USING MOBILE DEVICES FOR MIXED-REALITY INTERACTIONS WITH EDUCATIONAL LABORATORY TEST-BEDS

BY JARED ALAN FRANK AND VIKRAM KAPILA
DEPARTMENT OF MECHANICAL AND AEROSPACE ENGINEERING
NEW YORK UNIVERSITY TANDON SCHOOL OF ENGINEERING

Recent advances have led to rapid growth in the complexity of technologies. For example, the arrival of systems that couple cyber elements (computing and communication) with physical dynamics are expected to bring revolutionary changes to the way humans learn, live, and work [1]. By interconnecting state-of-the-art interface technologies (e.g., computer vision, interactive graphics, haptics, and motion sensing) with educational laboratory test-beds, a new class of educational platforms can be developed with the potential to transform the way people learn by reducing the effort required to command, control, and monitor physical dynamics [2]. Moreover, rather than turning to specialized, *ad hoc* solutions, educational laboratories can exploit mobile devices, which have evolved to carry an unprecedented amount of sensing, storage, computation, and communication power (SSCC).

With an ever-expanding number of sensors and features, mobile devices can facilitate enhanced interactions with physical systems. Because of their popularity and familiarity, mobile devices can offer learners interfaces that are more intuitive than those on competing, specialized devices. That is, learners may be able to draw upon experience with their personal devices to operate physical systems effectively. Developing architectures and mobile apps to realize such systems requires overcoming the fundamental challenges of how best to use mobile technologies to i) capture and map user behavior to desired behavior of systems and ii) capture and display the system state to support situational awareness [3].

This paper explores a novel human-machine interaction (HMI) paradigm that utilizes the SSCC capabilities of mobile devices to provide intuitive interactions with dynamic systems. Our HMI paradigm addresses the aforementioned fundamental challenges by integrating computer vision, 3D virtual graphics, and touchscreen sensing to develop mobile apps that provide interactive augmented reality (AR) visualizations. While prior approaches used laboratory-grade hardware, e.g., personal computer (PC), vision system,
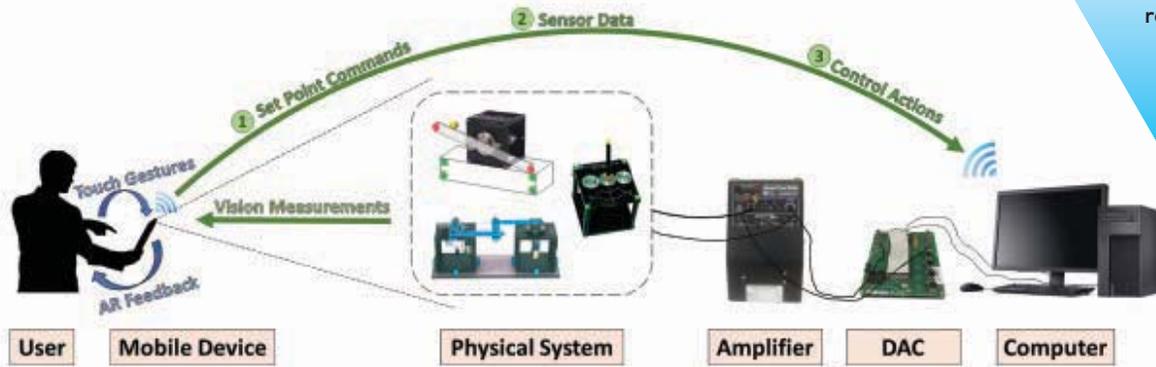
etc., for streaming video to remote users [4], our approach exploits the inherent mobility of mobile devices to provide users with mixed-reality (MR) environments in which the laboratory test-bed and augmented visualizations coexist and interact in real-time to promote immersive learning experiences that don't yet exist in engineering laboratories. After describing the basic components of our approach, we use representative laboratory test-beds to present three architectures that support our HMI paradigm and employ mobile devices in three distinct roles: user interaction, sensing, and control (**Figure 1**). First, interacting with a 2D robot mechanism by manipulating its virtual representation on a tablet allows learners to reinforce their conceptual understanding of spatial relationships and robot kinematics. Second, allowing learners to utilize personal
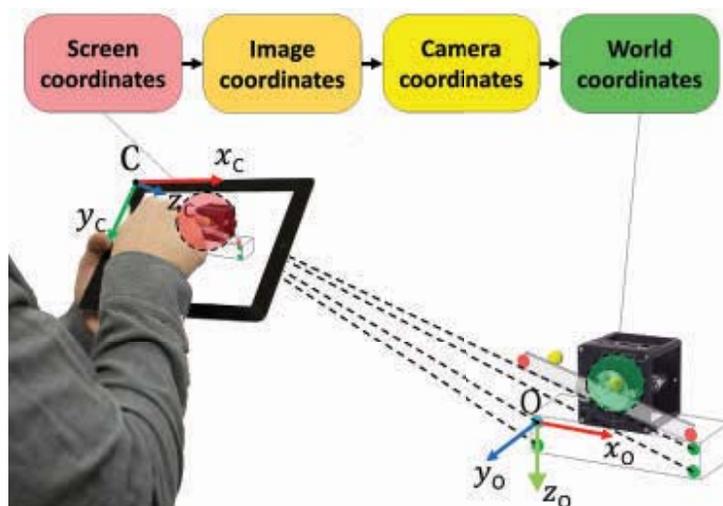

