Light Painting with 3D Printed Robotic Arm

Xiongyi Cui, Peng He, Brian Hetherman, Savanna Horan, Christopher Mehegan, Murphy Wonsick, and Qizong Wu

***Abstract*—The purpose of this project is to design, simulate, print, construct, program and utilize a 5 DOF robotic arm for the purpose of creating "light painting" photographs. In order to accomplish this, we will make use of the full range of forward and inverse position and velocity kinematics. We propose to design, simulate, and realize end effector trajectories to produce predictable and desirable final photographs, the nature of which will demonstrate mastery of the robot's kinematics. This arm should be viewed as a prototype with potential future use as an educational tool in the teaching of robot kinematics and dynamics.**

I. Introduction

LIGHT painting is a technique used by photographers in which either a light source or a camera is moved while a long exposure photograph is taken. The effects created by this technique can be stunning and unexpected, and are generally delivered unaltered by computer manipulation. Some examples of factors which can affect the characteristics of a light painting are the intensity, speed, and smoothness of movement of the light source being manipulated. An example of light painting is shown in Figure 1.

We proposed, as our class project for Robot Dynamics, to design, model, simulate, and build a small, low-cost robotic arm using 3D printing technology and easily obtainable components. The team was successful in this endeavor, and was able to utilize this arm in the creation of several light painting photographs. The end effector of the robot arm was composed of a series of lights, which were moved through various paths in order to create the desired final photographs. We believe that, with certain alterations and expansions, this concept could be an excellent tool in the teaching of robot kinematics and dynamics. Because the quality of the final photograph is largely determined by factors controlled by the robot's kinematics and dynamics, this robot provides a method of learning where mastery of concepts is immediately rewarded by striking images. The purpose of this paper is to document the progression of the project, to explain the steps taken in order to achieve our deliverables, and to showcase our results. Additionally, areas for future development of this project will be discussed at the end of this paper.

II. Literature Review

Because this project was not particularly oriented towards research, but rather involves a demonstration of the team's mastery of concepts covered in class on a newly designed robotic platform, there is not a great deal of literature review to be done. We are, of course, relying heavily on the kinematics and dynamics concepts discussed in depth in [1], [2], and [3]. Additional resources were not found to be necessary during the execution of this project, though future work on this concept will likely involve significantly more research. It should be observed that we have made every effort to use the same system of notation as put forward by Spong et al. in [3].

All authors are currently with the Worcester Polytechnic Institute. Any questions regarding this paper should be directed to Christopher Mehegan at cjmehegan@wpi.edu.

Fig. 1. An example of Light Painting. Source: youtube.com

III. Requirements

A. Budgetary Requirements

Because this project was completely funded from within the group, there was a fairly rigid budgetary constraint. At the inception of this project, we determined that the robot arm, in its entirety, shall not cost greater than $150.00. This constraint did limit the quality of some of the key components of the robot arm. These components did, in fact, impact the final deliverables. These limitations will be addressed in detail later in the appropriate sections of this paper. The table below shows a table of our supplies and the associated costs.

Item	Cost per Unit	Quantity
Servos	$3.00	5
Arduino Mega	$60.00	1
Printing Material	$5.00	1
Miscellaneous	$20.00	1

Our total cost for this project was approximately $100. We were able to finish significantly under budget, in spite of requiring multiple printing cycles. Our costs were also lower than expected because we did not have to buy a power source.

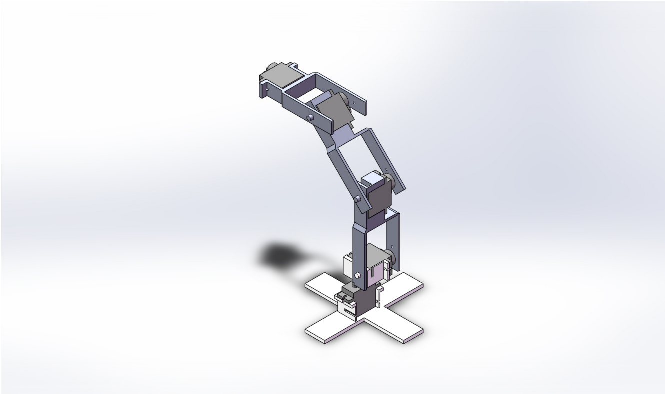

Fig. 2. Final Solidworks Model.

