

# Towards Teleoperation-based Interactive Learning of Robot Kinematics using a Mobile Augmented Reality Interface on a Tablet

Jared A. Frank<sup>1</sup> and Vikram Kapila<sup>1</sup>

**Abstract**—The integration of augmented reality (AR) techniques in user interface design has enhanced interactive experiences in teleoperation of robots, hands-on learning in classrooms, laboratory, and special education, and user training in an array of fields, e.g., aerospace, automotive, construction, manufacturing, medical, etc. However, AR-based user interfaces that command machines and tools have not been fully explored for their potential to enhance interactive learning of engineering concepts in the laboratory. This paper outlines the development of a mobile application executing on a tablet device, which renders an immersive AR-based graphical user interface to enable users to monitor, interact with, and control a four-link underactuated planar robot. Computer vision routines are used to extract real-time, vision-based measurements of the robot's joint angles and end effector location from the live video captured by the rear-facing camera on the tablet. The obtained measurements are used to render AR content to offer users with additional visual feedback. Touch gesture recognition is implemented to allow users to naturally and intuitively command the robot by tapping and dragging their fingers at desired locations on the tablet screen. Experimental results show the performance and efficacy of the proposed system as it is operated in two different modes: one in which the user has direct control over the angles of the actuated links of the robot and one in which the user has direct control over the end effector location.

## I. INTRODUCTION

In engineering education, many concepts that are taught in a traditional classroom setting can be better understood with the use of interactive animated multimedia or hands-on experience. Thus, the standard approaches taken by most curricula and educators have included *i*) providing students with interactive software applications that simulate engineering systems to help visualize concepts and *ii*) allowing students to perform laboratory activities to encourage experiential learning through exposure to real physical systems. Augmented reality, the projection of virtual content onto the real worldview, is used to provide users with visual aids that enhance interaction with physical systems [3] and foster immersive and stimulating learning experiences [6]. Recent years have witnessed several efforts that raise the bar in educational innovation, e.g., by integrating immersive interactions based on AR into the classroom to provide effective and engaging instruction in an array of fields, including, physics [5], chemistry [13], and geometry [11],

and to provide training with advanced laboratory and industrial equipment, such as engines [12] and atomic force microscopes [21].

On the work-site, AR has been leveraged in the teleoperation of robots [8], [19], [20] while in the laboratory, software applications have incorporated AR to enable users to perform visually stimulating remote experiments [16]. However, to improve abilities in the mechanical design and analysis of robotic systems, students require platforms that reinforce their understanding of fundamental concepts in robotics, such as conventions for attaching coordinate frames to the links of a kinematic chain, the derivation of kinematic equations, and in the use of these equations to solve the forward and inverse kinematics problems.

Kinematic chains are important mechanisms that play critical roles in engineering and robotic applications, e.g., serial chains in robotic manipulators, Watt four-bar linkages in vehicle suspension systems, and slider-crank mechanisms in automobile engines. The extremities of humanoids such as Robonaut, NASA's space robot [1], and the legs of bipedal walkers [18], are typically modeled as kinematic chains and designed to be underactuated to reduce power consumption and costs. Some virtual and augmented reality user interfaces have been developed in an effort to help visualize these systems and teach the concepts associated with them, by using dynamic simulations and augmented reality books [14], [15]. However, these prior user interfaces are limited in that they do not provide the benefits associated with controlling a real physical mechanism.

To prepare students for designing and working with such robotic systems, the development of a mobile application with touch-based interaction and AR feedback is proposed for use in an engineering laboratory. The application is tested on a tablet device and allows users to interact with, monitor, and control a four-link planar robot with two actuated links. Specifically, the user simply manipulates the robot's virtual representation, which is overlaid on the live view as it is captured by the rear-facing camera on the tablet. That is, users are offered an intuitive and natural touch-based method to command the angular positions of the actuated links either directly, by tapping and dragging the virtual links to their desired orientations, or indirectly, by touching and dragging the end effector to a desired location. As the user manipulates the virtual components, the forward and inverse kinematics equations are executed by the application to update the AR display of the virtual mechanism. When the user releases his finger from the AR interface, set points are wirelessly transmitted to the physical system to drive the robot mecha-

This work is supported in part by the National Science Foundation awards RET Site EEC-1132482, GK-12 Fellows DGE: 0741714, and DRK-12 DRL: 1417769, and NY Space Grant Consortium grant 48240-7887.

<sup>1</sup>Mechatronics and Controls Lab, Mechanical and Aerospace Engineering, NYU Tandon School of Engineering, Brooklyn, NY 11201 [jared.alan, vkapila]@nyu.edu

nism to the desired configuration. The AR virtual mechanism can be used to observe the spatial relationships between attached coordinate frames, the forward effect that changes in angles of the actuated joints have on the configuration of the mechanism (e.g., the location of the end effector), as well as the inverse effect that changes in position of the end effector have on the configuration of the mechanism. Control-related phenomena, such as settling time, steady-state error, and oscillations about the set point, may also be observed and used to tune the control gains. The result is an immersive lab experience that can engage students and allow them to conduct inquiry-based investigations with physical equipment using their personal tablet devices. Although this study investigates the development and performance of a mobile user interface to provide such an immersive experience, an evaluation of its effectiveness for student learning will be examined in future work.

