

# Solid Modelling Interaction with Sensors for Virtual Reality Welding

Bharath V. G.<sup>1\*</sup> and Rajashekar Patil<sup>2</sup>

<sup>1</sup> School of Mechanical Engineering, REVA University, Bengaluru, Karnataka, India

<sup>2</sup> Department of Mechanical Engineering, Atria Institute of Technology, Bengaluru, Karnataka, India

**Abstract.** The imperative part of strong displaying in virtual reality framework is to improve the movement of a user with definitive movement control in genuine intelligent condition. In a real environment motion characteristic is based on sensors and servomechanisms where as in virtual reality systems the motion fundamentals are kinematic in nature. To achieve physically correct interactivity suitable dynamic constraints, should be imposed which can be obtained by augmented reality interface. Beyond input and output hardware, the underlying software plays a very important role in virtual reality systems. It is responsible for the managing of input/output devices, analysing incoming data and generating proper feedback. This research paper focusses on movement displaying for connection between CAD models and Virtual Reality Models using 3ds Max 2017 and Unity 3d softwares. The paper portrays different equipment arrangement of sensors and Arduino for virtual reality welding.

**Keywords:** Welding, Virtual Reality, 3ds Max 2017, Unity 3d, Solid modelling, Arduino

## 1 Introduction

### 1.1 Virtual Reality

Virtual Manufacturing technique is a virtual reality technology to create virtual environment on the computer screen to simulate the physical world. The expertise and knowledge base gained from the work in the virtual environment enables the user to apply more meaningfully in the real-life situation. Banerjee and Tesic [1] has worked in Graphical User Interface, Virtual Reality advancements, auto translation.

### 1.2 Literature Survey

Balyiss [2] emphasized on VR methods with better 3D perception of a question and clarified hypothetical strong displaying strategies utilizing Virtual Reality Modelling Language (VRML) and 3ds Max. Bharath and Rajashekar Patil [3] created diverse parts of virtual welding stage and enlivened the stream of welding utilizing VR instruments. Leo

---

\* Corresponding author: [bharathvg@gmail.com](mailto:bharathvg@gmail.com)

Louis [4] considered the working standard of Arduino, its equipment/programming elements and its favourable circumstances of building new gadgets of our own to make and execute creative items to worldwide market. Mendes et al. [5] clarified that the Arduino Mega 2560 is utilized for different sorts of utilization, for example, laser cutting, CNC and 3D Printing. Deepak et al. [6] has emphasized the significance of Arduino and its intelligence amongst machine and programming in 3D Printer.

## 2 Methodology

Motion modelling of virtual reality welding involves parameters such as modelling, simulation and integrity between virtual model and realistic model.

### 2.1 Methodology of modelling and simulation

3D modelling of virtual welding using 3DS Max software illustrates modelling of an entire virtual welding platform, simulation of a welding in virtual mode, algorithm.

### 2.2 Methodology of motion modelling

Unity 3d software is a game engine which illustrates the simulation of an entire model in a game mode and integrates model between virtual and real environment. Arduino is a microcontroller which integrates several sensors to a game engine by serial communication coding. Fig. 1 illustrates the methodology of Virtual Reality Welding.

