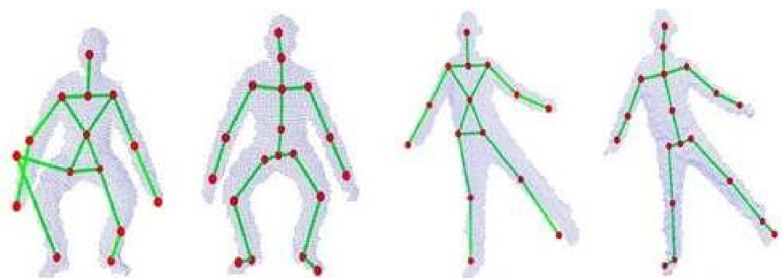
Group 16 - Pose Capture

18-748 Wireless Sensor Networks, Spring 2018

Website link : https://rinipatel.github.io/PoseCapture/



Christopher Hung (christoh@andrew.cmu.edu)

Rini Mukund Patel (rinip@andrew.cmu.edu)

Tushar Goyal (tgoyal1@andrew.cmu.edu)

Project Mentor : Alex He (mengwenh@andrew.cmu.edu)

1. PROJECT SUMMARY

We implemented a wireless system that captures a person's movements as they move their limbs. The system uses a network of nodes each with an IMU to relay the orientation measurements for the respective limbs to a master on a PC, which processes the data and represents the estimated pose in form of graphical 3D model.

2. PROJECT MOTIVATION

The recording of human position and pose has many application including film industry, gaming, virtual reality, robotics, medical training such as physical therapy etc. Many of these techniques are based on either computer vision based approach or wired sensor frame based approach [1].

However, such techniques are not very ideal for human pose capturing under all circumstances.. In case of computer vision based techniques, they call upon the strong requirement for a person to be visible in front of capture device. For example, a robot mimicking the actions of a human operator, the robot would either need to face the operator or have to be fed with the video feed of the operator via a camera present on the setup. Also, the illumination conditions and obstacles prevent the accuracy of these methods.

Similarly, there are sensor frame based systems which are constrained in form of portability and flexibility of movements. The frame skeleton would need to be lightweight strong material or else the weight of frame itself can exceed the user's weight and is deemed to be unusable under certain medical conditions. A good frame with high flexibility of movements and lightweight material like carbon fiber can increase the cost the application in which they are being used. Lastly, these frames have to be custom made for various body physique and age groups, which itself is a very arduous task.

One another interesting methodology to capture the pose in literature is using acoustic methods. It uses the markers emitting signals, attached to the body and set of receivers to receive the signals and try to estimate the pose. The technique suffers from the interference from the environment and the same problem as vision based methods, where the person needs to be in line of sight, and lack a way for receivers to be portable. [1], [2]

This report describes a wireless approach with IMU sensors where the sensors could be attached to the body and the receiver software running on the host would use the inertial data to estimate the pose of the subject. The term inertial sensor comes from the group of sensors consisting of 3-axis accelerometer and gyroscope coupled with a magnetometer. The methodology doesn't suffer from portability and cost concerns since the sensors are relatively cheap and can be reused for multiple physique and age groups and have no constraints of weight and frame design for body movement etc. Additionally, unlike the cases of optical capture techniques where the video feed should be directly pointing at the human subject, the only constraint on the methodology would be the wireless network range to which sensors can send the data to the master where the pose can be estimated.

3. PROJECT GOALS

The main goal of the project is to get a full body capture of the person using a) wireless sensor network, which is b) robust under variety of environmental conditions, and c) use the orientation data from IMUS to post process in the form of 3D body model. The master uses WiFi for communication with nodes via user datagram protocol (UDP). Nodes send the absolute orientation information with respect to world coordinate system and hence, have robustness to intermittent packet losses. The main focus is on developing system with low latency and getting more real-time estimation of movements.

The system exhibits natural extension for variety of use cases such as animation character, virtual reality etc. Here, the post-processing stage would differ according to the application being used. For example, in case of film making, the pose capture might be used for VFX and animation.

