6 REFERENCES

[1] Who.int, 'Ageing and health', 2018. [Online]. Available: http://www.who.int/mediacentre/factsheets/fs404/en/ [Accessed: 23- Jun- 2018].

[2] Duque Garcia, Ismael “Adapting Robot Behavior in Smart Homes: a Different Approach Using Personas” IEEE RO MAN 2013.

[3] A. Dohr, R. Modre-Opsrian, M. Drobics, D. Hayn, and G. Schreier, "The internet of things for ambient assisted living," in Information Technology: New Generations (ITNG), 2010 Seventh International Conference on, 2010, pp. 804-809.

[4] M. Arndt, S. Wille, L. de Souza, V. F. Rey, N. Wehn, and K. Berns, "Performance evaluation of ambient services by combining robotic frameworks and a smart environment platform," *Robotics and Autonomous Systems,* 2013.

[5] C. Reinisch, M. J. Kofler, and W. Kastner, "ThinkHome: A smart home as digital ecosystem," in *Digital Ecosystems and Technologies (DEST), 2010 4th IEEE International Conference on*, 2010, pp. 256-261.

[6] F. K. Aldrich, "Smart homes: past, present and future," *Inside the smart home,* vol. 1, pp. 17-39, 2003.

[7] S. Schmid, M. Sifalakis, and D. Hutchison, "Towards autonomic networks," in *Autonomic Networking*, ed: Springer, 2006, pp. 1-11.

[8] W. Feng, "Remote service provision for connected homes," 2010.

[9] X. Zhu, Y. Yu, Y. Ou, D. Luo, C. Zhang, and J. Chen, "System Modeling of a Smart-Home Healthy Lifestyle Assistant," in *Agents and Data Mining Interaction*, ed: Springer, 2013, pp. 65-78.

[10] H. Lee and J. Kwon, "The Smart Home Service System Architecture for Healthy and Safe Human Life," 2013.

[11] S. Brownsell, D. Bradley, S. Blackburn, F. Cardinaux, and M. S. Hawley, "A systematic review of lifestyle monitoring technologies," *Journal of telemedicine and telecare,* vol. 17, pp. 185-189, 2011.

[12] D. J. Cook, "How smart is your home," *Science,* vol. 335, pp. 1579-1581, 2012.

[13] C. Ramos, J. C. Augusto, and D. Shapiro, "Ambient intelligence—The next step for artificial intelligence," *Intelligent Systems, IEEE,* vol. 23, pp. 15-18, 2008.

[14] D. J. Cook, J. C. Augusto, and V. R. Jakkula, "Ambient intelligence: Technologies, applications, and opportunities," *Pervasive and Mobile Computing,* vol. 5, pp. 277-298, 2009.

[15] H. Lee and J. Kwon, "The Smart Home Service System Architecture for Healthy and Safe Human Life," 2013.

[16] F. Viani, F. Robol, A. Polo, P. Rocca, G. Oliveri, and A. Massa, "Wireless Architectures for Heterogeneous Sensing in Smart Home Applications: Concepts and Real Implementation," 2013.

[17] I. Khan, A. Mahmood, N. Javaid, S. Razzaq, R. Khan, and M. Ilahi, "Home Energy Management Systems in Future Smart Grids," *arXiv preprint arXiv:1306.1137,* 2013.

[18] J. Paradiso, P. Dutta, H. Gellersen, and E. Schooler, "Guest Editors' Introduction: Smart Energy Systems," *Pervasive Computing, IEEE,* vol. 10, pp. 11-12, 2011.

[19] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati*, et al.*, "Smart Grid and Smart Homes: Key Players and Pilot Projects," *Industrial Electronics Magazine, IEEE,* vol. 6, pp. 18-34, 2012.

[20] S. Ahmad, "Smart metering and home automation solutions for the next decade," in *Emerging Trends in Networks and Computer Communications (ETNCC), 2011 International Conference on*, 2011, pp. 200-204.

[21] D. Niyato, L. Xiao, and P. Wang, "Machine-to-machine communications for home energy management system in smart grid," *Communications Magazine, IEEE,* vol. 49, pp. 53-59, 2011.