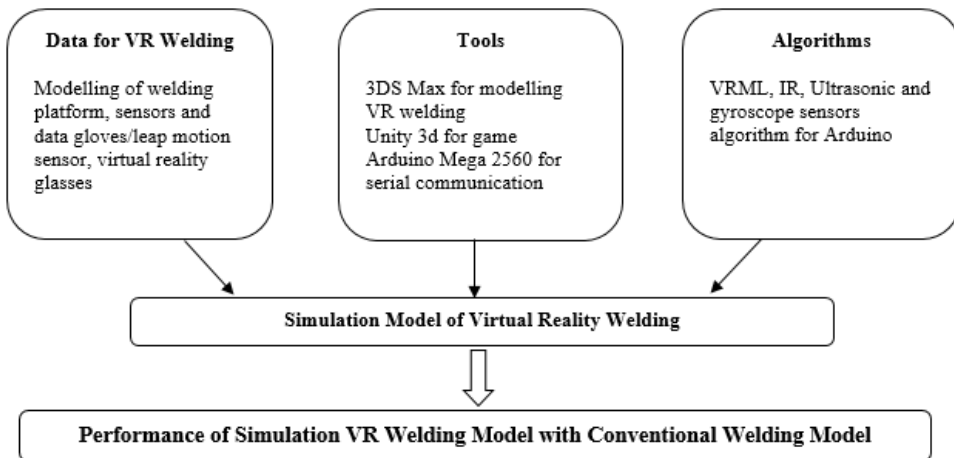


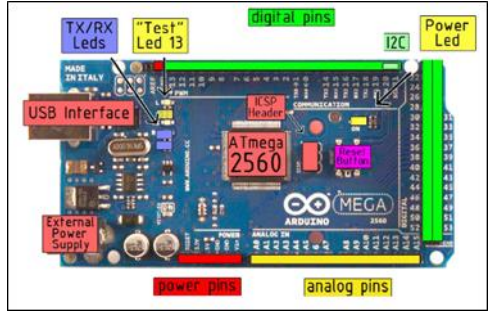
Fig. 1. Methodology of Virtual Reality Welding

## 3 Design

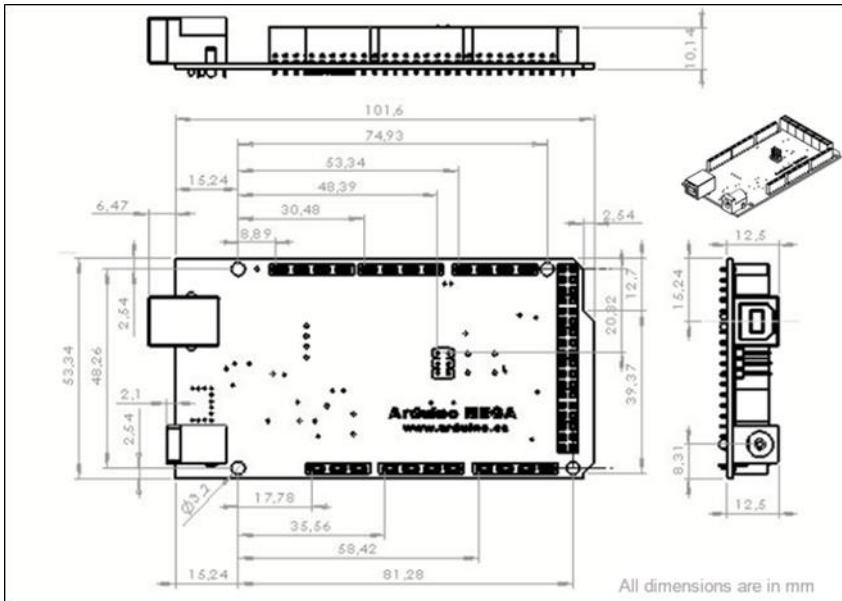
### 3.1 Arduino Mega 2560

The Arduino Mega 2560 is a microcontroller which has 54 advanced info/yield pins, 16 simple data sources, 4 Universal Asynchronous Receiver-Transmitter (UART) equipment serial ports for Transistor Logic (TTL) (5V) serial correspondence, a 16 MHz precious stone oscillator, a Universal Serial Bus (USB) association, a power jack, an In-Circuit

Serial Programming (ICSP) header, and a reset catch. It essentially associates with a PC with a USB link or power it with an AC-to-DC connector or battery to begin. The Mega is perfect with most shields intended for the Arduino. Fig. 2a demonstrates the arrangement of Arduino Mega 2560 with key marking and Fig. 2b gives a nitty gritty specialized drawing of Arduino Mega 2560 composed utilizing Solidedge ST9 programming.