4. KEY USE CASES

The most definitive use of our project is in computer animation. Rather than manually moving the limbs of a character, the animator can build a model of the character, act out the desired motion physically, and apply the recorded motions directly to the limbs of the model. This animation could be used in film making, game development etc.

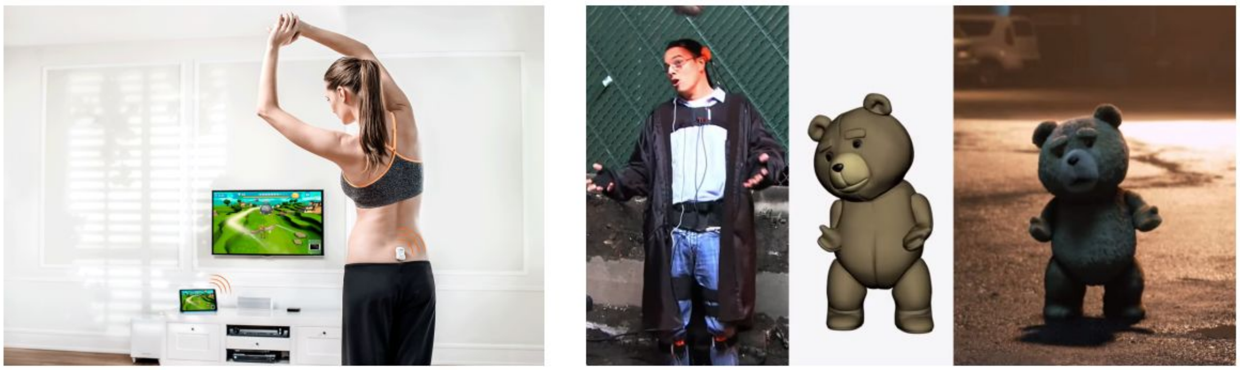


Figure 1: Examples illustrating the use of multiple IMUs placed on the human body to estimate its pose. a) Back pain therapy using serious gaming. IMUs are placed on the chest-bone and on the pelvis to estimate the movement of the upper body and pelvis. b) Actor Seth MacFarlane wearing 17 IMUs to capture his motion and animate the bear Ted [2].

Another use case is in virtual reality. With an accurate and low-latency solution, pose capturing can add another degree of immersion into the virtual world, where the user's motions are mirrored perfectly by the user's avatar. In case of robotics, the pose might be used to mimic the human behavior and hence, the pose estimated would be used to move the robot into a pose. This can be used to cases like where strong hydraulic robots could be used to move heavy weight objects around and human operator can just use the behavior to control the robot.

The most compelling use case for our methodology would be the cases when we require mobility, flexibility and line of sight requirement is not necessary. For example, consider the case when a pose capture application is required in medical scenarios where various age groups and physique need to be considered, the wireless IMU technique has significant advantageous over frames.

5. SYSTEM DESIGN

To implement our solution, we built a network of several sensor nodes that communicate to a master. The data collected from these nodes allows the master to get the orientation of each node with respect to global frame. With the nodes situated on key parts of the user's body, the orientation and position of each node will give enough information for the master to estimate a pose.

Key hardware components for our project:

- IMU Sensors (SparkFun 9DoF Razor IMU M0)
- Batteries for IMUs (Li Ion Batteries (3.7V 400mAh))
- WiFi shields (ESP8266 WiFi modules)

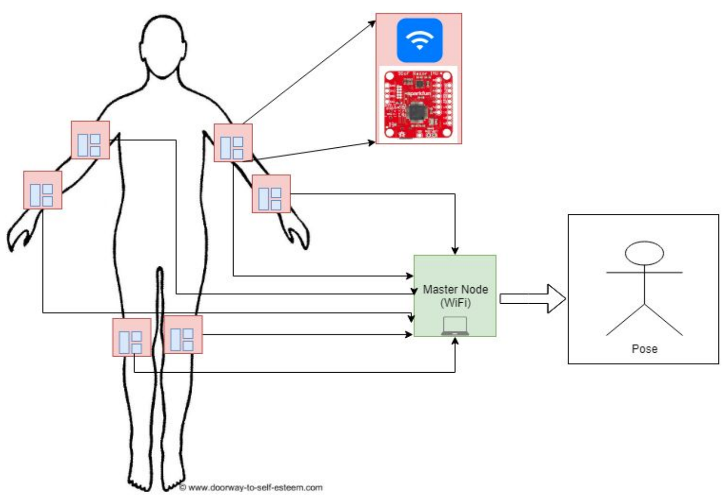


Figure 2:. IMUs attached to body are used to estimate pose by aggregation of sensor samples at master node.