FIGURE 2 Illustration of HMI using color markers and the touchscreen taps.

devices as vision-based sensing tools to interact with a ball and beam system obviates the need to equip the test-bed with alternative sensing instrumentation. Third, providing interactive graphical tools within the interface permits learners to interact with a DC motor arm to command its orientation and interactively redesign a digital controller, thus allowing an exploration of the effects of closed-loop pole locations on dynamic system response characteristics, e.g., stability, damping, and steady-state error. Experimental results demonstrate the performance and efficacy of the presented systems and attempt to address the benefits and challenges of the proposed approach.

APPROACH

Digital cameras and touchscreens constitute two powerful and attractive components of today's mobile devices. When used together, judiciously, highly visual and interactive environments may be realized on the device screen. Using a marker-based vision technique, the relative poses of various objects in the scene can be obtained as a learner points the mobile device from an arbitrary perspective. These poses provide the foundation for measuring positions and orientations of physical elements in the system that can be used to drive realistic projections of virtual elements in the scene and in the feedback control of the system itself. By allowing learners to command real-world elements by manipulating their corresponding virtual elements through touchscreen gestures, the interface becomes transparent, directing users' attention to performing tasks rather than struggling to use the interface. This allows users with relatively little training to operate complex processes with ease, comfort, and delight.