The paper is organized as follows. In Section II, an overview of the proposed system, including the test-bed, are presented. Section III outlines the design of the user interface. Section IV discusses the forward kinematic, inverse kinematic, and dynamic modeling of the test-bed, as well as the control design. Section V describes the image processing routine executed on the mobile application to obtain vision-based measurements required for displaying AR content. Section VI presents results of the experiment to evaluate the performance of the proposed AR-based user interface while Section VII offers some concluding remarks.

## II. SYSTEM DESCRIPTION

The system used in this study is a laboratory test-bed composed of a planar four-link mechanism with two joints actuated by DC motors (see Figure 1). The links are attached using joints and the motors are clamped to a lab desk for structural support such that the motors' output shafts are a distance of  $2l_a$  apart from each other, where  $l_a$  denotes the link length for each of the four links in the mechanism and  $l_a = 125$  mm. The joint that attaches the second and third links is treated as the end effector of the system, and the objective is to accurately place the end effector at a desired location by controlling the angular position of the two motors. The configuration depicted in Figure 1 is the home configuration of the robot. It is about this configuration that the effect of the two motors are approximately decoupled (see Section IV).

The motors are driven by power amplifiers, which receive control signals from a desktop computer via a data acquisition and control (DAC) board. The desktop computer hosts and runs the MATLAB/Simulink environment, and performs the computation of control actions based on feedback from encoders and tachometers of the motor test-beds and set points received over Wi-Fi from the tablet mobile application. The mobile application captures video from the rear-facing onboard camera as it is pointed at the experiment from an arbitrary perspective. Colored markers affixed to the test-bed are detected by an image processing program running on the application, and used to obtain vision-based

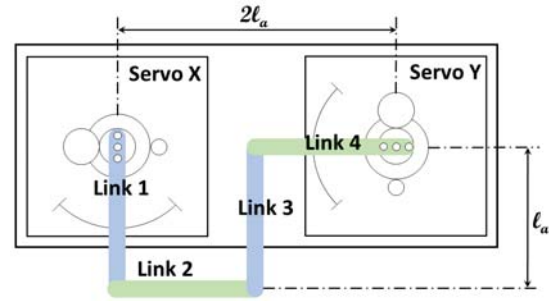


Fig. 1: Figure illustrating the planar robot in its home configuration.

measurements of the configuration of the links to display AR content such as rigidly attached coordinate frames. The components that comprise the system are shown in Figure 2.

## III. USER INTERFACE

The tablet device used in this study is an Apple iPad 2, which has a 9.7 inch (250 mm), 1024×768 pixel multi-touch display, 1 GHz dual-core processor, and a 0.7-megapixel rear-facing camera. Open source and third-party libraries, such as Open Source Computer Vision Library (OpenCV), Open Source Graphics Library (OpenGL), and CocoaAsyncSocket are used to process video frames, render AR content, and communicate over TCP/IP, respectively.

The user interface developed for this work (see Figure 3) consists of a window in which a live video stream from the rear-facing camera on board the tablet is shown at 30 frames per second. Projected onto this view are purple virtual links that lie in the horizontal plane with the links of the actual planar robot in the video and represent the desired configuration for the mechanism. Marker-based image processing techniques are used to register the augmented content in the view and constrain it to real world references, for example the coordinate frames that are rigidly attached to each link. The user may command the configuration of the test-bed by tapping on the touchscreen and dragging his finger on the screen to manipulate the virtual mechanism. This touch-based interaction operates in two alternative modes.

In the first mode, called forward mode, the two actuated links are highlighted and the user may tap either one to select it and drag his finger along the screen to modify the link's orientation about the motor shaft. As the user drags his finger, the equations of forward kinematics are used to update the display of the virtual mechanism to show the user how altering the angle of the link will effect the configuration of the entire mechanism. In the second mode, called inverse mode, a circle is highlighted in the horizontal plane that represents the desired location of the end effector. While dragging his finger across the screen, the user changes this desired location, and inverse kinematics equations are used to compute the required angular orientations of the actuated links and update the display of the virtual mechanism.

When the user releases his finger from the touchscreen, the desired actuated joint angles are wirelessly transmitted to the test-bed, whether they are computed directly from the user's touch gestures in the forward mode or via the inverse

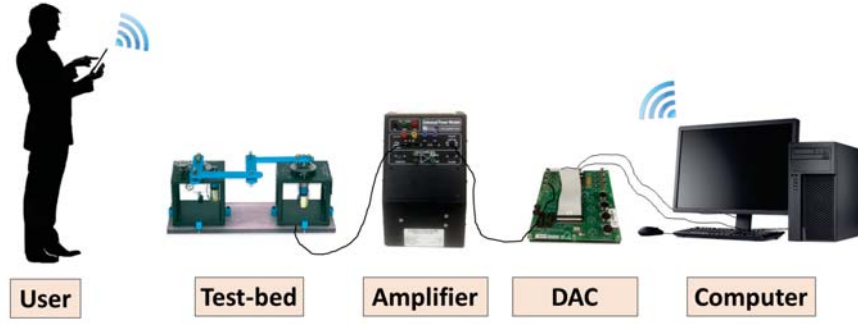


Fig. 2: Schematic representation of the proposed system for interacting with a planar robot.