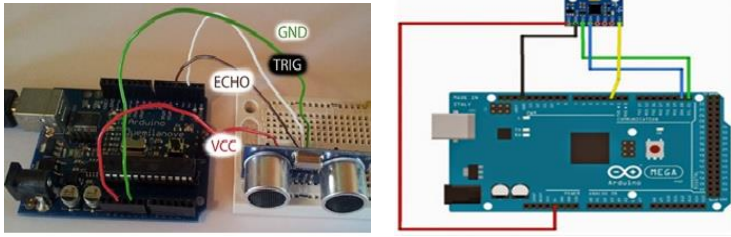
**Fig. 2a.** Top view of Arduino Mega 2560



**Fig. 2b.** Technical Drawing of Arduino Mega 2560

### 3.2 Sensors and its coding for interactivity

Ultrasonic sensor transmits an ultrasound at 40,000 Hz which goes through the air and if there is a protest or impediment on its way, it will skip back to the module [7, 8]. Computation of the separation is acquired by considering the travel time and the speed of the sound. Ultrasonic sensor is utilized as a part of virtual welding to gauge the separation of a terminal development from one end to flip side of the workpiece [9, 10]. Fig. 3 demonstrates ultrasonic sensor and MPU 6050. The MPU-6050 gadgets consolidate a 3-hub spinner and a 3-hub accelerometer on a similar silicon bite the dust, together with an installed Digital Motion Processor (DMP), which forms complex 6-hub Motion Fusion calculations. These Motion Tracking gadgets are intended for measuring edge development of a terminal. Table 1 demonstrates the coding of Ultrasonic sensor and MPU 6050 sensor.



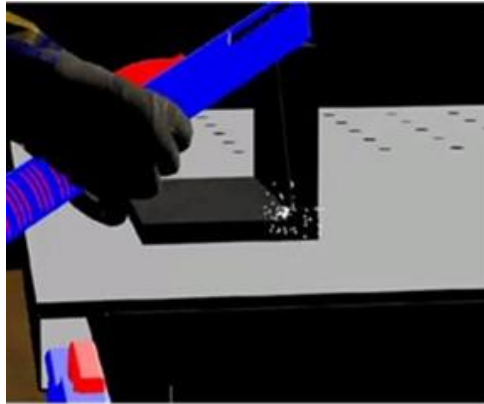
**Fig. 3.** Ultrasonic sensor and MPU 6050 Sensor

**Table 1.** Ultrasonic and MPU 6050 sensor coding.

<pre> const int trigPin = 2; const int echoPin = 4; void setup() {   // initialize serial communication:   Serial.begin(9600); } void loop() {   long duration, inches, cm;   pinMode(trigPin, OUTPUT);   digitalWrite(trigPin, LOW);   delayMicroseconds(2);   digitalWrite(trigPin, HIGH);   delayMicroseconds(10);   digitalWrite(trigPin, LOW);   pinMode(echoPin, INPUT);   duration = pulseIn(echoPin, HIGH);   // convert the time into a distance   inches = microsecondsToInches(duration);   cm = microsecondsToCentimeters(duration);   Serial.print(inches);   Serial.print("in, ");   Serial.print(cm);   Serial.print("cm");   Serial.println();   delay(1); } </pre>	<pre> #include&lt;Wire.h&gt; const int MPU_addr=0x68; int16_t AcX,AcY,AcZ,Tmp,GyX,GyY,GyZ; int minVal=265; int maxVal=402; double x; double y; double z; void setup(){   Wire.begin();   Wire.beginTransmission(MPU_addr);   Wire.write(0x6B);   Wire.write(0);   Wire.endTransmission(true);   Serial.begin(9600); } void loop(){   Wire.beginTransmission(MPU_addr);   Wire.write(0x3B);   Wire.endTransmission(false);   Wire.requestFrom(MPU_addr,14,true);   AcX=Wire.read()&lt;&lt;8 Wire.read();   AcY=Wire.read()&lt;&lt;8 Wire.read();   AcZ=Wire.read()&lt;&lt;8 Wire.read();   int xAng = map(AcX,minVal,maxVal,-90,90);   int yAng = map(AcY,minVal,maxVal,-90,90);   int zAng = map(AcZ,minVal,maxVal,-90,90);   x= RAD_TO_DEG * (atan2(-yAng, -zAng)+PI);   y= RAD_TO_DEG * (atan2(-xAng, -zAng)+PI);   z= RAD_TO_DEG * (atan2(-yAng, -xAng)+PI);   Serial.print("AngleZ= ");   Serial.println(z);   Serial.println("-----");   delay(400); } </pre>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## 4 Virtual Welding Simulation Model