[22] T. Buracchio, H. H. Dodge, D. Howieson, D. Wasserman, and J. Kaye, "The trajectory of gait speed preceding mild cognitive impairment," *Archives of neurology,* vol. 67, p. 980, 2010.

[23] M. Hogan, "Technophobia amongst older adults in Ireland," 2006.

[24] S. Brownsell and D. Bradley, *Assistive technology and telecare: forging solutions for independent living*: The Policy Press, 2003.

[25] G. Williams, K. Doughty, and D. A. Bradley, "A systems approach to achieving CarerNet-an integrated and intelligent telecare system," *Information Technology in Biomedicine, IEEE Transactions on,* vol. 2, pp. 1-9, 1998.

[26] B. Brumitt, B. Meyers, J. Krumm, A. Kern, and S. Shafer, "Easyliving: Technologies for intelligent environments," in Handheld and ubiquitous computing, 2000, pp. 12-29.

[27] A. Kuutti, A. Dvoryanchikova, A. Lobov, J. L. M. Lastra, and T. Vantera, "A device configuration management tool for context-aware system," in Industrial Informatics (INDIN), 2012 10th IEEE International Conference on, 2012, pp. 10-15.

[28] G. Bin, "An Ontology-based Programming Platform for smart artefact system. ," Unpublished MA thesis vol. Keio University 2009.

[29] T. Perumal, A. R. Ramli, C. Y. Leong, S. Mansor, and K. Samsudin, "Interoperability for smart home environment using web services," International Journal of Smart Home, vol. 2, pp. 1-16, 2008

[30] C. Fabbricatore, H. Boley, and A. P. Karduck, "Machine learning for resource management in smart environments," in Digital Ecosystems Technologies (DEST), 2012 6th IEEE International Conference on, 2012, pp. 1-6

[31] G. Mantas, D. Lymberopoulos, and N. Komninos, "Security in Smart Home Environment," Wireless Technologies for Ambient Assisted Living and Healthcare: Systems and Applications, p. 170, 2010

[32] S. Brownsell, D. Bradley, F. Cardinaux, and M. Hawley, "Developing a systems and informatics based approach to lifestyle monitoring within eHealth: part I-technology and data management," in Healthcare Informatics, Imaging and Systems Biology (HISB), 2011 First IEEE International Conference on, 2011, pp. 264-271.

[33] M. Gritti, M. Broxvall, and A. Saffiotti, "Reactive self-configuration of an ecology of robots," in Proc of the ICRA-07 Workshop on Network Robot Systems, Rome, Italy, 2007.

[34] G. D. Abowd, A. F. Bobick, I. A. Essa, E. D. Mynatt, and W. A. Rogers, "The aware home: A living laboratory for technologies for successful aging," in Proceedings of the AAAI-02 Workshop “Automation as Caregiver, 2002, pp. 1-7.

[35] G. Weiss, Multiagent systems: a modern approach to distributed artificial intelligence: The MIT press, 1999.

[36] S. A. Shafer, B. Brumitt, and J. Cadiz, "Interaction issues in context-aware intelligent environments," Human-Computer Interaction, vol. 16, pp. 363-378, 2001.

[37] Y. Alsafi and V. Vyatkin, "Ontology-based reconfiguration agent for intelligent mechatronic systems in flexible manufacturing," Robotics and Computer-Integrated Manufacturing, vol. 26, pp. 381-391, 2010.

[38] S. Schmid, M. Sifalakis, and D. Hutchison, "Towards autonomic networks," in Autonomic Networking, ed: Springer, 2006, pp. 1-11.

[39] J. Kim, H.-s. Choi, H. Wang, N. Agoulmine, M. J. Deerv, and J. W.-K. Hong, "POSTECH's U-Health Smart Home for elderly monitoring and support," in World of Wireless Mobile and Multimedia Networks (WoWMoM), 2010 IEEE International Symposium on a, 2010, pp. 1-6.

[40] Y. Al-Safi and V. Vyatkin, "An ontology-based reconfiguration agent for intelligent mechatronic systems," in Holonic and Multi-Agent Systems for Manufacturing, ed: Springer, 2007, pp. 114-126.