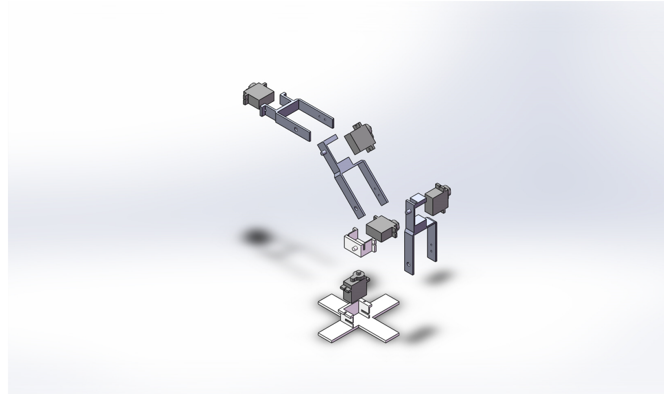

Fig. 3. Final Solidworks Model (exploded view).

While some of the desired functionalities of the robot arm could not be realized because of the quality of the purchased components, we were able to achieve the desired goal of creating light painting photographs, even considering the rigid budgetary constraints. A list of suggested upgrades and their expected impact on future deliverables can be found in the Future Development section of this paper.

B. *Design Requirements*

In our preliminary investigation of this project, we identified several important requirements that our system must meet, and were prepared to construct a system with six degrees of freedom. We determined, however, that only five degrees of freedom were necessary in order to properly manipulate the light on the end effector to create the desired effects. As such, we simplified our project's design in order to meet our objective while still remaining on budget. As can be seen in Figure 2, the robot's wrist lacks one of the degrees of freedom necessary to be properly spherical. The final servo allows the tool tip to be rotated. When combined with a tool tip that offsets a series of lights, the manipulability of the final photographs was increased. The effects this produces are quite appealing, and can be seen in the Results section of this paper.

IV. Design

After viewing several previously constructed 3D printed robot arms, we began to design our arm in Solidworks. The final model, shown in Figures 2 and 3, represent work that was done both before and in conjunction with the kinematic/dynamic analysis and the hardware selection phases of the project. Once it became clear that we would be able to use the same micro servos at each joint, we were able to finalize our arm design.

In order to properly complete the design phase of the project, the servos themselves had to be modeled in Solidworks. This model can be seen in Figure 4. After the servo was modeled, we were able to design a servo cage, which would then be fit into repeatable links. Figure 3 shows an exploded view of our final design. The components that needed to be printed were then converted to assembly files and sent to the printer.

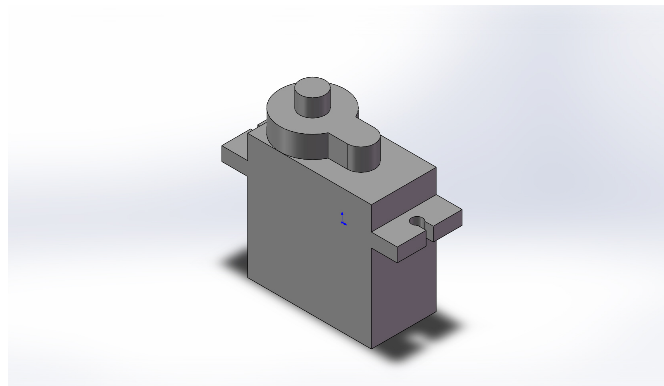

Fig. 4. Micro Servo Model.

One of the main considerations in the design of this arm was in keeping the links relatively open. This served two purposes. Firstly, the open links allowed for servos to be easily swapped in and out of the arm in the event of motor failure. Secondly, the design of the links was made to be easily executed by a hobbyist 3D printer.

V. Kinematics

A. *Forward Kinematics*

As we have done many times in class, the forward kinematics began with the assignation of reference frames and the creation of Denavit-Hartenberg parameters. The D-H parameters for this robot can be seen in the table below.

