Controlling a remotely located Robot using Hand Gestures in real time: A DSP implementation

Jagdish Lal Raheja, Gadula Arun Rajsekhar, Ankit Chaudhary

Machine Vision Lab, Digital Systems Group
CSIR-Central Electronics Engineering Research Institute
Pilani, Rajasthan, India - 333031
{jagdish@ceeri.ernet.in; garaj32@gmail.com; dr.ankit@ieee.org}

***Abstract*— Telepresence is a necessity for this time as we can't reach everywhere and also it's useful in saving human life at dangerous places. A robot, which could be controlled from a distant location, can solve these problems. This could be via communication waves or networking methods. Also controlling should be real time and so smooth that it can actuate on every minor signal in an effective way. Robot control through hand gesture in real time has been an investigative area among researchers. This paper discusses a method controlling a robot over the network from a distant location. The robot is controlled by hand gestures which were captured by camera live. A DSP board TMS320DM642EVM was used to implement image pre-processing and fastening the system. PCA was used for gesture classification and robot actuation would happen according to predefined procedures. Classification information was sent over the network in the experiment. This method is robust and could be used to control any kind of robot over distance.**

***Keywords:* Robot Control, Hand Gestures, Vision Robotics, Ethernet, Telerobotics, PCA**

I. INTRODUCTION

There have been many robotic applications where a smooth and real time control is needed according to human requirements. Sometimes it's very difficult to transfer input to the robot using keyboard as human need to convert his input from ideas to keyboard values. It is tricky in the case of robot movement in different directions. Robot control through hand gesture is a developing area where a human can ask the robot to do something just by hand movements and no need to enter numeric values using the keyboard. Also, the input information could be transferred over the network to control a robot remotely. This can be used to control any type of robot for different actuation [1]. It can be used to control the robotic arm remotely where the human presence is not conducive for his/her safety or to control a TV or PC monitor in order to replace the infrared remote [2].

This paper demonstrates a system for controlling the robot by merely showing hand gestures in front of a camera. High-resolution camera connected to a Digital Signal Processing (DSP) based embedded board, which captures the image in real time and followed by the image pre-processing. In previous work, the algorithmic computations were performed on PC or on some control devices [3]. The pre-processing takes maximum time in any image processing algorithm [4-5]. The block diagram of the approach is shown in Fig. 1.

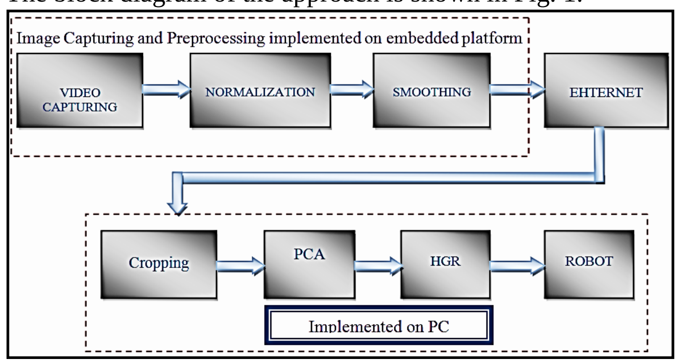

Fig.1. Block diagram for System

This work implements algorithms on a DSP processor TMS320DM642EVM to determine actuation using various hand gestures without touching the surface of the screen. This provides manyfold increase in speed over PC implementation. Robust real-time performance is thus made possible and this increases the scope of applications to include ones where high frame rate is necessary. The DSP board setup is shown in Fig. 2.