[41] J. A. Johnsen, "A Semantic Web-driven Approach to Self-Configurating Computer Systems," Norwegian University of Science and Technology, 2006.

[42] T. Gu, H. K. Pung, and D. Q. Zhang, "A service-oriented middleware for building context-aware services," Journal of Network and computer applications, vol. 28, pp. 1-18, 2005.

[43] G. T. McKee, D. I. Baker, and P. S. Schenker, "Task-directed configuration of networked robotic agents," in Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on, 2002, pp. 2752-2757.

[44] L. E. Parker and F. Tang, "Building multirobot coalitions through automated task solution synthesis," Proceedings of the IEEE, vol. 94, pp. 1289-1305, 2006.

[45] R. Lundh, L. Karlsson, and A. Saffiotti, "Dynamic self-configuration of an ecology of robots," in Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on, 2007, pp. 3403-3409.

[46] P. Clements, R. Kazman, and M. Klein, Evaluating software architectures: Addison-Wesley Reading, 2001.

[47] J. E. López de Vergara, V. A. Villagrá, C. Fadón, J. M. González, J. A. Lozano, and M. Álvarez-Campana, "An autonomic approach to offer services in OSGi-based home gateways," Computer Communications, vol. 31, pp. 3049-3058, 2008.

[48] M. Rodrıguez-Muro and D. Calvanese, "Quest, an OWL 2 QL reasoner for ontology-based data access," OWLED 2012, 2012.

[49] B. Guo, "An Ontology-based Programming Platform for Smart Artifact Systems," 2009.

[50] J. Shen, L. Wang, and Y. Sun, "Configuration of product extension services in servitisation using an ontology-based approach," International Journal of Production Research, vol. 50, pp. 6469-6488, 2012.

[51] G. Gharbi, M. B. Alaya, C. Diop, and E. Exposito, "AODA: an Autonomic and Ontology–Driven Architecture for service–oriented and event–driven systems," International Journal of Collaborative Enterprise, vol. 3, pp. 167-188, 2013.

[52] T. Heider and T. Kirste, "Smart Environments and Self-Organizing Appliance Ensembles," Mobile Computing and Ambient Intelligence, vol. 5181, 2005.

[53] R. Lundh, L. Karlsson, and A. Saffiotti, "Dynamic self-configuration of an ecology of robots," in Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on, 2007, pp. 3403-3409.

[54] L. Karlsson, "Conditional progressive planning under uncertainty," in IJCAI, 2001, pp. 431-438.

[55] E. Garcia, M. A. Jimenez, P. G. De Santos, and M. Armada, "The evolution of robotics research," Robotics & Automation Magazine, IEEE, vol. 14, pp. 90-103, 2007.

[56] P. Lin, K. Abney, and G. Bekey, "Robot ethics: Mapping the issues for a mechanized world," Artificial Intelligence, vol. 175, pp. 942-949, 2011.

[57] G. A. Bekey, Autonomous robots: from biological inspiration to implementation and control: The MIT Press, 2005.

[58] J.-H. Kim, Y.-D. Kim, and K.-H. Lee, "The third generation of robotics: Ubiquitous robot," in Proc of the 2nd Int Conf on Autonomous Robots and Agents, 2004.

[59] G. Bekey and J. Yuh, "The status of robotics," Robotics & Automation Magazine, IEEE, vol. 15, pp. 80-86, 2008.

[60] E. Garcia, M. A. Jimenez, P. G. De Santos, and M. Armada, "The evolution of robotics research," Robotics & Automation Magazine, IEEE, vol. 14, pp. 90-103, 2007.

[61] C. Buiu, F. Cazan, and R. Ciurlea, "Developing of a Service Robot to Recognize and Sort Waste," in Proceedings of the 16th International Conference on Control Systems and Computer Science, 2007, pp. 22-26.

[62] M. Weiser, "Some computer science issues in ubiquitous computing," Communications of the ACM, vol. 36, pp. 75-84, 1993.