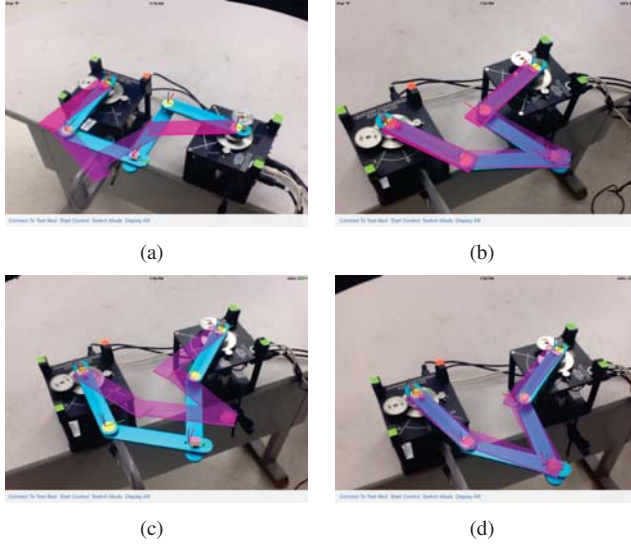


Fig. 3: Screenshots from the interface (a) before a command is issued in forward mode, (b) after a command is issued in forward mode, (c) before a command is issued in inverse mode, (d) after a command is issued in inverse mode.

kinematics equations in the inverse mode. Although touches on the touchscreen are not in the plane of the mechanism, a simple transformation allows for the natural manipulation of the AR virtual system (see Section V). Four buttons are available at the bottom of the interface to wirelessly connect and disconnect to the test-bed, to enable and disable the sending of commands to the test-bed, to switch between the two modes of interaction, and to turn on and off the display of AR content.

#### IV. MODELING AND CONTROL

The kinematic and inverse kinematic modeling framework used in this paper is adopted from [2] and is presented below for completeness.

##### A. Forward Kinematics

To analyze the kinematic relationships of the four-link mechanism, six coordinate frames are affixed to the mechanism (see Figure 4(a)). These coordinate frames include two fixed frames ( $F_1$  and  $F_6$ ) attached to the motor shafts, two moving frames ( $F_2$  and  $F_5$ ) attached to the ends of links 1 and 4, and two moving frames ( $F_3$  and  $F_4$ ) attached to the

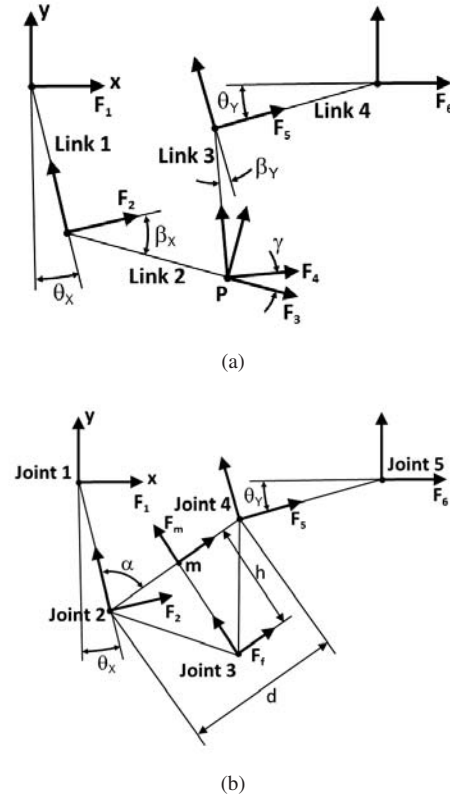


Fig. 4: Diagrams illustrating the links, joints, and coordinate frames attached to the mechanism.

ends of links 2 and 3 whose origins are coincident with each other. To accurately depict the predicted configuration of the planar robot as a function of the angular positions of the motors,  $\theta_x$  and  $\theta_y$ , the transformation matrices between the coordinate frames are written as

$$\begin{aligned} T_{12} &= T(\theta_x, l_a, -(\frac{\pi}{2} - \theta_x)), & T_{23} &= T(-\beta_x, l_a, -\beta_x), \\ T_{16} &= T(0, 2l_a, 0), & T_{65} &= T(\theta_y, l_a, -(\pi - \theta_y)), \\ T_{54} &= T(-\beta_y, l_a, -(\frac{\pi}{2} + \beta_y)), & T_{43} &= T(-\gamma, 0, 0), \end{aligned}$$

where

$$T(\theta, l, \beta) \triangleq \begin{bmatrix} \cos(\theta) & -\sin(\theta) & l \cos(\beta) \\ \sin(\theta) & \cos(\theta) & l \sin(\beta) \\ 0 & 0 & 1 \end{bmatrix}.$$



We now need to solve the forward kinematics problem to determine the location of the end effector relative to the fixed coordinate frame  $F_1$ , given the desired values of  $\theta_x$  and  $\theta_y$ . The desired values of  $\theta_x$  and  $\theta_y$  will be available on the interface directly from the user interactions with the virtual actuated links. Knowing  $\beta_x$ ,  $T_{13} = T_{12}T_{23}$  can be computed to yield the location of the end effector and similarly knowing  $\beta_y$ ,  $T_{14} = T_{16}T_{65}T_{54}$  can be computed to yield the location of the end effector. Unfortunately,  $\beta_x$  and  $\beta_y$  are not known. Thus, following [2], we utilize forward kinematic equations resulting from geometric relationships associated with a triangle connecting the ends of the actuated links 1 and 4 and the end effector (see Figure 4(b)). To describe these geometric relationships, two additional coordinate frames are assigned: a frame  $F_m$  attached to the midpoint between Joints 2 and 4 and a frame  $F_f$  attached such that its origin is coincident with frames 3 and 4 and its orientation is aligned with frame  $F_m$ . Denoting the distance between the two new frames ( $F_m$ – $F_f$ ) as  $h$  and the distance between Joints 2 and 4 as  $d$ , the following transformation matrices are immediate

$$T_{2m} = T\left(\left(\frac{\pi}{2} - \alpha\right), \frac{d}{2}, \left(\frac{\pi}{2} - \alpha\right)\right), \quad T_{mf} = T\left(0, h, -\frac{\pi}{2}\right).$$