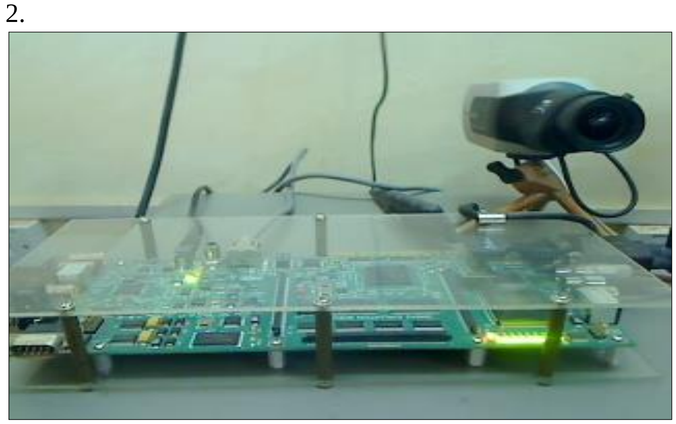

Fig.2. DSP processor (DM642EVM) with high-resolution camera

II. BACKGROUND

The Idea of telepresence, as well as its past and future, is well described by Tachi [6]. He said, "Teleexistence technology allows for highly realistic sensation of existence in remote places without actual travel". He discussed many implementation of robots, made for teleoperation in the specific field. Ma [7] presented a haptic glove with teleoperation experiments. This was capturing each finger's movement for operations. Li [8] showed virtual robot teleoperation based on hand gesture recognition. Adaptive Neuro-Fuzzy Inference Systems and Support Vector Machines were used for classification. Wen [9] presented a hand gesture based on robotic surgery based on augmented reality. Hand gesture signals were used to implement specified RF needle insertion plans.

Greenfield [10] discusses touchless hand gesture device controller with an image sensor. This method was based on hand motion and optical pattern recognition. DU [11] presented hand gesture based controlling of the robot manipulator. He used Kalman and particle filters to track hands. Although it was a marker less method but used leap motion to detect hand positions. Samsung has also built a TV which could be controlled using hand gesture from a certain distance [12].

Our presented method does not use any sensor or marking scheme. It is based on Natural Computing where anyone can use the system irrespective of skin color and hand size. The robot used in this experiment would also provide its surroundings visual so that the controller can see what robot can do at that location. Principle component analysis (PCA) was used for classification.

III. IMPLEMENTATION

The developed system is able to use live video stream for gesture identification. It sniffs frames of the live video stream in some time interval. Proposed technique to control robotic system using hand gesture display is divided into four subparts:

A. Gesture extraction from video stream
B. Extract region of interest from a frame
C. Determine gesture by pattern matching
D. Determine control instruction and robot actuation

The working of the PC implementation of the system is explained in Fig 3. These techniques were implemented in Matlab, Embedded Matlab, Simulink, and Code Composer Studio (CCS). An XDS560 PCI JTAG Emulator was fitted in a PCI-slot to support high-speed RTDX on the enabled processor for real-time data transfer over 2 MB/second.

