Robot Arm Using Origami Twisted Tower Design

Jaesung Oh [1], Eumin Hong [2]

[1] PhD Candidate, Hubo Lab, KAIST Korea Advanced Institute of Science and Technology, Seoul South Korea
[2] Horace Mann School, 231 West 246th Street, Bronx, NY 10471
milouhong45@gmail.com

Abstract: The origins of origami can be traced to Asia, where paper was folded to create children's toys, ceremonial gifts, and works of art. However, origami currently plays a different role: utilizing the flexibility and rigidity of paper, scientists implement origami to solve various engineering problems. The origami design chosen to be studied was the origami twisted tower, which is capable of stretching, compressing, and bending, in a way similar to that of the human arm.
[Jaesung Oh, Eumin Hong. **Robot Arm Using Origami Twisted Tower Design.** *Researcher* 2018;10(11):21-24]. ISSN 1553-9865 (print); ISSN 2163-8950 (online). http://www.sciencepub.net/researcher. 4. doi:10.7537/marsrsj101118.04.

Keywords: robotics, design, origami

1. Introduction

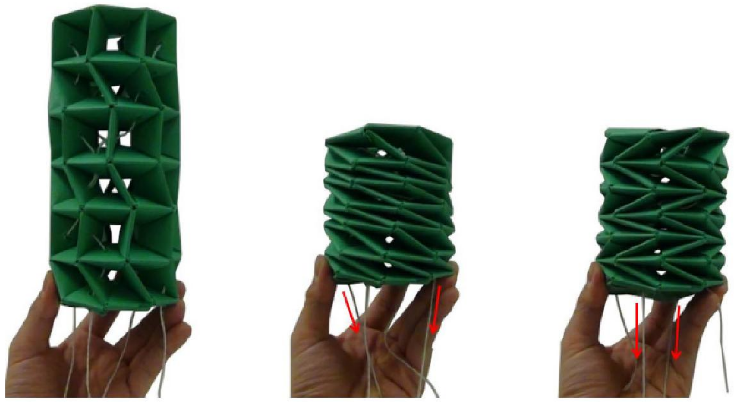
The motions of the origami twisted tower, designed by Mihoko Tachibana, are similar to the human arm as they are both capable of extending and rotating. This tower, whose basic structure is an octahedral prism, is made out of origami units (figures a and b). The tower itself (figure c) is ten units tall and has a circumference of eight units. The origami twisted tower has several unique properties, one of which is that a horizontal twisting motion generates vertical displacement capable of fully extending or compressing the tower (figures c and d). While both clockwise and counterclockwise rotations result in the same vertical displacement, the orientation of the top layer is determined by the rotational direction of each layer. When one of the eight sides of this origami tower is compressed, the tower bends (figure e). Another interesting property of the twisted tower is that the diameter of the structure does not change while being extended or contracted. The following sections will be about folding the twisted tower structure, integrating with the actuator, and reviewing the final robot.

2. Origami Twisted Tower

(a) (b) (c) (d) (e)

<Origami twisted tower structure and configuration>

(a): folding diagram for a single unit
(b): folded unit
(c): fully stretched origami tower
(d): fully compressed origami tower
(e): bent origami tower
*folding instructions: http://origamimaniacs.blogspot.com/2012/02/twisted-tower-by-mihoko-tachibana.html



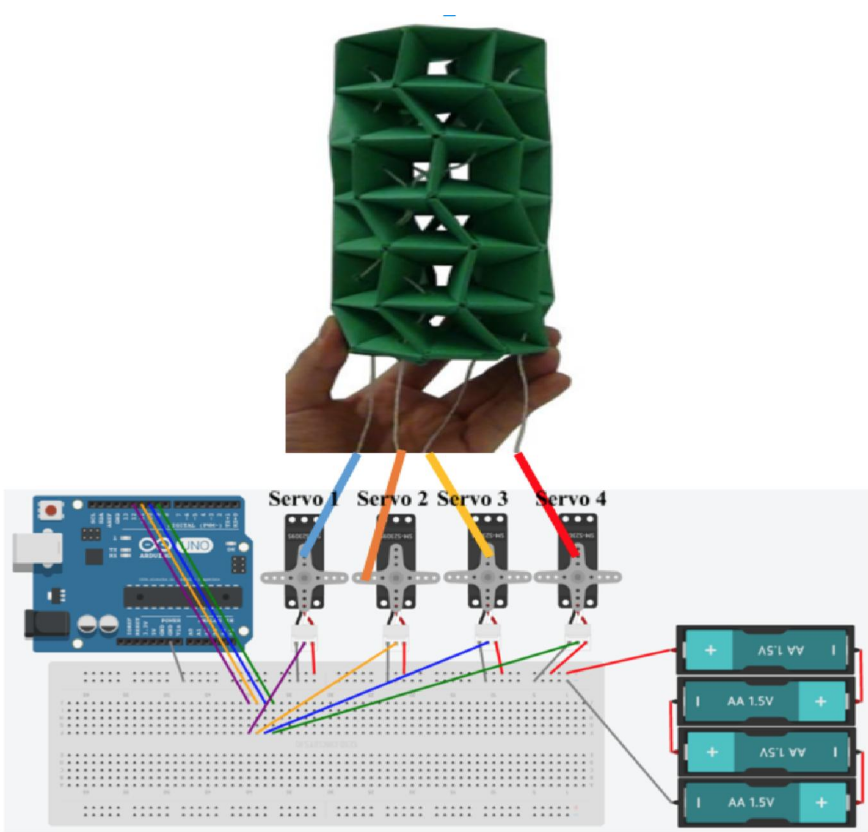
(a) (b) (c)

