.
31. Raskar, R., Welch, G., Cutts, M., Lake, A., Stesin, L., and Fuchs, H. The office of the future: a unified approach to image-based modeling and spatially immersive displays. *Proc. SIGGRAPH '98* (1998). 179-188.
32. Reilly, D., Rodgers, M., Argue, R., Nunes, M., and Inkpen, K. Marked-up maps: Combining paper maps and electronic information resources. Personal and Ubiquitous Computing, 10, 4 (2006), 215–226.
33. Reilly, R. and Hanson, A. Gesture recognition for augmentative human computer interaction, Engineering in Medicine and Biology Society, 2 (1995), 1275–1276.
34. Rekimoto, J. and Nagao, K. The world through the computer: Computer augmented interaction with real world environments. *Proc. UIST '95* (1995), 29–36.
35. Rekimoto, J. Tilting operations for small screen interfaces. *Proc. UIST '96* (1996), 167–168.
36. Sanneblad, J. and Holmquist, L. E., Ubiquitous graphics: combining hand-held and wall-size displays to interact with large images. *Proc. AVI '06* (2006), 373–377.
37. Schroeder, D. W., DePue, M. T., Kakarala, R., Xie, T., VanWiggeren, G. D. Tracking motion using an interference pattern, U.S. Patent 7,737,947, 2010.
38. Villar, N., Izadi, S., Rosenfeld, D., Benko, H., Helmes, J., Westhues, J., Hodges, S., Ofek, E., Butler, A., Cao, X., and Chen, B. Mouse 2.0: Multi-touch Meets the Mouse. *Proc. UIST '09* (2009), 33-42.
39. Wang, J. and Canny, J. TinyMotion: camera phone based interaction methods. *CHI '06 EA*, 339-344, 2006.
40. Welch, G. and Foxlin, E. Motion Tracking: No Silver Bullet, but a Respectable Arsenal. *Computer Graphics and Applications*, 22, 6 (2002), 24-38.
41. Wilson, A. and Shafer, S., Xwand: UI for intelligent spaces, in *Proc CHI '03* (2003), 545–552
42. Yang, X.-D., Mak, E., McCallum, D., Irani, P., Cao, X., and Izadi, S. Lensmouse: augmenting the mouse with an interactive touch display, *Proc. CHI '10* (2010), 2431–2440.
43. Yee, K. Peephole displays: Pen interaction on spatially aware hand-held computers. *Proc. CHI '03* (2003), 1–8.