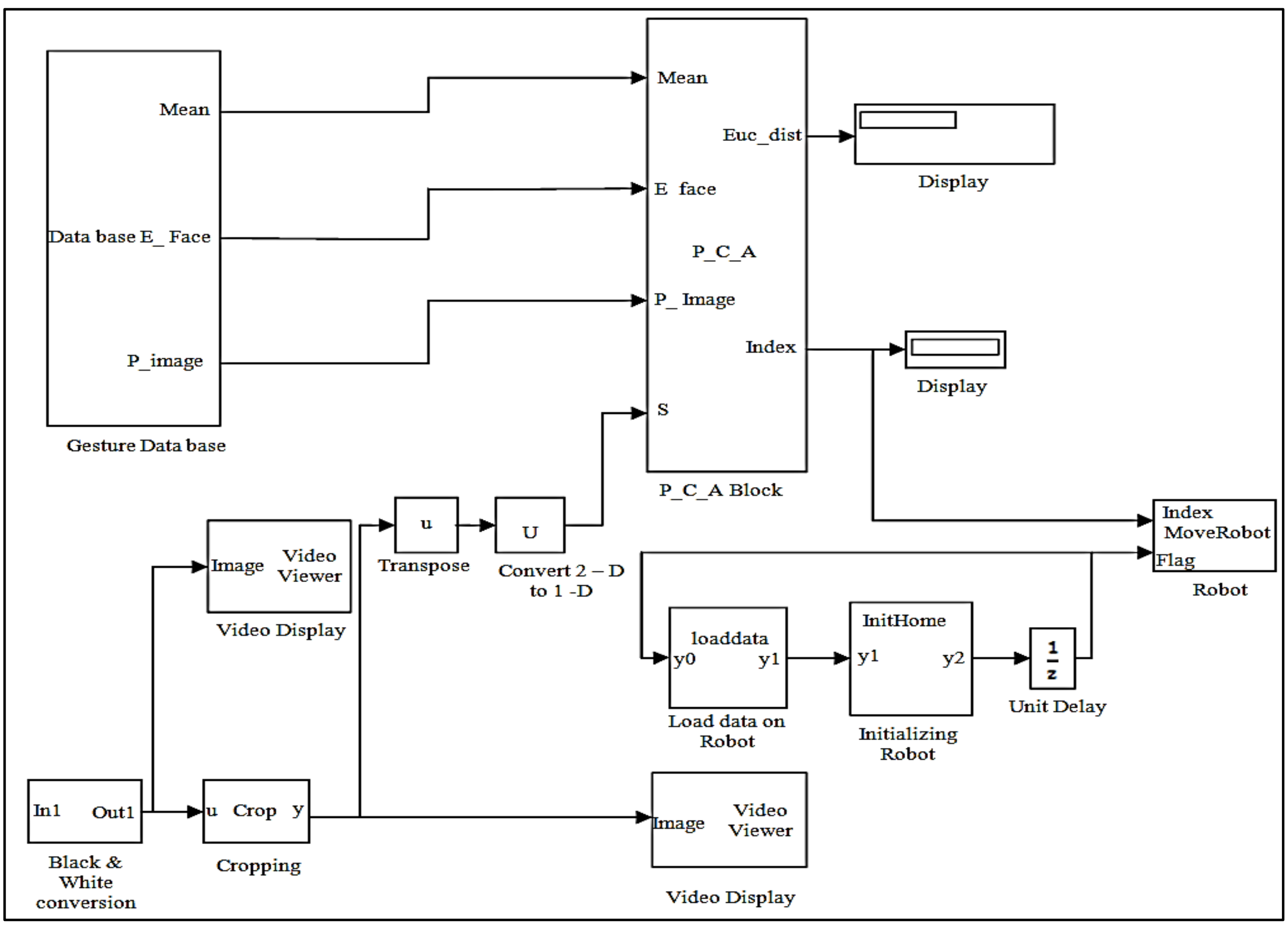

Fig.3. Implementation block diagram

A. Gesture extraction from video stream

The video was captured by the high-resolution camera and image pre-processing techniques are applied on it frame by frame [13]. Two light sources are required to remove shades, one below the camera and other above the camera. The camera can be NTSC or PAL. From video stream, one frame was captured in each 1/5 second. The preprocessing techniques such as smoothing and normalization reside on DSP processor DM642EVM. Fig 4 shows these two functions being executed on the DSP board. More details about pre-processing on DSP board are discussed in [14].

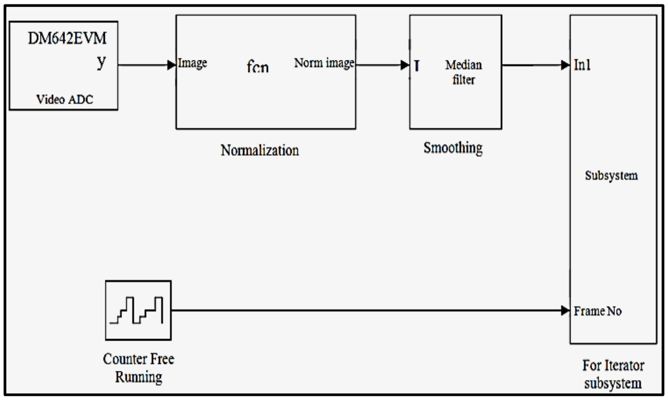

Fig.4. Image pre-processing on DSP board

This brings a big reduction in execution time for the whole algorithm. After these, the image frame was transferred to PC using UDP target to HOST Ethernet communication as shown in Fig 5.