Link	θ	**d**	**a**	α
1	q1	L1	0	$\pi/2$
2	q2	0	L2	0
3	q3	0	L2	0
4	q4	0	L2	$-\pi/2$
5	q5	L_e	0	0

These parameters were utilized to generate the homogeneous transformation matrices from link to link. Using Equation 1 below, we were able to acquire the transformation from the base frame to the tool tip.

$$T_e^0 = T_1^0 T_2^1 T_3^2 T_4^3 T_e^4 \tag{1}$$

B. Inverse Kinematics

The inverse kinematics were more difficult to generate than the forward kinematics. We were, however, able to simplify the problem somewhat by use of kinematic decoupling. In order to make the inverse kinematics easier to solve, the arm was broken into three separate parts. Link 1 causes a rotation of the end effector about the Z axis of the base frame. Links 2, 3, and 4 comprise a 3 degree of freedom planar arm. Taken together, these three links cause a rotation of the end effector about the Y axis of the base frame. Finally, the last link rotates the end effector about the base frame's X axis. All three of these parts are necessary if joint angles are to be determined by a desired position. Equations 2-15 below shows the inverse kinematics for our robot arm.

$$S_1 = sin(\theta_1) \tag{2}$$

$$C_1 = cos(\theta_1) \tag{3}$$

$$\varphi = \theta_1 + \theta_2 + \theta_3 \tag{4}$$

$$P_{3x} = L_1C_1 + L_2C_{12} + L_3C_{123} \tag{5}$$

$$P_{3y} = L_1S_1 + L_2S_{12} + L_3S_{123} \tag{6}$$

$$P_{2x} = P_{3x} - L_3C_{123} \tag{7}$$

$$P_{2y} = P_{3y} - L_3S_{123} \tag{8}$$

$$C_2 = \frac{(P_{2x})^2 + P_{2y})^2 - L_1^2 - L_2^2}{2L_1L_2} \tag{9}$$

$$S_2 = \pm\sqrt{1 - C_2^2} \tag{10}$$

$$\theta_2 = atan2(S_2C_2) \tag{11}$$

$$C_1 = \frac{(L_1 + L_2C_2)P_{2x} - L_2S_2P_{2y}}{(P_{2x})^2 + (P_{2y})^2)} \tag{12}$$

$$S_1 = \frac{(L_1 + L_2C_2)P_{2y} - L_2S_2P_{2x}}{(P_{2x})^2 + (P_{2y})^2)} \tag{13}$$

$$\theta_1 = atan2(S_1C_1) \tag{14}$$

$$\theta_3 = \varphi - \theta_1 + \theta_2 \tag{15}$$

An important aspect of these equations for inverse kinematics is that they can return invalid joint configurations. Figure 5 shows two separate mathematically correct solutions for a single desired tip position. It can be clearly seen, however, that the top configuration would cause the arm to collide with itself, and is therefore not a valid joint configuration for that tip position.

In order to eliminate these collisions, we incorporated a simple line intersection checker to determine if any of the links (approximated by a straight line from joint to joint) were intersecting. We were able to do this because we knew that the arm would always be operating in the same plane given our design.

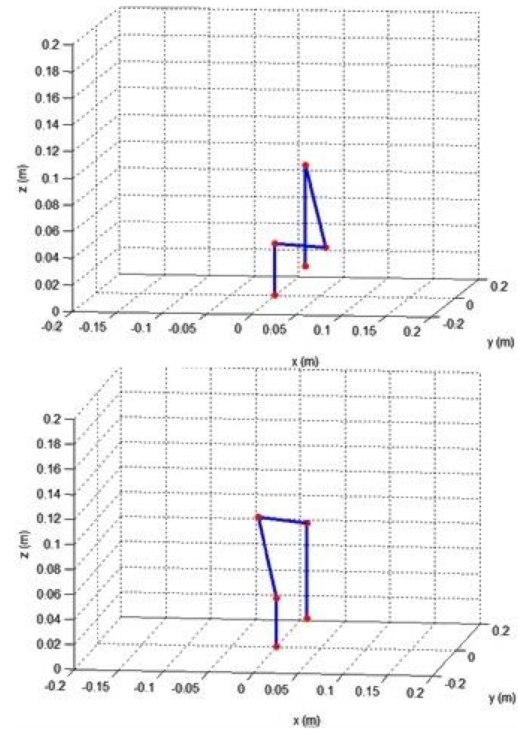

Fig. 5. Two solutions to the Inverse Kinematics problem, top configuration invalid.