<Origami twisted tower actuation by thread>

(a): Actuation made by thread pulling
(b): Contraction by torsional movement
(c): Contraction by alternative torsional movement

3. Integration with motor to operate origami tower

By connecting the threads to servo motors, the origami tower can be controlled like a human. Immediately below is the set-up diagram of the origami tower actualization and the threads connecting the origami twisted tower and the servo motors. Following the previous diagram is Arduino code that can be used to control the origami tower by controlling the servo motors.



<Set-up of Origami Tower Actualization>

```
 #include <Servo.h> //include the servo library

// declaration of initial position and pin number, etc...
int pull=135; //servo motor pulling command
int release=45; ////servo motor releasing command
int servoPin1 = 9; //declare pin for servo number 1
int servoPin2 = 10; //declare pin for servo number 2
int servoPin3 = 11; //declare pin for servo number 3
int servoPin4 = 12; //declare pin for servo number 4
int servoDelay =15; //delay to allow the servo to reach position;


// creating servo object from servo.h
Servo myservo1; // create a servo object for servo number 1
Servo myservo2; // create a servo object for servo number 2
Servo myservo3; // create a servo object for servo number 3
Servo myservo4; // create a servo object for servo number 4


//initial setup code
void setup() {
  Serial.begin(9600); //start serial port
  myservo1.attach(servoPin1); //declaring the pin number for servo 1
  myservo2.attach(servoPin2); //declaring the pin number for servo 2
  myservo3.attach(servoPin3); //declaring the pin number for servo 3
  myservo4.attach(servoPin4); //declaring the pin number for servo 4
}
```

```
//loop code which runs over and over
void loop() {
  //input from the user is given by the serial monitor w,a,s,d
  Serial.println("a: left, d: right, s: contraction, w: extension  ");
  if(Serial.available()>0)
  {
    char var=Serial.read(); //input read from the serial monitor

    // switch and case formula for servo movement about each inpuut
    switch(var)
    {
      case 'w':
      Serial.println("Extension!!");
      myservo1.write(pull);
      myservo2.write(pull);
      myservo3.write(release);
      myservo4.write(release);
      break;
      case 'a':
      Serial.println("left!!");
      myservo1.write(pull);
      myservo2.write(release);
      myservo3.write(pull);
      myservo4.write(release);
      break;
```

```
      case 's':
      Serial.println("contraction!!");
      myservo1.write(pull);
      myservo2.write(pull);
      myservo3.write(pull);
      myservo4.write(release);
      break;
      case 'd':
      Serial.println("right!!");
      myservo1.write(release);
      myservo2.write(pull);
      myservo3.write(pull);
      myservo4.write(pull);
      break;
      default:
      Serial.println("input signal wrong!!");
      break;

    }

  }
  delay(servoDelay); //delay for the servo to reach the position
}
```

<Arduino Code for Origami Tower Actualization>

4. Results

Each servo motor can either pull or release its thread, and specific combinations of pulling or releasing can bend, contract, or extend the tower. The origami tower is controlled by user input. The tower is bent when the "a" and "d" keys are pressed, contracted when the "s" key is pressed, and extended when the "w" key is pressed. By connecting the origami twisted tower to servo motors, it becomes to automate and control the movements of the tower.

5. Conclusion

Although the origami-folding mechanism is highly labor intensive, recent research has made it possible to automate simple valley and mountain folds, indicating potential that more complex folding mechanisms can be automated in the future. Since origami has the dual property of flexibility and rigidity, origami automation is applicable to many fields, not only robotics and structural design. From this project, a flexible robot arm was modeled as a special origami structure named "origami twisted tower." This structure takes advantage of the inherent properties of origami and controls the tower movement from motor actuation by thread connection between the tower and the motors.

Reference:

1. Deformable wheel robot based on origami structure, DY Lee, 2013 ICRA.
2. Origami Bot-I: A thread-actuated origami robot for manipulation and locomotion, E Vander Hoff, 2014 ICRA.
3. Design and analysis of an origami-based three-finger manipulator, D Jeong, Robotica 2018.
4. Three-Dimensional Printable Origami Twisted Tower: Design, Fabrication, and Robot Embodiment, Tao Liu, IEEE Robotics and Automation Letters 2018.
5. 메카솔루션 블로그: https://goo.gl/7R9eHN

11/17/2018