[63] M. Weiser, "The computer for the 21st century," Scientific American, vol. 272, pp. 78-89, 1995.

[64] M. Tenorth, U. Klank, D. Pangercic, and M. Beetz, "Web-enabled robots," Robotics & Automation Magazine, IEEE, vol. 18, pp. 58-68, 2011.

[65] X. Chen, J. Xie, J. Ji, and Z. Sui, "Toward Open Knowledge Enabling for Human-Robot Interaction," Journal of Human-Robot Interaction, vol. 1, pp. 100-117, 2012.

[66] M. Inja, N. Heijne, S. Nugteren, and M. de Waard, "Project ai-the darpa robotics challange-footloose," Project Report, Universiteit van Amsterdam, 2013

[67] support.xbox.com, 'Kinect for Windows v2 or an Xbox Kinect sensor', 2014. [Online]. Available: http://www.microsoft.com/en-us/kinectforwindows/.. [Accessed: 23- Jan- 2018].

[68] msdn.microsoft.com,' Kinect for Windows Sensor Components and Specifications', 2014. [Online]. Available: http://msdn.microsoft.com/en-us/library/jj131033.aspx. [Accessed: 18- Feb- 2018].

[69] msdn.microsoft.com,' Getting the Next Frame of Data by Polling or Using Events', 2014. [Online]. Available: https://msdn.microsoft.com/en-us/library/hh973076.aspx. [Accessed: 22- Feb- 2018].

[70] msdn.microsoft.com,' Color Stream', 2014. [Online]. Available: https://msdn.microsoft.com/en-us/library/jj131027.aspx. [Accessed: 25- Feb- 2018].