VI. Dynamics

Although we were not able to create a torque based controller for our robot arm, we did perform a basic analysis of our robot's dynamics. In future iterations of this robot, where torque control will become far more important, this will be an important area of research. Our analysis utilized the Euler-Lagrange approach described in [3], and assumed that all links were point masses located at the center of mass of the link.

The position kinematics were used to determine the positions of each joint, and the potential energies were calculated from these positions. The kinetic energies were determined by calculating the Jacobian based on the derivative of the position kinematics. Once the energies were acquired, the Lagrangian was calculated and the generalized equations of motion for the robot arm were generated using the method shown in Equations 16 and 17 below, where L is the Lagrangian and K and P are the total kinetic and potential engergies, respectively.

$$L = K - P \tag{16}$$

$$\tau = \frac{d}{dt}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} \tag{17}$$

The generalized forces generated by implementing this method with our robot's parameters produced equations that are of a size unsuitable to this paper. The code necessary to do this has been attached to this paper submission, as have screenshots depicting the final torque outputs.

VII. Hardware

After careful investigation of the robot's design, including kinematic and dynamic analyses, we were able to generate a finalized list of all necessary hardware properties. Ultimately, the component selections were, as mentioned above, largely based on cost.

A. *Printer*

We printed the links, supports, and servo cages for this robot using a Wanhao Duplicator 4S 3D Printer, currently owned by a team member. In the creation of the components, several prints were necessary in order to properly house all the servos while maintaining clearance for wires and other hardware. This is discussed in more detail in the Issues Encountered section of this paper.

B. *Servos*

All of the servos for this arm will be VicTsing SG90 9G micro servo motors. These have been selected based on their relatively low cost, and the robot arm was designed so as not to exceed the limitations of these servos.

During the initial design phase, we had planned to incorporate a more powerful servo into one or more of the joints. However, after performing the dynamics calculations for the robot, it became clear that we would not be affected by the torque limitations on the micro servos. While surprising, this fact did allow us to simplify the design of the arm, and is one of the main reasons that we were able to get results while remaining significantly under budget.

VIII. Simulate and Solve

In parallel to and following the design phase, our team utilized MATLAB and Simulink software to simulate the movement and tip trajectories of the robot arm. Figure 6 shows our final Simulink model, which takes in the final arm design via SimMechanics to animate the detailed model of our arm. The inputs for the Simulink model are the joint angles, angular velocities, and angular accelerations. We utilized Cubic Polynomial trajectory generation for each joint. We did explore Quintic Polynomial trajectory generation, but were unable to implement this on the actual robot due to the limitations imposed by our servo selection. Because the servos only accept joint angle inputs, both the Cubic and Quintic Polynomial trajectories were not possible.

In simulation, we first moved the tool tip from the Home Configuration, shown in Figure 7 to the initial point in our desired trajectory. Following this, the tip was moved through a desired trajectory. An example of a circular trajectory is presented in Figure 8. A video of this simulation in action has been included in the submission of this paper.

IX. Trajectory Generation

One concept that evolved during the simulation phase of the project was that a desired end effector trajectory could be drawn onto a 2D plane using OpenCV. This path could then be projected into the 3D workspace of the robot, discretized into points, and converted into joint angles using Inverse Kinematics. Figure 9 shows the projection of an OpenCV drawing into the simulated workspace of the robot. This approach did, however, run into unforeseen complications. These were largely caused by the quality of the components we purchased, and are discussed in the Issues Encountered section of this paper.

