MOTION CAPTURE TECHNOLOGY

Mr. Prakash Hongal[1], Mr. Arunkumar Joshi[2], Ms. Jnaneshwari A Doddamani[3]

[1,2]*Asst. Prof., Dept. of Computer Science and Engineering SKSVMACET, Laxmeshwar, Karnataka ,India*
[3]*Final Year Student Dept. of Computer Science and Engineering SKSVMACET, Laxmeshwar, Karnataka,Inadia*

ABSTRACT

In today's world capturing of motion from a video stream is a growing and challenging area. Motion capture system (Mocap) is widely used in today's realistic world including sports, entertainment, video gaming and many more. In this paper, firstly we provide an overview of different types of Mocap on the basis of Marker-based Mocap and Markerless Mocap along with their merits and demerits. Secondly, we discuss about the Applications of Mocap.and algorithms for real-time motion capture.

Keywords: *Animation, Mocap system, object detection and tracking, 3D animation, 2D animation, real-time system*

---

1. INTRODUCTION

Animation emerged out in the early 20th century in the year 1911.The small changes created an illusion which made the character in motion. It also helps the creators to control the movements of the character. Animation is basically a process of converting several images into an animation so that it seems like something is in motion. In 2D animation, the objects which are in motion are also called as sprites [1,2]. A sprite is an image to which, a specific location is allocated. To animate the sprite the location of the sprite is slightly changed frame by frame so that the sprite appears to be in motion. So there had to be something to adjust that flaw and that was the time when Motion Capture (Mocap) came into existence. Increase the speed of animation and decrease the work load on the animators. A technique was introduced which was called Mocap. By using this technique the movement of the real world objects can be captured and transfer the data into a tridimensional model. Since then the Mocap has evolved and it is most commonly used in film-making and video gaming industry so as to provide better results in less time, less work load on the animator and animation of faces to animate character's faces [2-4]. With the advancement of the technology, it is very difficult to bring the characters into motion so the mocap comes to play. By using it, movements can be created and hence the work load can be minimized. That is how the cut scenes and various character movements are made. Another place where mocap is applicable is the Film Industry. One of the leading studios using mocap is the MARVEL studio which is probably known to all. Mocap is used in various characters like the Hulk, the motion of the actor is recorded and the recorded data is then pasted on the character model and that is how the animated character hulk is given movement[5,7,8].

In Real-time system is difficult to define precisely due to the many interpretations and perspectives. From the engineering design perspective, we believe the most appropriate description is a system where the temporal aspects of its behavior are part of its specification" [1]. This implies that the system must respond in a reasonable time or before a deadline in a reliable manner. In general, one can divide the processing of motion capture into four functional modules: initialization, tracking, poses estimation, and recognition [2]. The real-time property plays a significant role in each stage of motion capture, especially in tracking. A case in point is the currently fast-developing Virtual Reality (VR) techniques and systems. One of the fundamental characteristics of VR is interaction [3], which dictates that the VR equipment should be able to capture the motion in time and then give a real-time response.

MOTION CAPTURE SYSTEMS

Motion capture is a technique which can't be replaced by any other technique. However, Motion based system (Mocap) is not the only technique used for animation. It is necessary to understand several ways and techniques of animation to make out a difference and see that why capture motion is the best alternative used for animation. Basically, Mocap is broadly categorized into (a) Marker-based Motion Capture and (b)Markerless Motion Capture [12, 14, 16].

Marker-based Motion Capture

In literature, there are several types of marker based motion capture systems. They are present to capture the motion of an object in video data. Some of them are listed below:

- **Acoustical Systems**

  In these systems, sound transmitters are positioned on the actor's main joints and three receptors are placed in the capture radius. Then, emitters are activated sequentially and generate a set of frequencies that are picked by the receptors and used to estimate the frequencies of emitters in 3D space. Due to sequential firing of transmitters, sometimes it becomes very difficult to gather the correct data[12,21].

Fig.1 Sound Transmitter Setup for Acoustical Systems [17]

- **Mechanical Systems**

  This system comprises of sliders and potentiometers that are arranged in a desired joint and also enable the display of their locations as shown in Fig.3.Despite being a traditional approach main advantage of these system is that they are not affected by any field of magnet or redundant reflections which make them easy to use [6,10].