A simulation model of virtual welding is planned utilizing 3ds Max programming and a VRML content is produced for intuitiveness amongst sensors and simulation model. Unity 3d is a game engine used to impart between simulation display and practical condition in which representation is made utilizing virtual reality glasses and information gloves. Fig. 4 shows simulation model of virtual welding arc is generated by electrode contact.



**Fig. 4.** Ultrasonic sensor and MPU 6050 Sensor

## 5 Conclusions

The present innovation is looking in front of virtual reality. As there is a shortage of materials step by step, what's to come will wind up noticeably virtual to expertise the works in welding. A virtual reality framework has been created to accomplish craft of welding in virtual environment with a prudent movement control gadgets of welding stage plan and resulting representation of circular segment welding. Different elements like camera scenes, VRML contents, serial correspondence coding are bound together to user for less demanding comprehension. In this present research work strong displaying of virtual welding is produced and created Arduino Mega code for ultrasonic and MPU sensors to communicate amongst virtual and genuine conditions. In future research scope haptic sensors which are costly, for example, Human Machine Interface and Programmable Logical Controller are utilized for better intelligence.

## Acknowledgements

The authors thank Mr. Prashanth S. N. and Nuthan, student of Atria Institute of Technology for their support on analyzing and suggesting inputs in Arduino Mega 2650.

## References

1. P. Banerjee and R. Tesic, Design of Virtual Objects for Exact Collision Detection in Virtual Reality Modeling of Manufacturing Processes, Proceeding of International Conference on Robotics and Automation, 18, 367-376 (1999).
2. G. M. Balyiss, A. Bowlyer, R. I. Talyor. and P. G. Willis, Theoretic Solid Modeling Techniques and Application Using the Virtual Manufacturing, Proceeding of CSG, 94 (1994)
3. V. G. Bharath and Rajashekar Patil, Designing Interactive 3D Modelling of Virtual Welding System from CAD to VRML, J of Control Theory and Applications, 10, 145-153 (2017)
4. Leo Louis, Working Principle of Arduino and using it as a Tool for Study and Research, J of Control Automation Communication and Systems, 1, 21-29, (2016)

5. Allen R. Mendes, Atur M. Mehta, Bhavya H. Gohil, Implementation of the Automatic and Interactive Chess Board, *J of Electrical and Electronics Engineering*, 9 01- 04 (2014)
6. Deepak D, Mohammed Murtaza, Prashanth S N, Harsha N, Rajashekar Patil and Bharath V G Economical 3d – Printer by Adopting FDM Technique, *J of Mechanical Engineering and Technology*, 8, 442- 447 (2017)
7. Manpreet Kaur, Distance Measurement of Object by Ultrasonic Sensor HC-SR04, *J for Scientific Research & Development*, 3, 503-505 (2015)
8. K. Shrivastava, Distance Measurement of an Object or Obstacle by Ultrasound Sensors using P89C51RD2, *J of Computer Theory and Engineering*, 2, 64-68 (2010)
9. Yi-Jen Mon, “The Gyroscope Sensor Test by Using Arduino Platform”, *J of Scientific & Technology Research*, 4, 398-400 (2015)
10. Kyujung Lee and Choong Ho Lee, Effect of Gyro Sensor to the Position Error in Mobile Robot, *J of Software Engineering and Its Applications*, 9, 41-48 (2015)