Marker-based Computer Vision

With the mobile device pointed at the system, video captured by the rear-facing camera is processed. Attaching visual markers to critical components facilitates their efficient detection, identification, and localization. Visual markers are detected using color segmentation, a computationally efficient and simple-to-implement approach. After arranging the markers in a plane with a known pattern relative to a specified coordinate frame, a geometric algorithm solves the marker association problem and 2D-3D point correspondences determine the pose of the specified coordinate frame relative to the camera coordinate frame, T_o^c. Once this pose is known, positions and orientations of system components con-
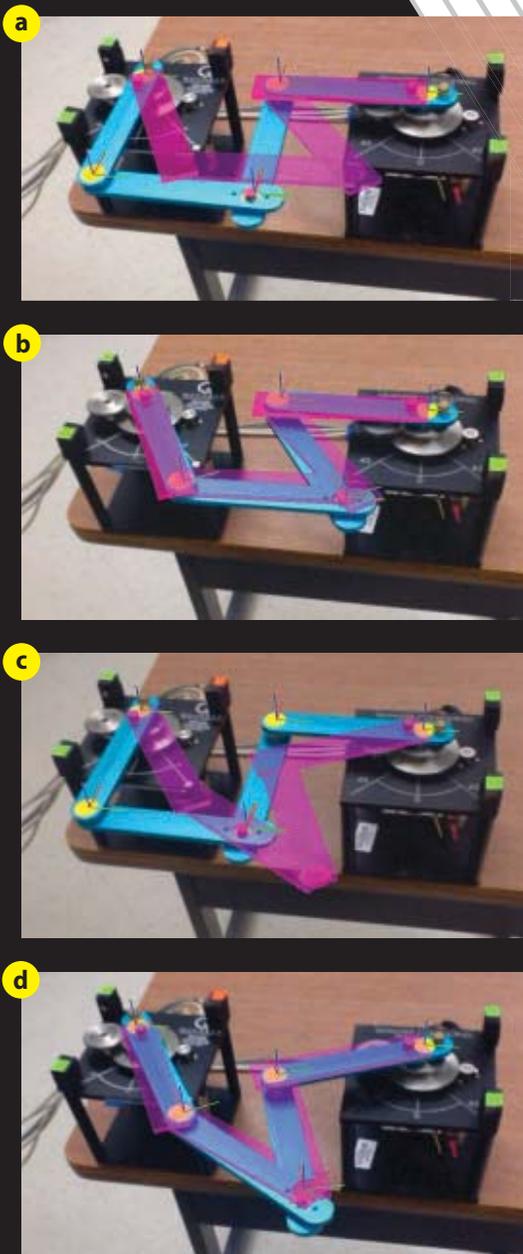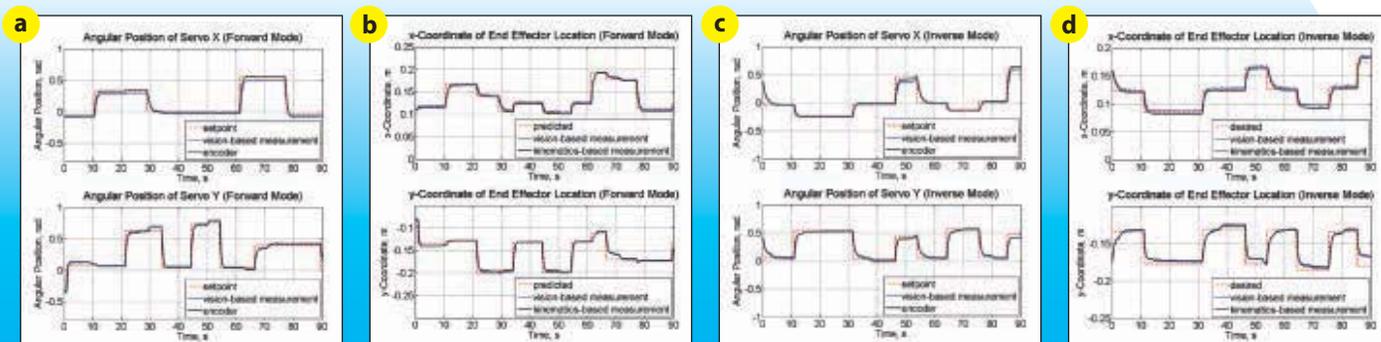
FIGURE 3 Screenshots from interface (a) before and (b) after a command is issued in forward mode, and (c) before and (d) after a command is issued in inverse mode.

FIGURE 4 Plots of issued commands and responses of the system in (a,b) forward mode and (c,d) inverse mode.

tained in the plane are extracted in real-world coordinates relative to the specified coordinate frame **(Figure 2)**.

Intuitive Interaction with AR

Spatial relations extracted through computer vision are used for realistic projection of virtual elements to render the MR environment on the screen of the mobile device. In this environment, the virtual graphics provide stimulating visual feedback to enhance user monitoring of the system and can be manipulated to intuitively operate the system. Although virtual manipulations are composed of 2D gestures on the touchscreen, they correspond to 3D commands for the planar system. For touchscreen taps, transformations are used to determine the location in the plane of the system intended by the user. First, the tap coordinates are converted to image coordinates through a simple resolution conversion. Next, $(T_o^c)^{-1}$ is used to map the image coordinates of the tapped location to coordinates in the plane of the system relative to the specified coordinate frame **(Figure 2)**. Once known, these real-world coordinates are used to send commands to the system and to drive the virtual graphics. By simulating the system's governing equations on the app, the virtual graphics can provide responsive and predictive visualizations of system behavior before commands are issued. This combination of low-latency video from the device camera, extrasensory visualizations afforded by AR, and fluid interactivity provided by the touchscreen can yield new immersive experiences with laboratory hardware.