Now the location of the end effector can be found in terms of  $\theta_x$ ,  $\theta_y$ ,  $d$ ,  $\alpha$ , and  $h$ . Calculating  $d$ ,  $\alpha$ , and  $h$  is straightforward and follows from applying the Pythagorean theorem and the law of cosines to the geometry of Figure 4(b)

$$d = \sqrt{({}^1P_{2,x} - {}^1P_{4,x})^2 + ({}^1P_{2,y} - {}^1P_{4,y})^2},$$

$$\alpha = \arccos\left(-\frac{|{}^1P_4|^2 - |{}^1P_2|^2 - |d|^2}{2|{}^1P_2||d|}\right),$$

$$h = \sqrt{l_a^2 - \left(\frac{d}{2}\right)^2},$$

where  ${}^1P_2 = ({}^1P_{2,x}, {}^1P_{2,y})$  is the origin of  $F_2$  attached at Joint 2 with respect to  $F_1$ , found from  $T_{12}(1:2,3)$  and  ${}^1P_4 = ({}^1P_{4,x}, {}^1P_{4,y})$  is the origin of  $F_5$  attached at Joint 4 with respect to  $F_1$ , found from  $T_{15} = T_{16}T_{65}(1:2,3)$ . After  $d$ ,  $\alpha$ , and  $h$  are calculated, the predicted location of the end effector with respect to the fixed coordinate frame  $F_1$  is found by evaluating  $T_{2m}$  and  $T_{mf}$  and applying the transformation  $T_{1f} = T_{12}T_{2m}T_{mf}$ .

### B. Inverse Kinematics

To accurately position the end effector of the planar robot at a desired location, the necessary angles of the actuated links corresponding to the desired location must be known. Therefore, the inverse kinematics problem is solved to allow the system to compute the joint angles for a particular location of the end effector.

The coordinates of the desired location for the end effector,  $P$ , are expressed with respect to the fixed Frame 1 attached to the first motor shaft,  ${}^1P = (P_x, P_y)$ , and with respect to the fixed Frame attached to the second motor shaft,  ${}^6P = (\hat{P}_x, \hat{P}_y)$ . Since the  $y$ -coordinate of the origin of each of the two fixed frames are the same, we obtain  $\hat{P}_x = P_x + 2l_a$ ,  $\hat{P}_y = P_y$ . These

locations are used to obtain the following solution to the inverse kinematics problem, as presented in [2]

$$\theta_x = 2\text{atan}\left(\frac{2l_aP_x - \sqrt{4l_a^2P_x^2 - P_x^4 - 2P_x^2P_y^2 - P_y^4 + 4l_a^2P_y^2}}{(P_x^2 + P_y^2 - 2l_aP_y)}\right),$$

$$\theta_y = -2\text{atan}\left(\frac{2l_a\hat{P}_y + \sqrt{4l_a^2P_y^2 - \hat{P}_x^4 - 2\hat{P}_x^2P_y^2 - P_y^4 + 4l_a^2\hat{P}_x^2}}{(\hat{P}_x^2 + P_y^2 - 2l_a\hat{P}_x)}\right).$$

### C. Dynamic Model

The system used in this study is composed of two DC motor test-beds, each of which contains a gearbox and output shaft. We assume that the proposed system is tested in a small neighborhood about the home configuration depicted in Figure 1. This is because, in this configuration, Links 2 and 3 can be assumed to undergo pure horizontal and pure vertical translation in response to small rotations of Servo X and Servo Y, respectively. That is, when the left servo, Servo X, is given a small positive angular displacement and the right servo, Servo Y, is held at its zero orientation, the end effector is given a small displacement in the positive  $x$ -direction and the displacement in the  $y$ -direction is negligible. Similarly, a small positive angular displacement in Servo Y, while holding Servo X at its zero orientation, results in a small displacement of the end effector in the negative  $y$ -direction with negligible  $x$ -direction displacement.

This behavior of the system is confirmed through two separate experiments along each of the coordinate axes, with results shown in Figure 5. The end effector is manually moved along the  $x$ -axis while measuring the angular position of Servo Y using the encoder of the motor test-bed. By plotting the angular deviation of Servo Y against the  $x$  position of the end effector, we find a region in which Servo X can freely control the  $x$ -coordinate of the end effector with negligible effect on Servo Y orientation (see Figure 5(a)). The result is a narrow region about the  $x$ -coordinate of the end effector's home location in which the effect on Servo Y orientation is negligible. As the distance from the home location increases, a boundary is reached beyond which Servo Y begins to exhibit large angular deviations as the  $x$ -coordinate of the end effector is changed. Constraints on the  $x$ -direction movement of the end-effector are defined at the points at which the deviation of Servo Y become nontrivial, which are approximately 5 cm from the home location in either direction. The boundary constraints for the  $y$ -direction are found in a similar manner, and yield similar results (Figure 5(b)).