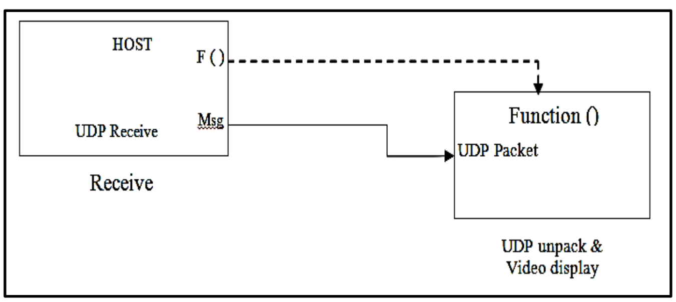

Fig.5. UDP Target to host Communication

While Fig 6 shows that data being sent from PC to HOST. The remaining section of the system such as Cropping and PCA are running on PC.

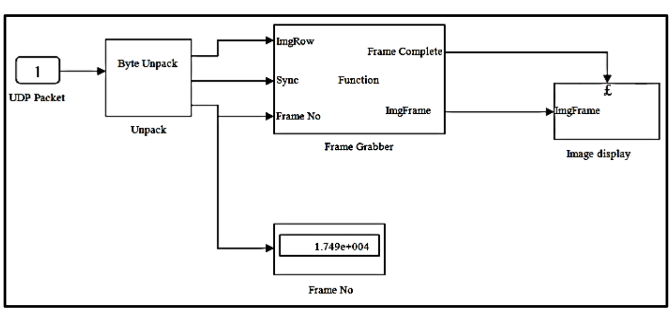

Fig.6. Data Transfer to Host

Each frame is analyzed to detect motion of the object and to find the frame where there is little or no motion. Using global thresholding each frame is converted into a binary image and then it is compared with two previously captured frames to find the differences in their pixel values. The difference between pixel values is counted with both frames and added to find total points of difference. Since the binary image has only values either 1 or 0, so XOR function can give locations where mismatch occur. If *frame1*, *frame2*, *and frame3* are three matrixes containing three frames captured in three continuous time slot respectively then:

$$fr1 = frame1\ XOR\ frame3$$
$$fr2 = frame2\ XOR\ frame3$$
$$mismatch\_matrix = fr1\ OR\ fr2$$

A total point of mismatch is equal to the total number of *one*s in matrix *mismatch_matrix*. If the point of a mismatch for any frame is less than 1% of a total number of pixels in that frame, it is motionless frame (i.e. the frame which is supposed to be contained complete gesture). This frame is selected for further processing of gesture recognition.

B. Extract ROI from selected frame

The frame may contain some extra information other than hand gesture, which is not required. Therefore before proceeding further, these extra parts must be removed. For cropping, convert selected frame into a black-and-white image using global thresholding. Now extra black rows and columns are determined, from where the object of interest starts appearing. This is done by searching from each side of the binary image and moving forward until white pixels encountered are more than offset value. Experimental results show that offset value set to 1% of total width gives a better result for compensation of noise. If size of selected image is ***m*n*** then:

$$Hor\_offset = m/100$$
$$Ver\_offset = n/100$$

Min_col= minimum value of column number where total number of white pixels are more than *Hor_offset*.
Max_col= maximum value of column number where total number of white pixels are more than *Hor_offset*.
Min_row= minimum value of row number where total number of white pixels are more than *Ver_offset*.
Max_row= maximum value of row number where total number of white pixels are more than *Ver_offset*.

Now remove those parts of a hand which is not used in gesture presentation i.e. removal of the wrist, arm etc. Because theses extra parts are of variable length, pattern matching with gesture database gives unwanted result due to limitations of the gesture database. So part of hand before wrist is to be cropped out. Statistical analysis of hand shape shows that either we pose palm or fist in which width is lowest at wrist and highest at the middle of the palm. Therefore extra hand part can be cropped out from wrist by determining the location where minimum width of vertical histogram found. Cropping point is calculated as:

$$Global\ Maxima = \text{column number where height of histogram is highest}$$

Cropping point = column number where height of histogram is lowest

The cropping point is searched between the first column and column number of Global Maxima because the height of histogram automatically decreases exponentially due to the thinness of finger. So in this way we crop from wrist instead of from fingers. Figure 7(a) shows a frame containing hand gesture before and after cropping.