7 APPENDIX

APPENDIX A - MATHLAB Function for Developing of Artificial Neural Network

```
function [A] = Test9(in1,in2,in3,in4,in5,in6,in7,in8,in9)
A = myNeuralNetworkFunction([in1;in2;in3;in4;in5;in6;in7;in8;in9]);
end


function [Y,Xf,Af] = myNeuralNetworkFunction(X,~,~)
%MYNEURALNETWORKFUNCTION neural network simulation function.
%
% Generated by Neural Network Toolbox function genFunction, 05-Dec-2017 22:24:21.
%
% [Y] = myNeuralNetworkFunction(X,~,~) takes these arguments:
%
%   X = 1xTS cell, 1 inputs over TS timesteps
%   Each X{1,ts} = 9xQ matrix, input #1 at timestep ts.
%
% and returns:
%   Y = 1xTS cell of 1 outputs over TS timesteps.
%   Each Y{1,ts} = 3xQ matrix, output #1 at timestep ts.
%
% where Q is number of samples (or series) and TS is the number of timesteps.


%#ok<*RPMT0>


% ===== NEURAL NETWORK CONSTANTS =====


% Input 1
x1_step1_xoffset = [0;0;0;0;0;0;0;0;0];
x1_step1_gain                                                                    =
[0.00917431192660551;0.0072202166064982;0.00706713780918728;0.00602409638554217;0.006
2111801242236;0.0122699386503067;0.0127388535031847;0.0111731843575419;0.011363636363
6364];
x1_step1_ymin = -1;


% Layer 1
b1 = [-3.1625132944198859;-7.3862728403466651;-2.6367860349243379;7.328723831161847;-
0.82538657295576856;3.0587174366787631];


IW1_1    =    [0.37444841065763013    -1.4129577512708835    -1.9820179860343781    -
0.96597197929904688  -0.081970406721104505  -1.3326140949796201  -1.2244491798918469  -
1.1187575042779068      -0.87035998362423461;3.6128809018999175      3.2173951205557367
0.97754992065566815     10.338277416233556     4.4493408541859756     18.625609153503092
16.332561518743539   -20.362008645919843   -26.580325001198773;-2.5624568223803195   -
2.5249495374986712   0.22522009190015813   -3.4438289616037419   -3.280031190449014   -
9.3289917171942172    -15.677936280543257    7.2849479523255329    12.278743202759596;-
51.084825520115814     23.015873732625444     30.94750176830598     3.6550679706568587
25.740635753820296    -36.405167885624216    -10.72879717951059     15.671413795987336
17.208346343380416;0.50212152741271698   -1.6101907886173501   -0.91678922467797352   -
```

```
0.44374655763784682    -0.8641754370199376    1.2708542462485748    0.54348842062360714
0.48444568155563666     2.4298446967979626;1.9042255769052392     0.80404686730400665
1.1335955206754409     1.381990267112347     1.1220096113197597     0.99541503615800497
1.2129916341762605 1.0737270518963984 0.31176872838885755];


% Layer 2

b2 = [-1.000000883412842;1.3749353572479331e-05;-1.0000174482665234];

LW2_1   =   [-0.0062422957139925241   1.2885404630670678e-08   1.0000001739369313   -
1.2558787766337351e-09 5.0102954479073749e-08 0.99375878738306456;0.83476939205943046
1.0000001932009654  1.6757095910432272e-07  1.0000000739226675  -9.8544692687228781e-08
-0.16524451785092353;-1.0573694974960179  -1.0000002325309683  -1.0000002834808355  -
1.0000000871135086 9.1930231633440667e-08 -0.057351951305608759];


% Output 1

y1_step1_ymin = -1;

y1_step1_gain = [2;2;2];

y1_step1_xoffset = [0;0;0];


% ===== SIMULATION ========


% Format Input Arguments

isCellX = iscell(X);

if ~isCellX, X = {X}; end;


% Dimensions

TS = size(X,2); % timesteps

if ~isempty(X)

    Q = size(X{1},2); % samples/series

else

    Q = 0;

end


% Allocate Outputs

Y = cell(1,TS);


% Time loop

for ts=1:TS


    % Input 1

    Xp1 = mapminmax_apply(X{1,ts},x1_step1_gain,x1_step1_xoffset,x1_step1_ymin);


    % Layer 1

    a1 = tansig_apply(repmat(b1,1,Q) + IW1_1*Xp1);


    % Layer 2

    a2 = repmat(b2,1,Q) + LW2_1*a1;


    % Output 1

    Y{1,ts} = mapminmax_reverse(a2,y1_step1_gain,y1_step1_xoffset,y1_step1_ymin);
```

```
end

% Final Delay States
Xf = cell(1,0);
Af = cell(2,0);

% Format Output Arguments
if ~isCellX, Y = cell2mat(Y); end
end

% ===== MODULE FUNCTIONS ========

% Map Minimum and Maximum Input Processing Function
function y = mapminmax_apply(x,settings_gain,settings_xoffset,settings_ymin)
y = bsxfun(@minus,x,settings_xoffset);
y = bsxfun(@times,y,settings_gain);
y = bsxfun(@plus,y,settings_ymin);
end

% Sigmoid Symmetric Transfer Function
function a = tansig_apply(n)
a = 2 ./ (1 + exp(-2*n)) - 1;
end

% Map Minimum and Maximum Output Reverse-Processing Function
function x = mapminmax_reverse(y,settings_gain,settings_xoffset,settings_ymin)
x = bsxfun(@minus,y,settings_ymin);
x = bsxfun(@rdivide,x,settings_gain);
x = bsxfun(@plus,x,settings_xoffset);
end
```

APPENDIX B - C# Cording for Developing interface in- between MATHLAB and C#

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace NeuralInter;
{
    public interface IFindPos
    {
        double[] Test9(double Num1, double Num2, double Num3, double Num4, double
Num5, double Num6, double Num7, double Num8, double Num9);
```

```
    }
}
```

APPENDIX C - C# Cording for Interfacing Kinect into C#

```
using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

using LightBuzz.Vitruvius;

using System.Speech.Recognition;

using System.Speech.Synthesis;

using Newtonsoft.Json;

using Advantech.Adam;

using NeuralNetwork2;

using Microsoft.Kinect;

using System.Windows.Threading;

using System.Threading;

using System.Net.Http;

using Advantech.Common;


namespace HomeBot

{

    /// <summary>

    /// Interaction logic for MainWindow.xaml

    /// </summary>

    public partial class MainWindow : Window

    {
```

```
FindPos2IFindPos net = new FindPos2IFindPos();
KinectSensor _sensor;
MultiSourceFrameReader _reader;
PlayersController _playersController;
HomeBotData _SensorData = new HomeBotData();
DispatcherTimer MainTimer;
SpeechRecognitionEngine recEngine = new SpeechRecognitionEngine();
private static ComPort Com1 = new ComPort(3);


private static System.Timers.Timer ComTimer;
private static System.Timers.Timer NetTimer;