EXAMPLE ARCHITECTURES

Three HMI implementations illustrate architectures in which the mobile device provides the user interface, performs the sensing, and implements the estimators and controllers for the system **(Figure 1)**. Although the mobile device used in these implementations is the Apple iPad 2, chosen for its larger screen, similar implementations can employ smartphones. Open source libraries are used to process images, render AR, and communicate using the TCP/IP protocol. See [5] for illustrative videos.

2D Planar Robotic Mechanism

The first HMI architecture uses the tablet to provide an immersive user interface to interact with a physical system and is illustrated with a four-link, 2D robot with two links actuated by motors **(Figure 1)** [6]. The architecture is important in applications where educators seek to impart to learners the benefits of both virtual laboratories (e.g., interactive and visually stimulating interfaces) and hands-on laboratories (e.g., real data, real hardware) for enhanced experiential learning. Specifically, the tablet provides visual feedback and communicates commands while control is performed using measurements from the test-bed's embedded encoders. The joint connecting the second and third links acts as an end effector that can be commanded

FIGURE 5 Experimental results for the (a) ball position and (b) beam angle.

to a desired location. The motors are driven by power amplifiers that receive signals from a PC via a data acquisition and control board. The PC computes control actions based on set point commands received over Wi-Fi from the mobile app. Visual markers attached on five fixed columns in the plane and at each joint of the robot enable vision techniques to estimate the robot's plane, joint angles, and end-effector location. These measurements are used to register the AR content in the scene, e.g., coordinate frames attached at the robot joints and a virtual rendering of the robot. The touchscreen interface allows users to command the robot in one of two modes **(Figure 3)**. In the forward mode, the actuated links are controlled by manipulating virtual links and in the inverse mode, the end effector is controlled by manipulating a virtual circle representing the end effector. The app serves as an intuitive interface for users to learn about robot kinematics. To further aid HMI, the robot's forward and inverse kinematic equations are run within the app to compute the desired end effector location from the user-supplied joint angles in forward mode and compute the desired joint angles from the user-supplied end effector location in inverse mode. Thus, the interface
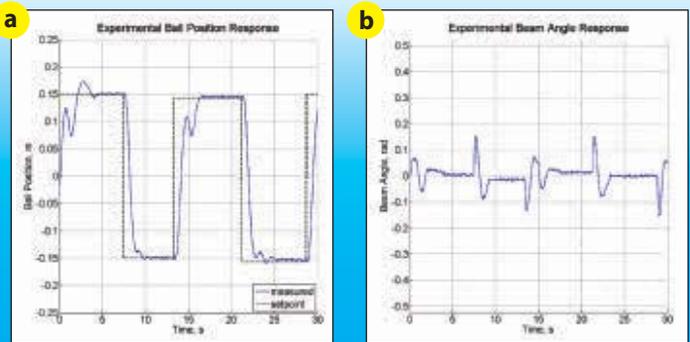
Figure 4d shows the actuated joint angle measurements from camera and encoders, and predicted orientations of the joint angles by propagating the desired end effector location through the inverse kinematics. Sensing limitations, such as uncertainties in camera parameters and limited image resolution, contribute to small discrepancies between the measurements from the tablet camera and those from the laboratory-grade sensors. However, as seen in **Figure 3**, these discrepancies are often negligible.