Each IMU sensor node collects raw sensor data, and and performs sensor fusion to obtain orientation euler angles (yaw, pitch and roll) using Direct Cosine Matrix (DCM) based Altitude Heading Reference System (AHRS) with gyro drift correction using accelerometer and magnetometer vectors by Madgwick et al [3]. The algorithm also reduces the noise in orientation and numerical errors.

The orientation measurements are sent using ESP WiFi modules to the master node PC. Master node having received data from all sensor nodes, offsets it with the initial calibrated values of pose and estimates the new pose. Each node has a fixed sampling time (100Hz) for getting the new data, and master periodically updates the visual representation of pose in form of 3D human model.

6. SUBSYSTEM INTERFACE

Hardware: The IMU board communicated with the WiFi shield using UART (universal asynchronous receiver-transmitter). The power (3.3v), ground, Tx, and Rx pins on the IMU were connected to the power, ground, Rx, and Tx pins on the WiFi shield in that order. The enable pin on the WiFi shield was connected to a separate 3.3v output on the IMU to have the WiFi constantly sending.

The WiFi chip was programmed to connect to a set access point broadcast from the PC, read data off the UART connection with the IMU until a stop character, and send the data over wifi to the PC using UDP.

Communication: The communication with PC happens via ad-hoc WiFi network configured statically to sensor nodes. The packets (70 bytes max) contain a unique sensor ID for the corresponding source followed by yaw, pitch and roll values as application layer payload. The packets are sent to PC via UDP layer protocol. All the sensor data is always sent to master directly via one hop communication. The PC aggregates the data into different streams with respect to the sensor ID values and applies the transformations on part or limb of 3D model with data corresponding to ID attached to limb. We can support ~100 (54.4Mbps/70B) nodes at a time.

Body model: We are using a hip rooted body model where all the movements of bones are rooted on a tree as shown in Figure 3. This defines the relative position of bones and helps in modelling body motion and movements correctly. For example, upper arm is parent of lower arm thus a movement in upper arm would cause motion in lower arm as well. The same model is being used by Blender and Unity.

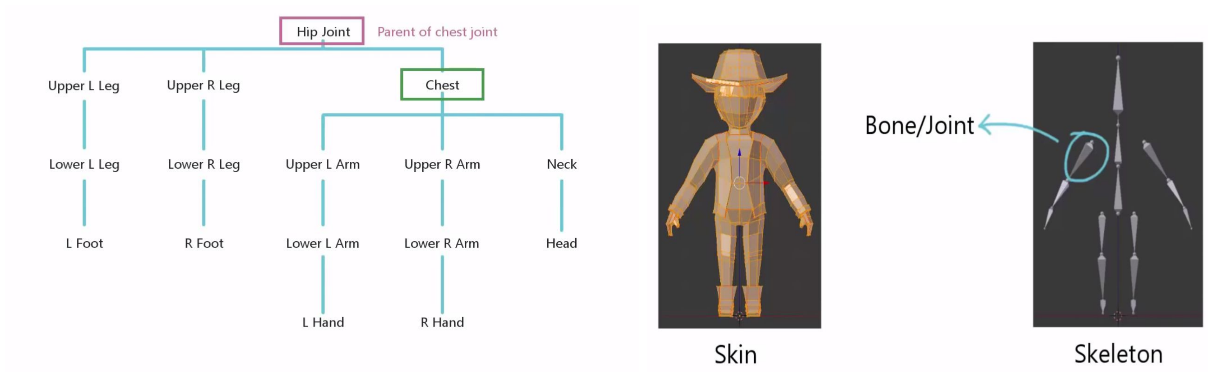


Figure 3: Hierarchical Body model and animation model

Animation:

Blender: The Blender visualization is powered by a series of controllers called once every program tick. One controller reads all the WiFi packets in the buffer and stores the latest one from each of the nodes. Another controller updates the position of the model using the 3 yaw, pitch, and roll angles. In the IMU's reference frame for the orientation it was attached to the body, yaw was a rotation around the Y axis, pitch was a rotation around the X axis, and roll was a rotation around the Z axis, applied in that order. We created a single rotation matrix by generating three rotation matrices with the specified angles about the corresponding axes and multiplying them in order. Because each of the IMUs had its own reference frame upon startup, we zeroed the rotations of each sensor by having the user get into a specified pose and recording the rotation of each sensor, and taking the composition of the inverse of that rotation with the current rotation to get the resulting total rotation.

Unity: Controllers are written in Visual Studio C#. One software thread running in background to continuously reads from WiFi UDP port of host PC, and event-driven controllers for each sensor node update the graphical movement of associated limb on screen at the rate of 10fps. The orientation data read from sensors are yaw, pitch, and roll values in the world coordinates. In order to translate the movement of the limb to the desired coordinate on screen, we first translate the sensor YPR values to respective XYZ euler angles as per the orientation of limb on screen and then perform two matrix multiplications. First matrix multiplication is with the inverse of rotational matrix of the original position for that node with the current position to get the current world position, and second multiplication with the result of first to the original position of the limb on screen in order to translate the movement of the limb to its counterpart on the screen.

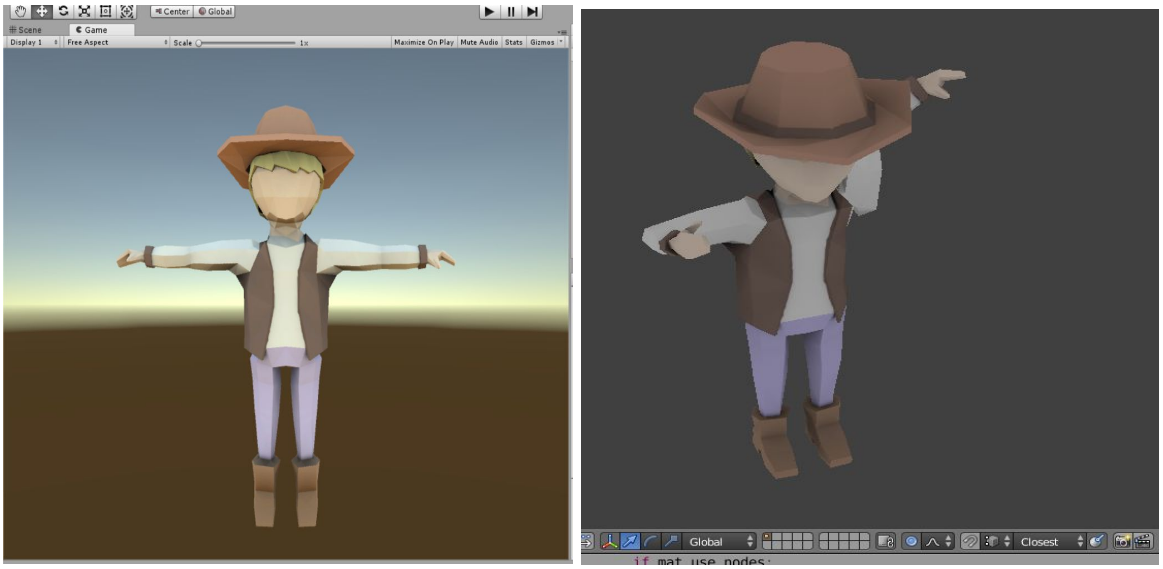


Figure 4: Illustration of final demo sequences on a) Unity 3D and b) Blender

7. LESSONS LEARNED

We started off with goal to build nodes with Arduino as processing core along with IMU and WiFi chip as sensing and communication peripherals. However, we learnt that the IMU being used i.e. SparkFun 9DoF Razor IMU M0 (SEN 14001) has a on chip SAMD21 microprocessor which can be used to interface to ESP8266 WiFi chip over serial and hence, eliminating the requirement for Arduino Uno boards. This helped us in reducing the hardware and software complexity for the nodes. This also helped us in reducing per sensor node cost by $22.