JointType _start1 = JointType.ShoulderRight;
JointType _center1 = JointType.ElbowRight;
JointType _end1 = JointType.WristRight;

JointType _start2 = JointType.ElbowLeft;
JointType _center2 = JointType.ShoulderLeft;
JointType _end2 = JointType.SpineShoulder;

JointType _start3 = JointType.AnkleRight;
JointType _center3 = JointType.KneeRight;
JointType _end3 = JointType.HipRight;

JointType _start4 = JointType.HipLeft;
JointType _center4 = JointType.KneeLeft;
JointType _end4 = JointType.AnkleLeft;

JointType _start5 = JointType.SpineShoulder;
JointType _center5 = JointType.ShoulderRight;
JointType _end5 = JointType.ElbowRight;

JointType _start6 = JointType.WristLeft;
JointType _center6 = JointType.ElbowLeft;
```

```
JointType _end6 = JointType.ShoulderLeft;

JointType _start7 = JointType.Head;
JointType _center7 = JointType.Neck;
JointType _end7 = JointType.SpineShoulder;

JointType _start8 = JointType.SpineBase;
JointType _center8 = JointType.HipLeft;
JointType _end8 = JointType.KneeLeft;

JointType _start9 = JointType.KneeRight;
JointType _center9 = JointType.HipRight;
JointType _end9 = JointType.SpineBase;



public MainWindow()
{
    InitializeComponent();
    MainTimer = new DispatcherTimer();
    MainTimer.Interval = new TimeSpan(0,0,0,0,500);
    MainTimer.Tick += MainTimer_Tick;



    // TimerCallback tmCallback = SensorCom;
    // Timer timer = new Timer(tmCallback,null, 1000, 1000);
    //Console.WriteLine("Press any key to exit the sample");
    // Console.ReadLine();

    //_dtTimer = new DispatcherTimer();
    /// _dtTimer.Tick += new System.EventHandler(SensorCom);
    // _dtTimer.Interval = new TimeSpan(0, 0, 0, 1); //Timespan of 2 seconds
    //  _dtTimer.Start();

    ComTimer = new System.Timers.Timer(2000);
    // Hook up the Elapsed event for the timer.
    ComTimer.Elapsed += ComTimer_Elapsed;
```

```
            ComTimer.AutoReset = true;
            ComTimer.Enabled = true;

            NetTimer = new System.Timers.Timer(2000);
            NetTimer.Elapsed += NetTimer_Elapsed;
            NetTimer.AutoReset = true;
           // NetTimer.Enabled = true;

            ComOpen();
            MainTimer.Start();

            Thread.Sleep(3000);



        }
void Reader_MultiSourceFrameArrived(object sender, MultiSourceFrameArrivedEventArgs
e)
        {
            var reference = e.FrameReference.AcquireFrame();

            // Color
            using (var frame = reference.ColorFrameReference.AcquireFrame())
            {
                if (frame != null)
                {
                    if (viewer.Visualization == Visualization.Color)
                    {
                        viewer.Image = frame.ToBitmap();
                    }
                }
            }

            // Body
            using (var frame = reference.BodyFrameReference.AcquireFrame())
            {
                if (frame != null)
```

```
                {
                    var bodies = frame.Bodies();

                    _playersController.Update(bodies);

                    Body body = bodies.Closest();

                    if (body != null)
                    {
                        viewer.DrawBody(body);