Ball and Beam System

Experimental results from the previous subsection suggest a second HMI architecture that employs the tablet as a user interface and its vision-based measurements for feedback control [7]. This enables learners to receive the benefits of the first architecture without requiring that the system carry as many laboratory-grade sensors, thus lowering the potential cost and size of the test-bed. The test-bed used is a ball and beam system **(Figure 1)** built from a DC-motor, a 0.5-meter long lexan beam, and a smooth 25.4 mm diameter ball. A PC wirelessly receives sensor data from the tablet and uses it for estimation and control. Visual markers are affixed to the system to estab-
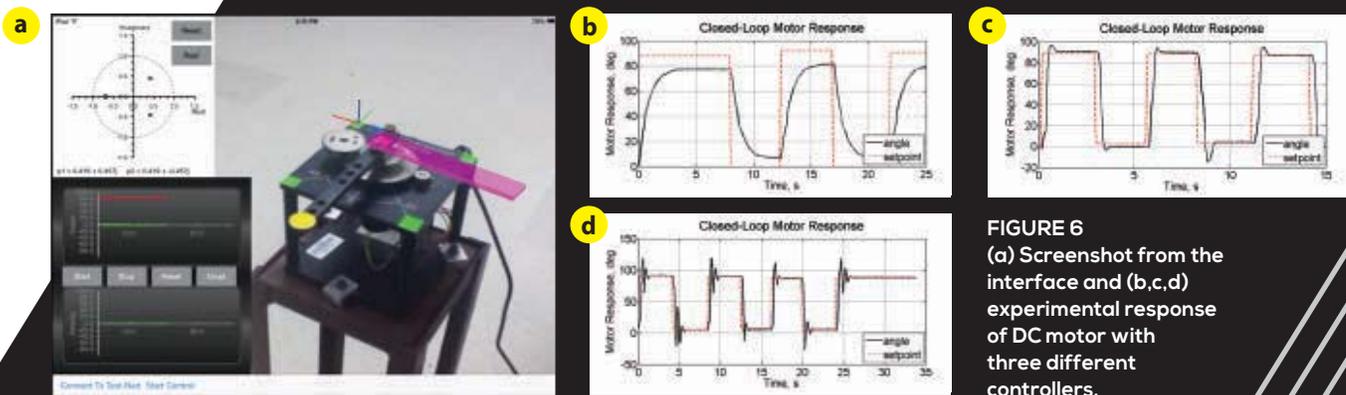

FIGURE 6 (a) Screenshot from the interface and (b,c,d) experimental response of DC motor with three different controllers.

provides predictive visualizations to reveal the configuration of the robot resulting from user interactions before it is driven to the desired configuration. To assess this HMI architecture, the system is experimentally commanded in both forward and inverse modes. **Figure 4a** shows the actuated joint angle measurements from camera and encoders. **Figure 4b** shows the end effector position measurements from camera and encoders, and predicted position of the end effector by propagating the desired motor angles through the forward kinematics. **Figure 4c** shows the end effector position measurements from camera and encoders.

lish a reference frame and to measure beam angle. The ball is colored yellow to track its position. Users command the position of the ball by tapping at desired locations on the beam image **(Figure 2)**. A principal challenge in this HMI architecture is to ensure that noise, resolution, and measurement delay introduced by the tablet's limitations are sufficiently small for closed-loop stability. Experiments are conducted to examine the effects of the vision-based measurements on closed-loop system behavior. Results indicate that image processing can be completed in 13.28 ms (s.d. 0.283 ms), which supports a 60 Hz frame rate. Analyzing vision measurements from various perspectives reveals that ball position can be measured with 0.7 mm precision, beam angle can be measured to within 0.2 degrees, and measurement noise does not vary significantly for different perspectives. **Figure 5** shows the ball position and beam angle response as a user commands the ball to approximately 25% and 75% along the beam length.

DC Motor

The stable response of the system in the previous subsection suggests a third HMI architecture wherein all interface, sensing, filtering, estimation, and control tasks are performed on the mobile device [8]. This allows learners to receive the benefits of the first and second architectures, and, in the spirit of [9], can replace expensive laboratory computers and software with low-cost microcontrollers, since tablets can perform all necessary computations. This architecture is illustrated with position control of a 15.24 cm motor arm **(Figure 1)**. In this implementation, the PC (alternatively a microcontroller) is only responsible for transmitting control signals received from the tablet. Vision sensing is used to measure the angular position of the motor arm. The tablet implements a discrete-time Kalman filter and a full-state feedback controller. The motor control user interface has three main views **(Figure 6a)**. A 30 Hz video is live-streamed in the large, right view, projected onto which is a purple semi-transparent virtual arm lying in the plane of the actual arm, representing the system set point. As the virtual arm is tapped and dragged, it pivots under the user's finger about its fixed end–attached to the orange marker in the scene–similar to the rotation of the actual motor arm about its axis. The two left-hand views of the interface contain plots. On top, users tap on an interactive pole-zero plot to select desired closed-loop poles, triggering a pole-placement formula for controller gain computation. On the bottom, dynamic plots display the set point command, vision-based measurements of the motor arm angular position, and angular velocity estimate from the Kalman filter. Buttons enable the user to start, stop, and reset plots and email collected data for post-processing. The resulting interface allows learners to interactively explore the effect of pole locations on system performance. To investigate the potential of this architecture, the tablet is used to design several controllers and issue 0° and 90° step commands. **Figures 6b-6d** show the issued set points and recorded responses for each control design. These results show that the interface allows users to both command the system and adjust various characteristics of its dynamic behavior, including steady-state error, settling time, and overshoot.

