Motion capture for Animation

Naman Luniya[1], Rishav[2], Ronak Jain[3]

[1,2,3] Department of Computer Science Engineering, MIT School of Engineering,MIT-ADT University, Pune, Maharashtra, INDIA

Abstract

Artificial intelligence and machine learning are becoming increasingly popular. In the realm of performance capture for the entertainment sector, we are investigating the use of artificial intelligence and machine learning. The present methods are generally prohibitively expensive for the general public. We are concentrating on using artificial intelligence and external software (such as Unity) together to create amazing motion capture outcomes. This will lower the cost of inputs while also increasing innovation.

Keywords – Performance capture, Motioncapture, AI/ML, Unity.

1. Introduction

The expansion of technological impact in the entertainment sector has shown to be effective. Motion Capture (MoCap) is the process of using cameras and specifically built suits to capture the movement of objects or people. Its impact on the entertainment business has been significant. Motion capture has been popular in movies and video games since the 1970s. In comparison to previous procedures, it has been complimented for its reduced latency, replication of complex moves, and data production. The demand for specific software and hardware, as well as the expense of relevant inputs, has been widely criticized. The goal of this project is to drastically lower the cost of MoCap. The app eliminates the need for sophisticated lenses and motion capture suits. With the help of Python's

OpenCV, scripting and Unity, we were able tocomplete the job.

The user uploads video to the app, which is thenpre-processed with OpenCV & Unity, which aids in the creation of animated clips of the user.These procedures demonstrate the capacity to capture performance in a coherent way.

2. Literature Survey

There has been work conducted in the field of face detection & its recognition along with development of mocap suits for better performance capture. The previous works have been focused on for better detection and capturing of coordinates. The prominent methodsfor face detection in this field are Eigen face detection, ADABOOST algorithm, Convex hull algorithm and Voila & Jones face detection. Thedevelopment in capturing of coordinates will help to create animations easier than to follow the tedious & expensive method of mocap suits.

The process of face detection is done using a single algorithm or with a combination of algorithms. Eigen Face detection is used with a combination of other algorithms[2]. It lacks in performance in comparison with the Haar Cascade classifier. The Haar cascade classifier isthe most efficient in terms of face detection and recognition [2,4].The downside being its sensitivity to the background. The ADABOOST algorithm is efficient in capturing coordinates for the detection, especially for hand and fingers [4]. It has shown better speed and accuracy with a drawback for longer trainingperiods[4,13].

Motion capture has been prominently done usingthe motion capture suits. The suits are expensivefor the general populace [8]. The work-faced issues related to capturing of limbs, legs & coordinates. Studies conducted on its cost effectiveness along with maintaining its standardstill lack solidity. Thus, there is a need to find alternate pathways for motion capturing.

Our intention is to develop an motion capturingapplication which captures the body and its coordinates, then, with the use of an external software such as Unity or Blender produces an animated result which is as good as the Mocap suits. The use of Python & C# for its OpenCV library and its easy collaboration with the Unityscripting plugin will ease our task.

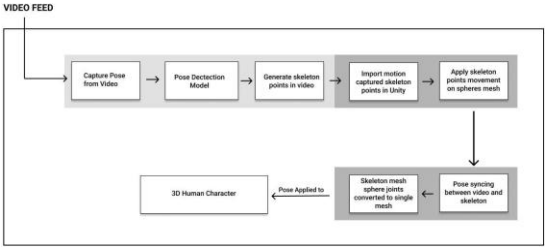

Figure 1: System Architecture The figure illustrates a proposed system of motion capture of a human using a combinationof hardware and software components.

3. Proposed System

A. System Architecture

For motion capture, it makes use of deep learning libraries. When the user first launches the app, they will be requested to upload a recorded video. Thereafter, pose detection will begin tracing the body's posture based on the video they provided. This will map to various body parts and collect angles between joints, as well as check for points in the video feed and attempt to capture different points and turn them into spheres. While this is happening, a person's face marks and distinct spots on their face will be captured and applied to a virtual character.

This will provide users with a unique experience, making it a fascinating great option and assist the entertainment business. Since it contains numerous character contents, the gaming business is the perfect application case for this.

B. Activity Diagram:

An activity diagram is a behavioral diagram that depicts the behavior of a system. An activity figure illustrates the system behavior from inception to delivery, emphasizing the different evaluation paths that exist during the activity.

Figure 4: Activity Diagram

The proposed system first recognizes the body in an input image or the video, then the model extracts image/video coordinates. The Unity software helps to create a model using basic scripting of python & C#. The external programme allows us to generate an animated rendition of our model, which is efficient enoughin comparison to MoCap suits.

5. Results & Discussions

The scripting enables the model's MoCap rendering to be developed without the use of an expensive suit or several lenses. The pre recordedvideo is first uploaded in python script which locates the 33 points of the body and stores theminto the text file.

The text file in which the landmarks are locatedis copied to Unity where C# script is used to get33 different spheres in those landmarks that combine together to form a human skeleton which will run according to different landmarksproduced from a python file, these spheres syncwith the landmarks and a video of the human skeleton is produced.
The test findings are represented in the diagramsbelow.