                        angle1.Update(body.Joints[_start1],     body.Joints[_center1],
body.Joints[_end1], 50);
                        angle2.Update(body.Joints[_start2],     body.Joints[_center2],
body.Joints[_end2], 50);
                        angle3.Update(body.Joints[_start3],     body.Joints[_center3],
body.Joints[_end3], 50);
                        angle4.Update(body.Joints[_start4],     body.Joints[_center4],
body.Joints[_end4], 50);
                        angle5.Update(body.Joints[_start5],     body.Joints[_center5],
body.Joints[_end5], 50);
                        angle6.Update(body.Joints[_start6],     body.Joints[_center6],
body.Joints[_end6], 50);
                        angle7.Update(body.Joints[_start7],     body.Joints[_center7],
body.Joints[_end7], 50);
                        angle8.Update(body.Joints[_start8],     body.Joints[_center8],
body.Joints[_end8], 50);
                        angle9.Update(body.Joints[_start9],     body.Joints[_center9],
body.Joints[_end9], 50);

                        tblAngle1.Text = ((int)angle1.Angle).ToString();
                        tblAngle2.Text = ((int)angle2.Angle).ToString();
                        tblAngle3.Text = ((int)angle3.Angle).ToString();
                        tblAngle4.Text = ((int)angle4.Angle).ToString();
                        tblAngle5.Text = ((int)angle5.Angle).ToString();
                        tblAngle6.Text = ((int)angle6.Angle).ToString();
                        tblAngle7.Text = ((int)angle7.Angle).ToString();
                        tblAngle8.Text = ((int)angle8.Angle).ToString();
                        tblAngle9.Text = ((int)angle9.Angle).ToString();
```

```
                    _SensorData.neck = Convert.ToInt16(tblAngle7.Text);
                    _SensorData.shoLeft = Convert.ToInt16(tblAngle2.Text);
                    _SensorData.shoRight = Convert.ToInt16(tblAngle5.Text);
                    _SensorData.ElboLeft = Convert.ToInt16(tblAngle6.Text);
                    _SensorData.ElboRight = Convert.ToInt16(tblAngle1.Text);
                    _SensorData.HipLeft = Convert.ToInt16(tblAngle8.Text);
                    _SensorData.HipRight = Convert.ToInt16(tblAngle9.Text);
                    _SensorData.KneeLeft = Convert.ToInt16(tblAngle4.Text);
                    _SensorData.KneeRight = Convert.ToInt16(tblAngle3.Text);


                }
            }
        }

    }
```

APPENDIX D - C# Cording for Getting sensor information's

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace HomeBot
{
    class HomeBotData
    {
        public  double[] NuralData;

        public  int neck;
        public  int shoLeft;
        public  int shoRight;
```

```
public  int ElboLeft;
public  int ElboRight;
public  int HipLeft;
public  int HipRight;
public  int KneeLeft;
public  int KneeRight;

public static double lux;
public static double Tempatrature;
public static double Humidity;
public static double OutputPower;

public static string RStr;

public  int sit;
public  int sitR;
public  int sta;



public  bool HumenState;

public  bool NotHumanDetect = true;
public  bool HumanDetect = true;

public  bool sitting = true;
public  bool sitReading = true;
public  bool standing = true;
public  bool Invalid = true;



public  bool VoiceMode = false;
public  bool PatternMode = false;
public  bool EnableZone1 = false;
public  bool EnableZone2 = false;
```

```
public  bool Light1On = false;
public  bool Light1Off = false;
public  bool Light1Mid = false;

public bool Light2On = false;
public bool Light2Off = false;
public bool Light2Mid = false;

public bool AutoMode = false;
public bool ManualMode = false;



public bool Zone1LightH = true;
public bool Zone1LightL = true;

public bool Zone2LightH = true;
public bool Zone2LightL = true;



//light and temp setting

public bool LightH = false;
public bool LightM = false;
public bool LightL = false;

public bool TempH = false;
public bool TempL = false;

//temp Alert System
public bool TempAlHi = true;
public bool TempAlLo = true;

//light level Alert System
public bool LightAlHi = true;
```

```
        public bool LightAlMd = true;
        public bool LightAlLo = true;

        public bool LightAlHi1 = true;
        public bool LightAlMd1 = true;
        public bool LightAlLo1 = true;

        public bool LightAlHi2 = true;
        public bool LightAlMd2 = true;
        public bool LightAlLo2 = true;

        public bool LightAlHi3 = true;
        public bool LightAlMd3 = true;
        public bool LightAlLo3 = true;

        public bool LightAlHi4 = true;
        public bool LightAlMd4 = true;
        public bool LightAlLo4 = true;

        public bool LHi = true;
        public bool LMd = true;
        public bool LLo = true;



    }
}
```