C. Determine gesture by pattern matching

Finally cropped gesture images were resized to matched database image sizes. Gesture database was created and stored in the binary form of size 60x80. PCA is faster when compared to neural network method which requires training database and more time along with high computation power. Also components in a human hand (fingers, palm, fist, etc.) are large enough when compared to noise. PCA method was used for pattern matching [15]. Control instruction generation for Robot

Different functions corresponding to each meaningful hand gesture were written and stored in a database for controlling a robotic hand. PUMA robotic model was used for testing robotic control. Ten different hand gestures are used for designing 10 specific moves of robotic hand of each arm in precise angle and direction. Movement commands were designed and written as a function in robot specific language corresponding to each meaningful hand gesture. Hand gesture processing, robot movement, and robot view are shown in Fig 7.

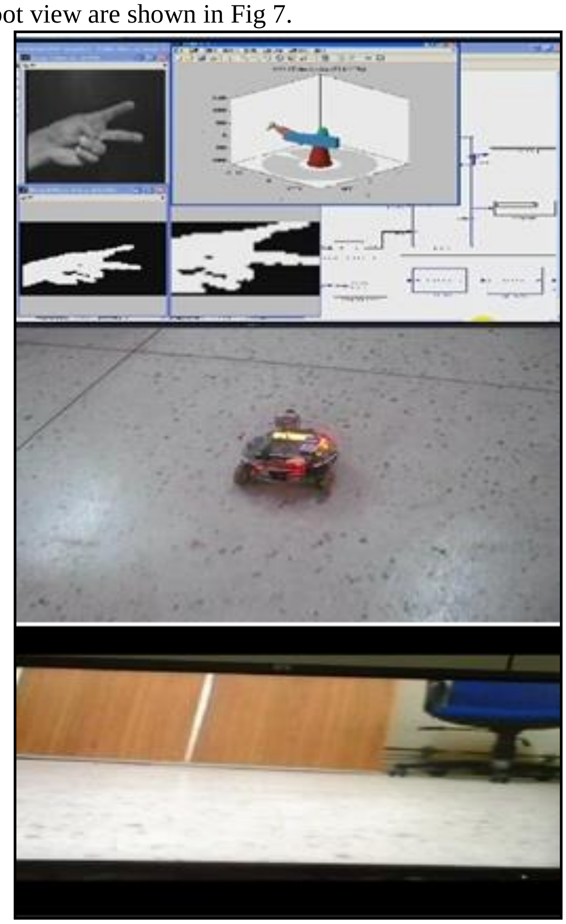

Fig.7. Robot movement and its vision

When a gesture is presented in front of a camera, it would be processed and the matched with stored gesture from the database. Instruction set corresponding to that gesture would be identified and passed to the robot for execution of stored commands. Immediately a robot would do action accordingly. The robot continues to do action according to gestures, shown by used on camera. When the robot is moving, visual information of its surrounding was sent wirelessly to the user, as this robot having a camera. It helps the user in obstacle avoidance and makes robot movement accordingly.

IV. Results & conclusion

The need of the existence in real-time inspired us to implement this system. A robot can be controlled over the network using hand gestures. The robot will move as per the gesture and would do movement and manipulation as per instruction. Technique proposed here is tested in the environment shown in figure 7. Database of gesture is stored into binary form of size 60X80 pixels so it takes less time and memory space during pattern recognition. Due to use of cropped image of gesture, our database becomes more effective as one image is sufficient for one type of gesture presentation. So neither we need to store more than one image for same gesture at different positions of image, nor have we to worry about positions of hand in front of camera. Experimental Result shows that it can detect hand gesture when we stop moving hand for one second. Accuracy of this system is 95%.This method can be applied to any type to the robot as Robot instructions were mapped on hand gesture signals. Currently, only ten gestures are implemented for experimental purpose, which could be extended as per requirement.