Thus, the dynamic behavior of the system may be treated as the decoupled behavior of two independent motor test-beds in a small 5 cm  $\times$  5 cm rectangular region around the home configuration of the system. The dynamics can be represented using first-order transfer functions from the input voltages  $V_x(s)$  and  $V_y(s)$  to the angular velocities of the shafts  $\Omega_x(s)$  and  $\Omega_y(s)$ , respectively, as follows

$$\frac{\Omega_x(s)}{V_x(s)} = \frac{K_x}{\tau_x s + 1}, \quad \frac{\Omega_y(s)}{V_y(s)} = \frac{K_y}{\tau_y s + 1}.$$

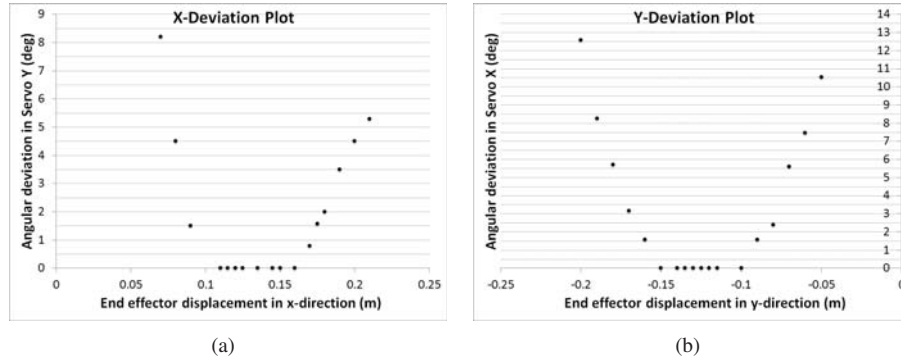


Fig. 5: Plots of experimental data used to locate the allowable region of the uncoupled dynamic model.

In these models,  $K_x$  and  $K_y$  are steady-state DC-gains and  $\tau_x$  and  $\tau_y$  are time constants associated with the step response of Servo X and Servo Y, respectively. These parameters are determined experimentally by applying unit-step inputs to each of the motors and choosing values that characterize the observed responses,

$$\begin{aligned} K_x &= 1.4597V, & \tau_x &= 0.035s, \\ K_y &= 1.4885V, & \tau_y &= 0.046s. \end{aligned}$$

#### D. Control Design

To control each motor such that the actuated links are oriented at the desired angles, proportional-integral-derivative (PID) controllers are designed according to the block diagram shown in Figure 6. Through block manipulation, the closed-loop transfer function is determined as follows

$$\frac{\Omega(s)}{R(s)} = \frac{\frac{KK_p}{\tau}s + \frac{KK_i}{\tau}}{s^3 + \frac{1+KK_d}{\tau}s^2 + \frac{KK_p}{\tau}s + \frac{KK_i}{\tau}}.$$

To design the dynamic response of this third-order system, closed-loop poles are placed such that the desired characteristic polynomial consists of two dominant stable poles and a third stable real pole located twice as far from the imaginary axis as the real part of the dominant poles

$$\begin{aligned} \lambda(s) &= (s^2 + 2\zeta\omega_n s + \omega_n^2)(s + 2\zeta\omega_n) \\ &= s^3 + 4\zeta\omega_n s^2 + (1 + 4\zeta^2)\omega_n^2 s + 2\zeta\omega_n^3, \end{aligned}$$

where  $\zeta$  and  $\omega_n$  are the damping ratio and natural frequency associated with a second-order system with poles corresponding to the dominant poles. Comparing this characteristic polynomial to that obtained by block manipulations, the values of the control gains can be expressed in terms of the plant dynamics (i.e.,  $K$  and  $\tau$ ) and the desired closed-loop dynamics (i.e.,  $\zeta$  and  $\omega_n$ )

$$K_p = \frac{(1 + 4\zeta^2)\omega_n^2 \tau}{K}, \quad K_i = \frac{2\zeta\omega_n^3 \tau}{K}, \quad K_d = \frac{4\zeta\omega_n \tau - 1}{K}.$$

The desired motor responses do not contain oscillations and are slow enough for users of the tablet application to visually observe as the mechanism is driven toward the desired configuration. Therefore, a settling time of 1 second

and a 0% overshoot are chosen, which, by plugging into the expressions for settling time and percent overshoot [7], gives  $\zeta = 1, \omega_n = 2$  rad/s. However, the transfer functions for the closed-loop dynamics of the motors reveal a finite zero located relatively near to the dominant poles. This will introduce profound oscillations to the response of the motors [7]. To compensate for this phenomenon, the desired value of  $\zeta$  is doubled to overdamp the system, and the following PID controller gains are calculated

$$\begin{aligned} K_{p,x} &= 1.6305, & K_{p,y} &= 2.1014, \\ K_{i,x} &= 0.7673, & K_{i,y} &= 0.9889, \\ K_{d,x} &= -0.3014, & K_{d,y} &= -0.1774. \end{aligned}$$

#### V. COMPUTER VISION

The tablet-based interface allows users to observe the mechanism from practically any perspective and still obtain vision-based measurements of joint locations and joint angles to accurately embed AR content, which includes coordinate frames fixed to the joint locations of the actual mechanism and predicted configurations of the mechanism as the user interacts with the virtual mechanism. After the user's taps and drags are registered on the touchscreen, these locations can be mapped to locations and orientations in the plane of the observed scene for natural touch-based interaction with the system. These features are made possible by the use of colored markers that are affixed to the test-bed and lie in a common plane in the scene.