CONCLUSIONS AND FUTURE DIRECTIONS

With mobile devices becoming the primary personal computing devices, this paper presents a novel approach to use them for HMI with laboratory test-beds. By pointing the rear-facing cameras of the mobile devices at the system from an arbitrary perspective, computer vision techniques retrieve physical measurements to render interactive AR content or perform feedback control. For the three suggested architectures, wherein the mobile device provides diverse capabilities of interaction, sensing, estimation, and control, we verify feasibility, investigate performance, illustrate benefits, and uncover challenges of our HMI paradigm. Three experimental implementations demonstrate the potential for providing predictive visualizations to improve situational awareness and graphical controls to adjust the dynamic behavior of the system. Virtual and remote experimentation using mobile devices lacks the benefit of presence in the laboratory with actual equipment. Our HMI paradigm facilitates engaging laboratory learning through hands-on experiences with equipment and interactive visualizations on personal mobile devices. Future work will examine the potential of our approach in teaching fundamentals of dynamic systems, automatic control, robotics, etc., through inquiry-based activities with students. ∎

REFERENCES

1 Rajkumar, R., 2012, "A Cyber–physical Future," *Proc. of the IEEE*, 100, pp. 1309-1312.

2 Garber, L., 2013, "Gestural Technology: Moving Interfaces in a New Direction," *Computer*, 46(10), pp. 22-25.

3 Frank, J.A., and Kapila, V., 2016, "Integrating Smart Mobile Devices for Immersive Interaction and Control of Physical Systems: A Cyber-Physical Approach," *Advanced Mechatronics and MEMS Devices II*, Springer, to appear.

4 Jara, C.A., *et al.*, 2011, "Hands-on Experiences of Undergraduate Students in Automatics and Robotics using a Virtual and Remote Laboratory," *Computers & Education*, 57(4), pp. 2451-2461.

5 NYU, 2016, "Mobile Interfaces to Interact with Laboratory Test-Beds," [Online]. Available: http://engineering.nyu.edu/mechatronics/videos/hmiarchs.html.

6 Frank, J.A., and Kapila, V., 2016, "Towards Teleoperation-based Interactive Learning of Robot Kinematics using a Mobile Augmented Reality Interface on a Tablet," *Proc. Indian Control Conference*, Hyderabad, India, pp. 385-392.

7 Frank, J.A., De Gracia Gomez, J.A., and Kapila, V., 2015, "Using Tablets in the Vision-Based Control of a Ball and Beam Test-bed," *Proc. Int. Conf. Informatics in Control, Automation, and Robotics*, Colmar, France, pp. 92–102.

8 Frank, J.A., Brill, A., and Kapila, V., 2016, "Interactive Mobile Interface with Augmented Reality for Learning Digital Controls Concepts," *Proc. Indian Control Conference*, Hyderabad, India, pp. 85-92.

9 Frank, J.A. and Kapila, V., 2014, "Development of Mobile Interfaces to Interact with Automatic Control Experiments," *IEEE Control Systems Magazine*, 34(52), pp. 78-98.

ABOUT THE AUTHORS

Jared Alan Frank, a mechanical engineering doctoral student at NYU, is conducting research on the development and evaluation of immersive apps for intuitive interactions with physical systems.

Vikram Kapila (vkapila@nyu.edu), a professor of mechanical engineering at NYU, has research interests in mechatronics, robotics, and STEM education.