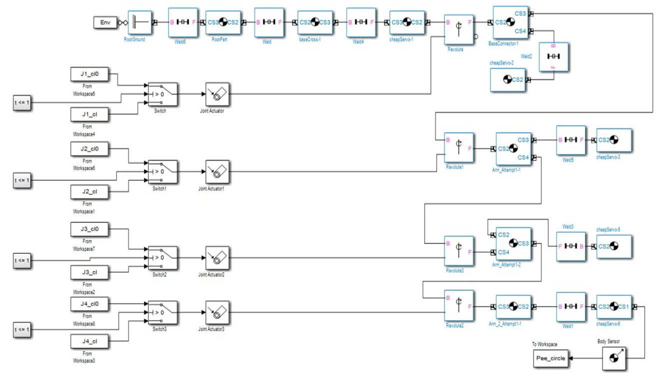

Fig. 6. System Simulink Model.

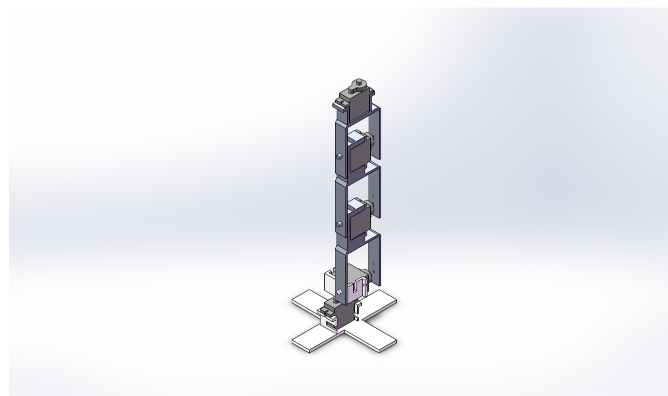

Fig. 7. Home Configuration.

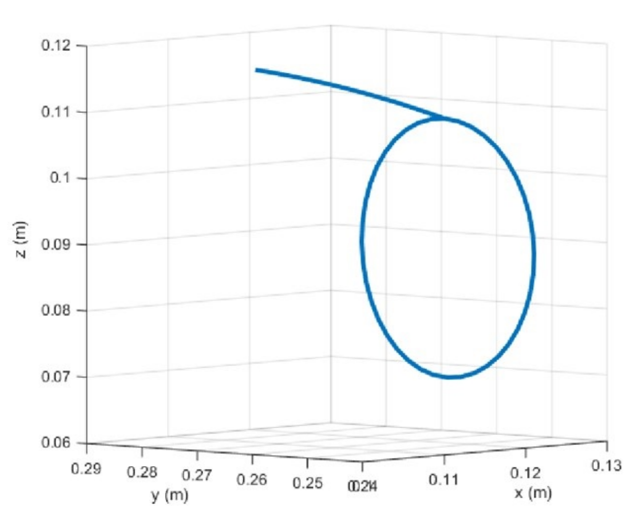

Fig. 8. Simulated end effector path.

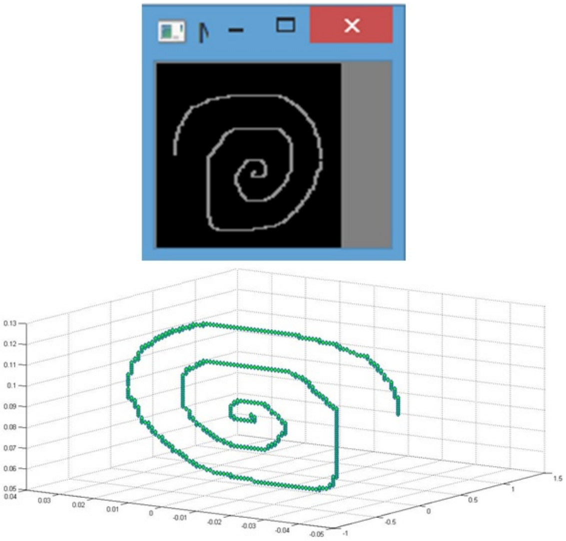

Fig. 9. Trajectory drawing and projection.