##### A. Homography

Stored in the mobile application is a reference image of the test-bed taken from a perspective in which the image plane is oriented approximately parallel to the plane of the test-bed containing colored markers, namely the plane of the top surfaces of the links in the mechanism (see Figure 7). The coordinate frame of this image is used as the reference frame for image-based measurements of marker locations. First, the coordinates of the centers of four green markers are detected in the reference image and matched with the coordinates of the four green markers in the arbitrary perspective of the scene captured by the tablet's camera. The markers, which are attached to small vertical posts fixed at the outside

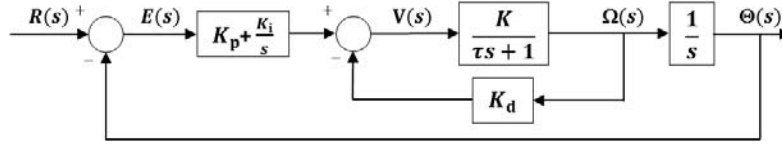


Fig. 6: A block diagram showing the design of the PID controllers.

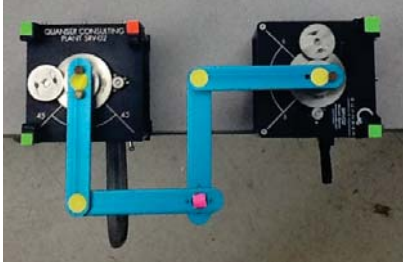


Fig. 7: Reference image used to calculate the homographies associated with arbitrary perspectives.

corners of the motor enclosures, are coplanar with the 2-D plane containing the yellow and pink markers affixed to the surfaces of the links. Next, following [9], we can compute a projective homography matrix,  $G$ , which represents a transformation between a point  $p = (u, v, 1)$  in pixels in the arbitrary image frame and the point  $p^* = (u^*, v^*, 1)$  in the reference image frame, up to a scale factor  $\alpha_g$ ,

$$\alpha_g p = G p^*.$$

The inverse of this matrix is used to transform the coordinates of markers from the video frame to the reference frame for measuring the locations and angles of various joints. This matrix inverse is also used as the transformation to map locations in the image touched by the user to their corresponding locations in the reference frame.

### B. Marker Detection

To compute a homography matrix for performing perspective transformations and conducting vision-based measurements of the locations and angles of joints, markers of four different colors are affixed to the test-bed. The markers are detected in the video captured by the device camera by segmenting the frames by color. The coordinates of the centers of markers are obtained using the concept of area moments. Experiments reveal that the majority of the required computation performed by the application is devoted to detecting the colored markers. Therefore, to maintain a frame rate of 30 Hz, a resolution of  $352 \times 288$  is used to keep computation time low. Moreover, rectangular regions of interest are created that are centered around clusters of markers detected in the previous frame, and serve as the search space for the markers in the following frame. These regions of interest are obtained by determining the minimal up-right bounding rectangle around the pixels that comprise the markers in the binary images. Similar techniques have been shown to reduce computation time in vision-based traffic applications [10].

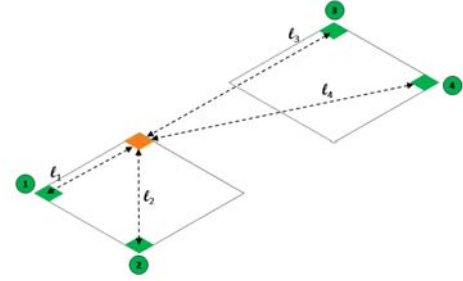


Fig. 8: Diagram of the markers placed on the top surfaces of the motor enclosures.

On the corners of the motor enclosures, an orange marker and four green markers are strategically placed to allow for clear visibility. Since markers may not be detected in the same order in a given frame or from a given perspective, the green markers are compared to each other by using their distances to the orange marker. By placing the orange marker as shown in Figure 8, the four green markers can be uniquely identified from most reasonable downward-facing perspectives of the tablet after they are sorted in order of increasing distance from the orange marker. This allows the green markers to be matched between the video frames and the reference image so that the homography matrix is calculated reliably. The homography matrix can then be used to obtain the projected coordinates with respect to the reference frame of the pink marker affixed to the end effector and each yellow marker affixed at the joints of the mechanism.

### C. Joint Location and Angle Measurements

After the homography matrix is calculated, the coordinates of the center points of the yellow and pink markers are transformed through its matrix inverse to determine the projected coordinates of the joints and end effector with respect to the reference frame. The differences between these locations and the location of the first green marker are used to render virtual coordinate frames attached to the joints and virtual links that pivot about the motor axes (see Section V-D). To orient the coordinate frames and links, joint angles can be determined in radians by taking the difference between the coordinates of two adjacent markers and using the inverse tangent function. If the centers of markers attached to the two ends of a link have projected coordinates  $p_1^* = (x_1^*, y_1^*)$  and  $p_2^* = (x_2^*, y_2^*)$  with respect to the reference frame, the joint angle corresponding to that link is calculated as

$$\theta = -\tan^{-1} \left( \frac{y_2^* - y_1^*}{x_2^* - x_1^*} \right).$$

#### D. Camera Pose for Rendering Augmented Reality Content

The rendering of well-registered AR content in the scene is critical for yielding immersive interactive experiences while teleoperating the 2D planar robot system. To overlay a 3D virtual mechanism of the planar robot onto frames as they are captured by the camera, poses of the links must be known relative to the coordinate frame of the camera. These poses are homogeneous transformations composed of rotation matrices and translation vectors in the Euclidean coordinate. Rendering the virtual links and coordinate frames such that they appear in the plane with the actual links requires the construction of a 3D coordinate system in the image with its origin located at the center of the first green marker. After expressing the locations of the remaining three markers in this coordinate system, calibrating the device camera [22], and solving a correspondence problem between the 2D projected locations and 3D locations of the markers [4], the pose of the camera is estimated with respect to the 3D coordinate system. To render the 3D content, this camera pose is inverted to obtain the pose of the 3D coordinate system with respect to the camera frame.