The system was implemented on DSP boards which bring its action time within real-time constraints. The robot is also providing visual information which could be used for different purposes including surveillance and object manipulation. In future, we would like to build the whole system-on-chip, so that it would be much smaller and low power consumption for better performance.

V. Acknowledgement

This research has been carried out at CSIR-Central Electronics Engineering Research Institute (CEERI), Pilani, India and Authors would like to thank Director, CSIR-CEERI for his active encouragement and support.

References

[1] B. Moniruzzaman, and R. Picking. "Gesture-controlled user interfaces, what have we done and what’s next." Proceedings of the Fifth Collaborative Research Symposium on Security, E-Learning, Internet and Networking, Darmstadt, Germany. 2009.

[2] D. Ionescu, B. Ionescu, C. Gadea and S. Islam, "An intelligent gesture interface for controlling TV sets and set-top boxes," *Applied Computational Intelligence and Informatics, 2011 6th IEEE International Symposium on*, Timisoara, 2011, pp. 159-164.

[3] A. Chaudhary, J. L. Raheja, “Bent fingers’ angle calculation using supervised ANN to control electro-mechanical robotic hand”, Computers & Electrical Engineering Elsevier, Vol. 39 , issue 2, 2013, pp. 560-570.

[4] J. Choi, N. Ko, D. Ko, Morphological Gesture Recognition Algorithm, Proceeding of IEEE region 10th international conference on Electrical and Electroic Technology, Coimbra, Portugal, 2001, pp.291-296.

[5] A. Chaudhary, J.L. Raheja, K. Das, S. Raheja, "A Survey on Hand Gesture Recognition in context of Soft Computing", Published as Book Chapter in “Advanced Computing” CCIS, Springer Berlin Heidelberg, Vol. 133, 2011, pp. 46-55.

[6] S. Tachi, , Telexistence, Lecture Notes in Computer Sciences, 2015, Springer, pp. 229—259.

[7] Z. MA and P. Ben-Tzvi, "RML Glove—An Exoskeleton Glove Mechanism With Haptics Feedback," in IEEE/ASME Transactions on Mechatronics, vol. 20, no. 2, April 2015 pp. 641-652.

[8] C. Li, H. Ma, C. Yang and M. Fu, "Teleoperation of a virtual iCub robot under framework of parallel system via hand gesture recognition," Fuzzy Systems, 2014 IEEE International Conference on, Beijing, 2014, pp. 1469-1474.

[9] W. Rong et al., Hand gesture guided robot-assisted surgery based on a direct augmented reality interface ,Computer Methods and Programs in Biomedicine , Vol 116 , Issue 2 , 68 - 80.

[10] USPTO Patent, Application No. 20080256494, 10/16/08

[11] G. Du and P. Zhang, "A Markerless Human–Robot Interface Using Particle Filter and Kalman Filter for Dual Robots," in IEEE Transactions on Industrial Electronics, vol. 62, no. 4, April 2015, pp. 2257-2264.

[12] http://www.samsung.com/ph/smarttv/common/guide_book_3p_si/thumb_up.html

[13] A. Chaudhary, J.L. Raheja, K. Das, “A Vision based Real Time System to Control Remote Robotic hand Fingers”, In proceedings of the IEEE International Conference on Computer Control and Automation, South Korea, 1-3 May, 2011, pp. 118-122.

[14] J.L. Raheja, B. Ajay, A. Chaudhary, “Real Time Fabric Defect Detection System on an Embedded DSP Platform”, Optik: International Journal for Light and Electron Optics, Elsevier, Vol. 124, Issue 21, 2013, pp. 5280-5284.

[15] S. K. Raju, et al. "Principal Component analysis for Gesture Recognition Using SystemC." Advances in Recent Technologies in Communication and Computing, ARTCom'09. International Conference on. IEEE, 2009.