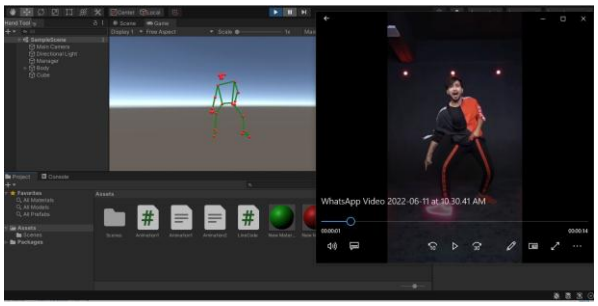

Figure 5.1: Coordinates extraction

Figure 5.2: Coordinates Extraction

6. Conclusion

Motion capture is a technology which has beenin use for a long time, but in terms of performance, animation is still in its infancy.

Even though improved tools for working with data are now available, technical competence is still required. People are increasingly employingmotion capture on a regular basis as a result of the increase of performance capture initiatives. Itwas difficult to include all components of motion capture in a single essay, particularly thesoftware-related ones. In the computer animation industry, motion capture has been pricey, but it is now universally recognized as a valuable weapon in the visual effects armory, and it will continue to be for a long time.

References

[1] Tejashree Dhawle, Urvashi Ukey, Rakshandha Choudante, "Face Detection and Recognition using OpenCV and Python", International Research Journal of Engineering and Technology (IRJET), Volume: 07 Issue: 10 |Oct 2020, pp. 1269-1271

[2] Emami, Shervin & Suciu, Valentin, "Facial Recognition using OpenCV", Journal of Mobile, Embedded and Distributed Systems, Vol. IV, No.1, 2012, pp.39-43.

[3] Mahamkali, Naveen Kumar & Ayyasamy, Vadivel, "OpenCV for Computer Vision Applications", Journal of Mobile, Embedded and Distributed Systems, Proceedings of National Conference on Big Data and Cloud Computing (NCBDC'15), March 20, 2015, pp.52-56.

[4] Prof. P Y Kumbhar, Mohammad Attaullah, Shubham Dhere, Shivkumar Hipparagi, "Real Time Face Detection and Tracking Using OpenCV", International Journal for Research inemerging Science and Technology, Volume-4, Issue-4, APR-2017, pp.39-43.

[5] Tsung-Han Tsai, Chih-Chi Huang and Kung-Long Zhang, "Embedded Virtual Mouse System by Using Hand Gesture Recognition",International Conference on Consumer Electronics-Taiwan (ICCE-TW), 2015, pp.352-352.

[6] Sherin Mohammed Sali Shajideen, Preetha V H, "Hand Gestures - Virtual Mouse for Human Computer Interaction", International Conferenceon Smart Systems and Inventive Technology, IEEE, 2018, pp.543-546.

[7]R. Suriya, Dr. V. Vijaya Chamundeswari, "A Survey on Hand Gesture Recognition for Simple Mouse Control", International Conference on Information Communication and Embedded Systems-2014

[8] Bradwell, Ben & Li, Baihua. (2008). A Tutorial on Motion Capture Driven Character Animation. Proceedings of the 8th IASTED International Conference on Visualization, Imaging, and Image Processing, VIIP 2008.

[9] Sidrah Liaqat, Kia Dashtipour, Kamran Arshad, Khaled Assaleh, Naeem Ramzan, "A Hybrid Posture Detection Framework: Integrating Machine Learning and Deep Neural Networks", IEEE SENSORS JOURNAL, VOL. 21, NO. 7, APRIL 1, 2021, pp.9515-9522.

[10] Niyati Gosaliaa, Priya Jainb, Ishita Shah, Dr. Abhijit R. Joshid, Neha Katree, Sameer Sahasrabudhe, "3D Gesture-Recognition BasedAnimation Game", International Conference onAdvanced Computing Technologies and Applications (ICACTA 2015), pp.713-717.

[11] Deepak Sharma, "A Review Paper on Motion Capturing Technology for Gaming", 2020 IJCRT | Volume 8, Issue 12 December 2020, pp.3156-3160.

[12] Lesage, Martin & Cherkaoui, Omar & Abouzaid, Faisal & Poirier, Martin & Raîche, Gilles & Riopel, Martin. (2007). Blender pluginimplementation for 3D collaborative work. 2557- 2569. 10.1109/ICSMC.2007.4413770.

[13] Gaurav, Ruchi & Kadbe, Premanand. (2015). Real time finger tracking and contour detection for gesture recognition using OpenCV. 2015 International Conference on Industrial Instrumentation and Control, ICIC 2015.974-977. 10.1109/IIC.2015.7150886.

[14] Lai, Hsiang & Lai, Han. (2014). Real-Time Dynamic Hand Gesture Recognition. Proceedings - 2014 International Symposium onComputer, Consumer and Control, IS3C 2014. 658-661. 10.1109/IS3C.2014.177.

[15] Rajendran, Gopika & Thomas, Ojus. (2020). Virtual Character Animation based onData-driven Motion Capture using Deep Learning Technique. 333-338. 10.1109/ICIIS51140.2020.934269