### VI. EXPERIMENTAL RESULTS

To assess the feasibility of using the proposed system to promote teleoperation-based interactive learning with the planar robot, an experiment is performed in which the tablet is used to command the system in both the forward and inverse modes. In the forward mode, the virtual actuated links are manipulated several times to issue commands to the test-bed. Figure 9(a) shows the set points for the motor angles generated during the interaction, which are issued by the tablet as the user interacts with the virtual links by tapping and dragging them to desired orientations. Figure 9(a) also shows measurements of the response of the motors, obtained from image processing on the tablet as well as from the encoders on the motor shafts. Next, Figure 9(b) shows the end effector's predicted  $x$  and  $y$  coordinates, which result from the user interaction with the actuated links by using the desired motor angles to propagate the forward kinematics equations on the tablet. Figure 9(b) also shows measurements of the response of the actual end effector, obtained from the image processing routine running on the tablet as well as by feeding measurements from the encoders on the motor shafts through the forward kinematics equations.

While interacting in the inverse mode, a virtual circle representing the set point for the end effector is manipulated several times to issue commands to the test-bed. Figure 10(a) shows the end effector's desired  $x$  and  $y$  coordinates, which in this mode are directly determined by the locations on the tablet screen touched by the user. Figure 10(a) also shows measurements of the response of the end effector of the test-bed, obtained from the image processing routine as well as from the propagation of forward kinematics equations with measurements from the encoder on the motor shafts. Next, the desired coordinates of the end effector are fed through the inverse kinematic equations on the tablet to compute the corresponding set points for the motors, which

are shown in Figure 10(b). Moreover, Figure 10(b) shows the measurements of the response of the motors, obtained from both the image processing routine and the encoders.

There are steady-state errors present in both motor angles and end effector location. Such errors are due to unmodeled mechanical phenomena such as deadband in the motor, backlash between gears, and stiction in the joints that connect the links of the mechanism. Moreover, there are small discrepancies between vision-based measurements and measurements obtained from encoders attached to the motors due to uncertainties in camera calibration and uncertainties in conversion factors between image coordinates and real world coordinates. Non-ideal scene illumination, limited image resolution, imperfect image quality, and noise in the image sensor all contribute to errors and uncertainties in the locations of the centers of colored markers and therefore variation in the calculation of the homography matrix. The mobile user interface allows the cumulative effect of these errors to become visible to the user as a discrepancy between the real and augmented mechanisms. Finally, note that as the user provides a set point command by manipulating the augmented mechanism, the response of actual mechanism displayed on the user interface provides a means to visualize resulting oscillations and steady-state error.

It has been experimentally determined that a small variable communication delay is present between the mobile user interface and computer, with a mean length of 28.7 milliseconds and standard deviation of 38.9 milliseconds. However, since this delay is on the order of one sampling interval and it effects only the reference command, it does not impact the closed-loop stability of the system. A video capturing the user experience of both the forward mode and inverse mode of the mobile application is available at [17].

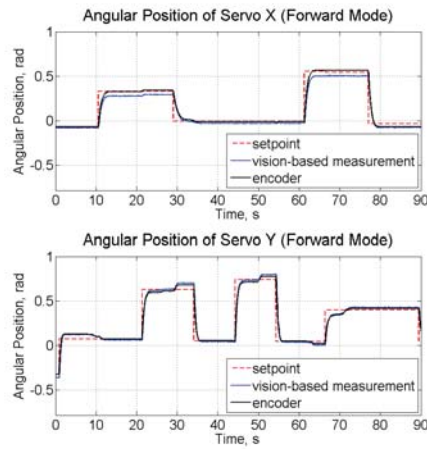
### VII. CONCLUSION

This paper presented a user interface on a tablet device that used its rear-facing camera, augmented reality, and computer vision techniques, as well as forward and inverse kinematics equations, to provide immersive interactive experiences while teleoperating a planar robot. By allowing users to directly manipulate a virtual representation of the kinematic chain, the goal is to reinforce ideas regarding spatial relationships associated with moving coordinate systems and conceptual understanding of forward and inverse kinematics. Future work will involve a comparative study assessing the educational potential of the proposed system in teaching these fundamental concepts in robotics through inquiry-based activities with engineering students.

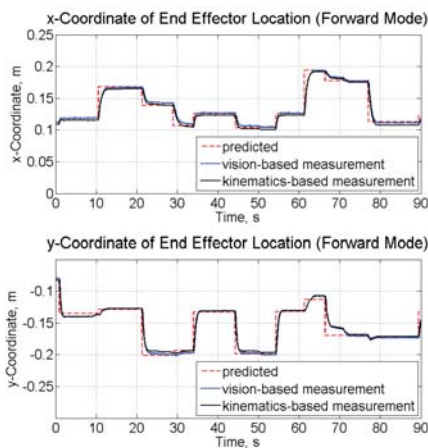
### REFERENCES

- [1] R. O. Ambrose *et al.*, "Robonaut: NASA's space humanoid," *IEEE Intelligent Systems*, vol. 15, no. 4, pp. 57–63, 2000.
- [2] J. Apkarian, "A comprehensive and modular laboratory for control systems design and implementation," *Quanser Consulting*, 1995.
- [3] R. T. Azuma *et al.*, "A survey of augmented reality," *Presence*, vol. 6, no. 4, pp. 355–385, 1997.
- [4] D. Baggio, *Mastering OpenCV with Practical Computer Vision Projects*. Packt Publishing Ltd, 2012.