Fig.2 Mechanical motion capture system

- **Magnetic Systems**

The workstation used for acquisition and processing of data is low-priced but the Fig 3. (A) Input Using Transmitters (B) Wireframe Model (C) 3D Model (D) Final Animated Object precision of data is moderately sky-scraping as shown in Fig.4. Sampling rate of these systems are approximately 100 frames per second (fps). These systems have one major disadvantage, i.e., a large numbers of cable wires are used to connect with the antenna which results in reduction of degree of freedom. To overcome this problem, Yabukami et.al. [8] developed a system where this issue is resolved.

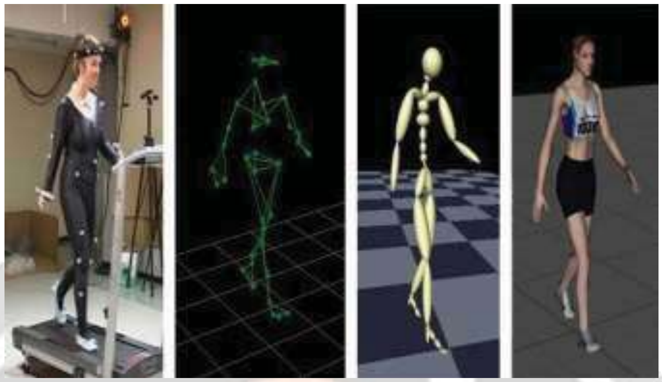

Fig.3 (A) Actor with Magnetic Mocap
(B) Closer View: Magnetic Mocap Sensor.

- **Optical Systems**

  In this mocap system, the actor uses a specially designed suit with reflectors which are located in their key articulations. In this system, a camera with high resolution is located to keep track of movements of the actors as shown in Fig.5.. Major disadvantage of these systems are transmitter occlusion, in case of tiny or small objects such as hand or interaction with others [10, 11].

Fig.4 Optical Capture Systems

Markerless Motion Capture

There is no special equipment needed to track the motion of an actor. The motion can be easily recorded from a videodata sequence using motion based algorithm used to track and detect the objects. This process is done using software by eliminating all the limitations including computational constraints as shown in Fig.6. For e.g. Microsoft's Kinect, a system for low down-cost motion capture to the masses [9-13].

Algorithm For Real-Time Motion Capture

Most popular algorithms used in motion capture employ Kalman filtering [21]. Because of different motion capture systems must meet different demands, more and more optimal and advanced Kalman filtering algorithms are put forward.

In [21], Yun and Bachmann present an extended Kalman filter designed for real-time estimation of human limb segments orientation. The extended Kalman filter uses the Quest algorithm as a preprocessing procedure to compute quaternion input for the filter. Then, data from small inertial and magnetic sensor modules containing triaxial angular rate sensors, accelerometers, and magnetometers, are processed. This new algorithm shows the feasibility of using inertial and magnetic sensor modules for real-time human body motion tracking.

Particle filtering is another robust approach for motion tracking where computation cost is heavy in a high dimensional pose space. Using a number of heuristics, Jauregui et al. demonstrate particle filtering improves the robustness of motion capture [22]. With semi-deterministic particle prediction based on local optimization, the system deterministically resamples the probability distribution for a more efficient selection of particles. From the perspective of achieving motion capture in real time, measurement steps are parallelized on graphics processing units (GPUs). Their experiment shows that their approach is capable of robust real-time 3-D motion capture using a consumer-grade computer and a webcam.

For more optimal and advanced algorithms of interest for real-time motion capture systems, the readers are referred to [23][24][25][26][27][28].

Kalman filtering algorithm

Kalman filtering is an algorithm that Provides estimates of Some unknown variables given the measurements observed Over time. Kalman filters have been demonstrating its usefulness in various applications. Kalman filters have relatively simple form and require small Computational power.

Algorithm steps

Step 1: Initialize System State.
Step 2: Reinitialize System State.
Step 3: Predict System State Estimate.
Step 4: Compute the Kalman Gain.
Step 5: Estimate System State and System State Error Covariance Matrix.