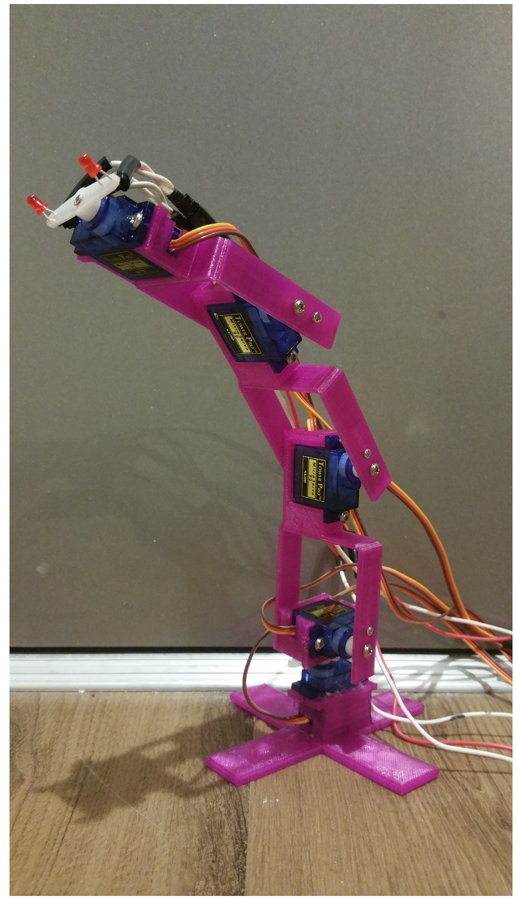

Fig. 10. Constructed Arm.

X. Construction

The construction phase of the project involved several steps. The parts designed in Solidworks were printed on our teams Wanhao Duplicator 4S 3D Printer, and were checked for consistency and accuracy, as well as compatibility with the purchased hardware. This phase of the project, perhaps predictably, was the largest source of delays for the entire project. Multiple prints of all components were made, and necessary alterations to the design were executed during this phase. This is discussed at more length in the Issues Encountered section of this paper.

The purchased hardware was evaluated, and limitations caused by this hardware were noted. The main limitation we discovered during this phase was that the servos we selected only accepted joint angles as their input. This impacted several aspects of our final product, and is discussed further in the Issues Encountered and Future Development sections of this paper.

After printing and the evaluation of all of the components was complete, the arm itself was constructed. The hardware was fit to the structure, and the electrical components were wired to the Arduino board. The final setup of the arm is shown in Figure 10 We were able to borrow a WPI-owned power supply, and this was used to power both the servos and the lights at the end effector.

XI. Synthesize Software

In order to execute planned trajectories, the microcontroller was linked to a laptop running MATLAB. The code necessary to run the robot in this manner has been attached to this paper submission.

XII. Test

After construction of the platform was complete and the connection to MATLAB was functioning properly, we were able to execute several end effector trajectories. During this phase of the project, several adjustments were made in order to properly align reference frames. It was also in this phase that we observed that our trajectory generation program produced angles that were too small to be actuated by the inexpensive servos we had purchased. This is discussed at greater length in the Issues Encountered section of this paper.

XIII. Execution

After testing to ensure that the platform is operating within desired parameters, we generated a series of light painting photographs. These photos were digitally captured using a Fujifilm X100s camera, and have been delivered without digital alteration.

XIV. Issues Encountered

A. Design and Printing Issues

Perhaps predictably, several delays for the project occurred during the process of printing the arm components. Complications during this phase included interference with wires and other servo hardware, size and fidelity issues with the printer, and links that were too thin and flexible to allow for quality construction. These problems were corrected by redesigning several aspects of the links and servo cages until prints of acceptable quality were produced. These issues did cause delays in testing and execution, but were solved in time to produce the final deliverables.

B. Dynamics Issues

Applying torque control to this robot would have been ideal, and our calculation of the robot's dynamics would have allowed us to design and implement a torque controller. However, due to the constraints placed on us by the servo selection, we were unable to do so. The servos cannot be controlled in this manner, and only accept joint angles as inputs. While this issue should have been anticipated during the design phase, doing so would not necessarily have changed any aspect of the project. Our primary constraint was our budget, and we feel that the concept and potential usefulness of this project has been proved. Future iterations of this project should absolutely take this into consideration, though, and this topic will be addressed in the Future Development section of this paper.