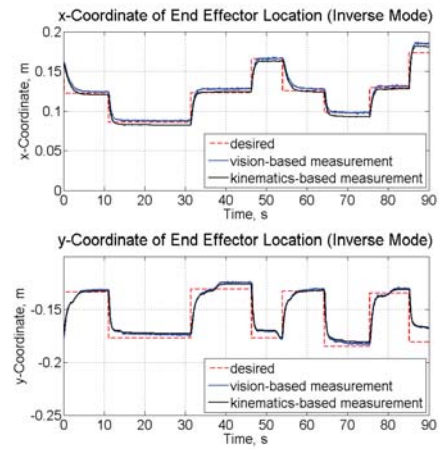
(a)



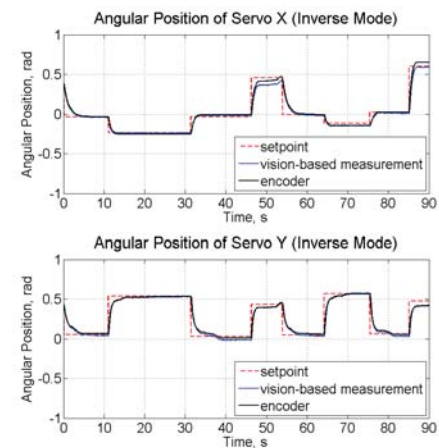
(b)

Fig. 9: Plots of issued commands and responses of the system in forward mode.

- [5] M. Billinghurst and A. Duenser, "Augmented reality in the classroom," *Computer*, no. 7, pp. 56–63, 2012.
- [6] C. Dede, "Immersive interfaces for engagement and learning," *Science*, vol. 323, no. 5910, pp. 66–69, 2009.
- [7] R. Dorf and R. Bishop, *Modern Control Systems (11th ed.)*. New Jersey: Prentice Hall, 2007.
- [8] A. Halme, J. Suomela, and M. Savela, "Applying telepresence and augmented reality to teleoperate field robots," *Robotics and Autonomous Systems*, vol. 26, no. 2, pp. 117–125, 1999.
- [9] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003.
- [10] V. Kastrinaki, M. Zervakis, and K. Kalaitzakis, "A survey of video processing techniques for traffic applications," *Image and Vision Computing*, vol. 21, no. 4, pp. 359–381, 2003.
- [11] H. Kaufmann and D. Schmalstieg, "Mathematics and geometry education with collaborative augmented reality," *Computers & Graphics*, vol. 27, no. 3, pp. 339–345, 2003.
- [12] F. Liarokapis *et al*, "Web3d and augmented reality to support engineering education," *World Transactions on Engineering and Technology Education*, vol. 3, no. 1, pp. 11–14, 2004.
- [13] P. Maier, M. Tönnis, and G. Klinker, "Augmented reality for teaching spatial relations," in *Conference of the International Journal of Arts & Sciences*, 2009.
- [14] W. Maqableh and M. Sidhu, "Interactive four-bar linkage 3d-simulation using augmented reality," in *Int. Conf. on Information Technology*, 2013.
- [15] W. F. Maqableh and M. S. Sidhu, "Preliminary evaluation of using augmented reality as a dynamic simulation learning tool for linkage



(a)



(b)

Fig. 10: Plots of issued commands and responses of the system in inverse mode.

- mechanisms," *International Journal of Computer Applications*, vol. 58, no. 4, pp. 19–23, 2012.
- [16] R. Marin, P. J. Sanz, and J. Sanchez, "A very high level interface to teleoperate a robot via web including augmented reality," in *Proc. IEEE Int. Conf. Robotics and Automation*, vol. 3. IEEE, 2002, pp. 2725–2730.
- [17] Mechatronics and Control Laboratory. (2015) Video capturing user experience of mobile application for interacting with 2D robotic mechanism. [Online]. Available: <http://engineering.nyu.edu/mechatronics/videos/teleop.html>.
- [18] J. Meneses *et al*, "Kinematics and dynamics of the quasi-passive biped pasibot," *Strojniški vestnik-Journal of Mechanical Engineering*, vol. 57, no. 12, pp. 879–887, 2011.
- [19] P. Milgram, A. Rastogi, and J. J. Grodski, "Telerobotic control using augmented reality," in *Robot and Human Communication, 1995. RO-MAN'95 TOKYO, Proceedings., 4th IEEE International Workshop on*, 1995, pp. 21–29.
- [20] P. Milgram *et al*, "Applications of augmented reality for human-robot communication," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, vol. 3, 1993, pp. 1467–1472.
- [21] W. Vogl, B. K.-L. Ma, and M. Sitti, "Augmented reality user interface for an atomic force microscope-based nanorobotic system," *IEEE Transactions on Nanotechnology*, vol. 5, no. 4, pp. 397–406, 2006.
- [22] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.