Another learning comes with respect to algorithms and implementation perspective for 3D model. Our estimation for a simplistic 3D model required us to learn some nitty gritties of graphics and animation development such as rigging, skeletal motion, world vs local coordinates and many more. This potentially opened us to a domain previously untouched by all of us and gave a learning experience rewarding in future with different aspects if they need be.

8. PROJECT STATUS AND SCHEDULE

Week of	Tasks done
March 20	Establish Wireless communication. Fabricate Uno with ESP8266 and Razor IMU as peripheral with ability to read raw sensor reading. We were able to read over WiFI from IMU without Arduino via laptop Ad-hoc WiFi. We started trying out to find out correct orientation using raw sensor values.
April 5	Stick Figure Pose Capture Come up with a way to create some stick figure based 3D plot in a crude way to show progress for intermediate demo. Define the interface between sensor and plotting GUI to integrate system. Figured out that magnetometer sensor needs to calibrated properly. Used the actual orientation values to connect the limbs one by one.
April 25	3D Figure Pose Capture Create 3D model using hip rooted movement model for an animation character. Modeled the animation using global frame to local frame translation on the limbs of various parts of the skeleton. Used actuators and component scripts in Blender and Unity respectively to convert the orientation values into respective XYZ rotation so as to show real 3D model with all the data.
May 2	Extra Credit Option - Add leg movement to body capture We successfully modeled the leg movements in Blender using IMU and were able to demonstrate leg movements.

9. WORK PARTITIONING

Our work methodology consisted of regular weekly meetings with trying to implement stuff together in pair-programming based environment. Due to the nature of the project, the work required us to sit together and collaborate on tasks such as hardware setup, data communication, calibration and working with 3D models. But in a nutshell, we partitioned the work on the larger granularity as listed below.

Chris: Hardware bring-up, Sensor calibration, Stick figure model, Python script in Blender, WiFi Comm

Rini: Sensor calibration, Body model, Unity C# code, AHRS algorithm, Demo person

Tushar: Sensor calibration, AHRS algorithm, Unity C# code, Python script in Blender, Website

10. CONCLUSION AND FUTURE WORK

To conclude this work, we have successfully

1. Built a wireless system which can estimate the pose and is affordable.
2. Built a network which is robust under environmental conditions and node failures.
3. Built an animated 3D body model that updates the user's pose in real-time.

But there are still many remaining possibilities to extend this work further, some of them are listed below:

Full Body Model: Currently we use 6 sensor nodes to capture hands and upper legs movements. Given more nodes, we can incorporate full body pose capture with lower legs, head and chest movements, as our body model framework is already designed keeping in mind this scalability option. This model will have to be foot rooted in order for torso to move and it will be capable of turning in 3D space and bending down..

Motion Tracking: A natural extension of pose capture is to have motion tracking ability. With integrating accelerometer sensor data over the time, capturing the motion of person is possible but difficult task as errors tend to accumulate over the time with integration. Sophisticated components like Kalman Filter for noise modeling can be used and motion tracking can be achieved but this for now is out of scope for this project.

Facial Expression: A cool feature with body capture would be having a way to capture the facial expressions of a person. As another extension to this project, computer vision based methodology using low cost USB Camera can be used for this cause.

11. REFERENCES

[1] Y. Zheng, K.-C. Chan, and C. C. L. Wang, "Pedalvatar: An IMU-based real-time body motion capture system using foot rooted kinematic model".

[2] M. Kok, J. D. Hol, and T. B. Schön, "Using Inertial Sensors for Position and Orientation Estimation".

[3] S. O. H. Madgwick, at el, "Estimation of IMU and MARG orientation using a gradient descent algorithm".