C. Trajectory Issues

The selection of the micro servos caused an additional limitation on what our team was able to execute with the final implementation of the robot. The trajectories generated by drawing in OpenCV, discussed in the Trajectory Generation section above, were produced by discretizing points along the hand drawn path. However, the differences in joint angles generated by this program were far too small for the servos to actuate. We attempted to compensate for this by subsampling the trajectory, enlarging the joint angle difference between each movement. This, however, introduced significant vibration and point overshoot of the arm. We believe that higher quality actuators would not have faced this issue, and this is noted in the Future Development section of this paper.

D. Lessons Learned

As a team, we learned several important lessons during the progression of this project. Perhaps the most notable lesson was that printing and construction, while not the main focus of the class or the project, was the area in which the most issues were encountered. It was vital that, during these delays, we continued to work on simulation and trajectory generation. It is perhaps axiomatic that the creation of a new piece of hardware will encounter unforeseen issues, and these delays need to be worked into the schedule of the project from the beginning.

XV. Results and Deliverables

Deliverables for this project include:

A. CAD Files

The CAD files necessary for 3D printing all of the robotic arm parts have be provided along with detailed information about any purchased hardware.

B. Simulation Files

A simulation of the arm in the MATLAB/Simulink environment, coupled with the above CAD file via SimMechanics has been submitted.

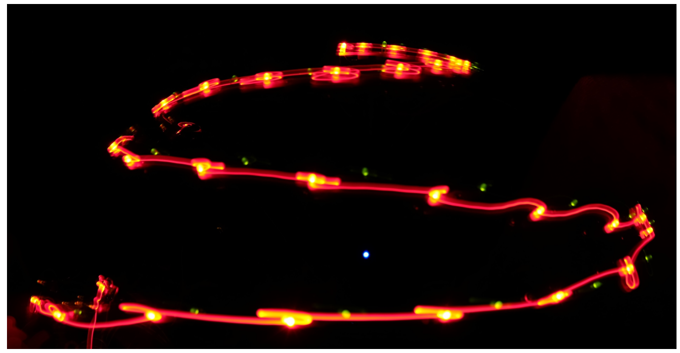

Fig. 11. Zig-Zag trajectory.

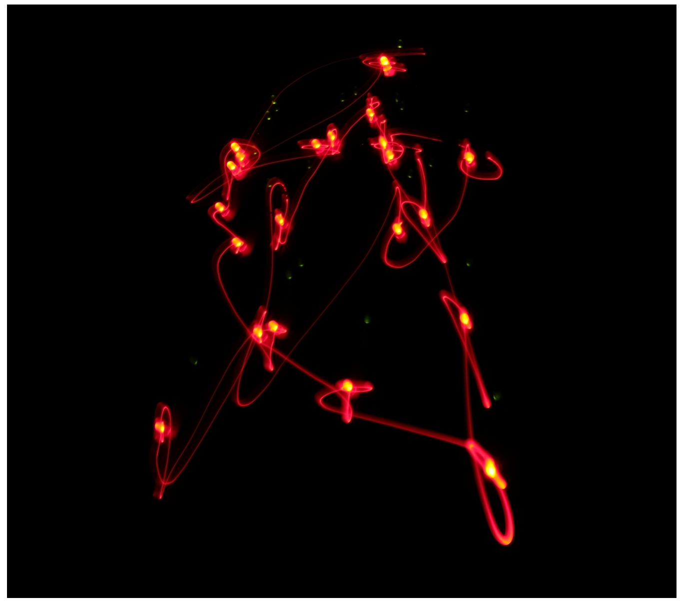

Fig. 12. Star trajectory.

C. Kinematics and Dynamics

The code necessary to generate the robot's kinematics and dynamics equations has been submitted.

D. Robot arm

A fully realized and functional robotic arm has been constructed and demonstrated.

E. Light Paintings

Figures 11 through 14 show a a variety of light paintings executed with this robot. Full size and resolution photographs have been included in this submission.

XVI. Future Development

This project, we feel, could become an excellent educational tool for the teaching of robot kinematics and dynamics. In order to best achieve this goal, we include below a partial list of suggested future developments. The goals for these developments are chiefly concerned with increasing the capabilities of the robot arm in order to allow the concepts learned in RBE 501 to be executed and demonstrated while creating higher quality light paintings.