APPENDIX E - C++ Cording for interfacing Smart Object

```
#include<ESP8266WiFi.h>
const char* ssid = "xxx";
const char* password = "xxxxxxx";

WiFiServer server(80);
int val = 0;
int dimming = 128;

void setup() {
        Serial.begin(9600);

  pinMode(D1, OUTPUT);// Set AC Load pin as output
  pinMode(D5, INPUT_PULLUP);
  attachInterrupt(digitalPinToInterrupt(D5), zero_crosss_int, RISING);

        Serial.println();
        Serial.println();
        Serial.print("Connecting to network");
        Serial.println(ssid);
        WiFi.mode(WIFI_STA);
        //WiFi.hostname("Smart Lamp");
        //WiFi.enableSTA(1);
        WiFi.begin(ssid, password);
        WiFi.config(IPAddress(192, 168, 43, 105), IPAddress(192, 168, 43, 105),
IPAddress(255, 255, 255, 0));

        while (WiFi.status() != WL_CONNECTED) {
                delay(500);
                Serial.print(".");
        }
        Serial.println("");
        Serial.println("WiFi connected");

        // Start the server
```

```
        server.begin();
        Serial.println("Server started");



        // Print the IP address
        Serial.println(WiFi.localIP());
}
void zero_crosss_int()  //function to be fired at the zero crossing to dim the light
{



  int dimtime = (75*dimming);    // For 60Hz =>65
  delayMicroseconds(dimtime);    // Wait till firing the TRIAC
  digitalWrite(D1, HIGH);   // Fire the TRIAC
  delayMicroseconds(10);                          //   triac   On   propogation   delay
// (for 60Hz use 8.33) Some Triacs need a longer period
  digitalWrite(D1, LOW);     // No longer trigger the TRIAC (the next zero crossing
will swith it off) TRIAC


}
void loop() {


        WiFiClient client = server.available();
        if (!client) {


                return;
        }


        // Wait until the client sends some data
        Serial.println("new client");
        while (!client.available()) {
                delay(1);
        }


        // Read the first line of the request
        String req = client.readStringUntil('\r');
```

```
        Serial.println(req);
        client.flush();

        // Match the request

        if (req.indexOf("/gpio/0") != -1) {
                val = 128;
        }
        else if (req.indexOf("/gpio/1") != -1) {
                val = 96;
        }

        else if (req.indexOf("/gpio/2") != -1) {
                val = 64;
        }

        else if (req.indexOf("/gpio/3") != -1) {
                val = 32;
        }

        else if (req.indexOf("/gpio/4") != -1) {
                val = 0;
        }
        else if (req.indexOf("/gpio/5") != -1) {
                String      p      =      "HTTP/1.1      200      OK\r\nContent-Type:
text/html\r\n\r\n<!DOCTYPE HTML>\r\n<html>\r\n";
                p += (val) ? "ON" : "OFF";
                p += "</html>\n";

                // Send the response to the client
                client.print(p);
                delay(1);
                return;
        }

        else {
```

```
                Serial.println("invalid request");
                client.stop();

                return;
        }
        // Set GPIO2 according to the request
        //analogWrite(D2, val);
 dimming = val;

        client.flush();

        // Prepare the response
        String s = "HTTP/1.1 200 OK\r\nContent-Type: text/html\r\n\r\n<!DOCTYPE
HTML>\r\n<html>\r\nLIGHT is now ";
        s += (val) ? "ON" : "OFF";
        s += "</html>\n";

        // Send the response to the client
        client.print(s);
        delay(1);
        Serial.println("Client disonnected");

        // The client will actually be disconnected
        // when the function returns and 'client' object is detroyed

}
```