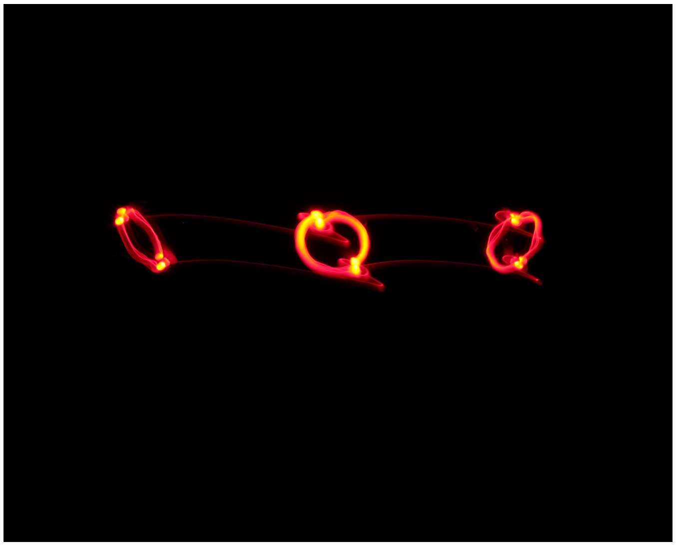

Fig. 13. Three Circles trajectory.

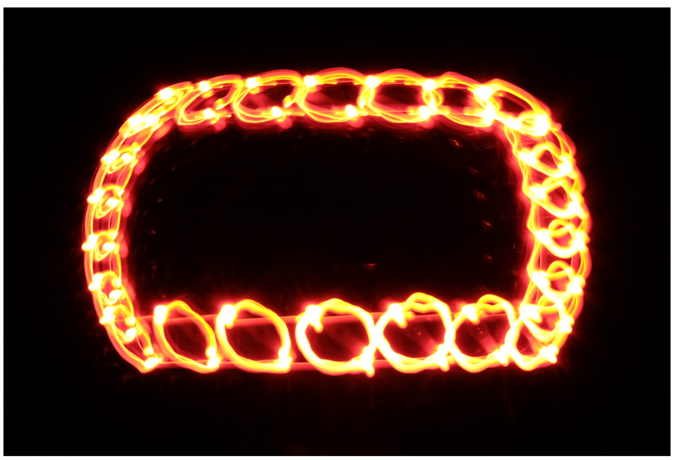

Fig. 14. Rotary trajectory.

A. Higher Quality Components

Utilizing higher quality servos or DC motors with encoders would significantly increase the efficacy of this robot arm. While the cost increase would not be insignificant, and the links and other aspects would need to be redesigned to accomodate the different motor sizes and weights, the benefits of this would be highly impactful on the usefulness of the final product as a teaching tool. Trajectory generation, smoothness of path, and torque control are all aspects of the class that would be teachable on this platform with the addition of higher quality components.

B. Light Emitter Alterations

A second interesting development for the application of creating interesting light paintings would be alterations to the light emitter at the end effector. For this project, we simply utilized a pair of red LEDs, spaced by the servo arm. Different designs of the end effector, perhaps involving a panel of LEDs in different colors, would allow for highly interesting image generation.

C. Wireless Control

A third future development for this project would be to utilize a small computer such as a Raspberry Pi to allow wireless control of the robot arm.

D. Intuitive Control

Expanding on the wireless control, another future expansion would be to incorporate intuitive control of the arm utilizing a joystick or other controller.

E. Gripper

A final sugggestion for future development would be to replace the light emitter at the end effector with a gripper. This would allow the robot to be used for more traditional applications, such as moving small objects from one point to another. This development could involve computer vision and edge detection as well.

REFERENCES

[1] Craig, John J. *Introduction To Robotics*, Reading, Mass.: Addison-Wesley Pub. Co., 1986

[2] Schilling, Robert J. *Fundamentals Of Robotics*. Englewood Cliffs, N.J.: Prentice Hall, 1990.

[3] Spong, Mark W., Hutchinson, Seth, and Vidyasagar, M. *Robot Modeling and Control*, Hoboken, NJ: John Wiley & Sons, 2006.