Kalman filter algorithm overview

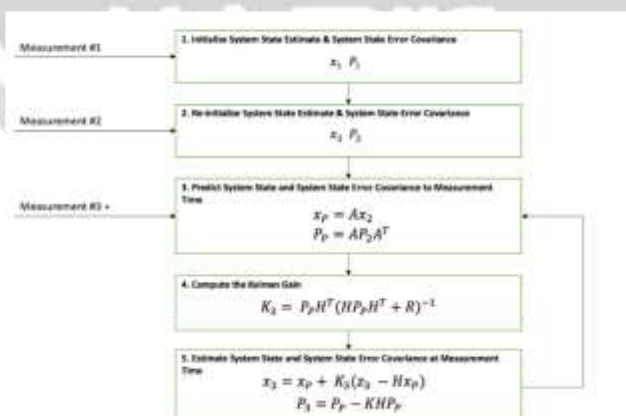

Fig.5 : Kalman Filter Algorithm Steps

The process diagram above shows the Kalman Filter algorithm step by step. I know those equations are intimidating but I assure you this will all make sense by the time you finish reading this article. Let's look at this process one step at a time. For your reference, here is a table of definitions that will be referred to throughout.

X	State variable	n x 1 column vector	Output
P	State covariance matrix	n x n	Output
Z	Measurement	m x 1 column vector	Input
A	State transition matrix	n x n matrix	System model
H	State-to-measurement matrix	m x n matrix	System model
R	Measurement covariance matrix	m x m matrix	Input
Q	Process noise covariance matrix	n x n matrix	System model
K	Kalman Gain	n x m	Internal

Kalman Filter Algorithm Reference Terms

Step 1: Initialize System State.

These equations show the input and output values for this Kalman Filter after receiving the first measurement.

$$z_1 = \begin{bmatrix} x_{m_1} \\ y_{m_1} \end{bmatrix} \qquad R_1 = \begin{bmatrix} \sigma_{x_m}^2 & \sigma_{xy_m} \\ \sigma_{xy_m} & \sigma_{y_m}^2 \end{bmatrix} \qquad t_1 = t_{m_1}$$

$$x_1 = \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} x_{m_1} \\ y_{m_1} \\ 0 \\ 0 \end{bmatrix} \qquad T_1 = t_{m_1}$$

$$P_1 = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} & \sigma_{x\dot{x}} & \sigma_{x\dot{y}} \\ \sigma_{xy} & \sigma_y^2 & \sigma_{y\dot{x}} & \sigma_{y\dot{y}} \\ \sigma_{x\dot{x}} & \sigma_{y\dot{x}} & \sigma_{\dot{x}}^2 & \sigma_{\dot{x}\dot{y}} \\ \sigma_{x\dot{y}} & \sigma_{y\dot{y}} & \sigma_{\dot{x}\dot{y}} & \sigma_{\dot{y}}^2 \end{bmatrix} = \begin{bmatrix} \sigma_{x_m}^2 & \sigma_{xy_m} & 0 & 0 \\ \sigma_{xy_m} & \sigma_{y_m}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Step 2: Reinitialize System State.

The system state estimate is reinitialized because a velocity estimate needs a second position measurement for computation.

Velocity is estimated with a linear approximation. As you most likely recall from high school physics, velocity is equal to the distance traveled divided by the time it took to travel that distance.

The updated system state estimate will be the second measurement's position and the computed velocity. The updated system state error covariance will be the second measurement's position accuracy and an approximated velocity accuracy. Note that this velocity accuracy approximation is something that can be tuned and adjusted after running data through your filter. There are different ways of approximating these values so if this doesn't match your approximation, that's okay!

Reinitialize System State in Equations

These equations show the input and output values for this Kalman Filter after receiving the second measurement. Note the velocity variance terms in the state covariance matrix. These values are being set to 104. In other words, this value indicates a large uncertainty for the velocity state values. In this example, the velocity units are meters per second.

$$z_2 = \begin{bmatrix} x_{m_2} \\ y_{m_2} \end{bmatrix} \qquad R_2 = \begin{bmatrix} \sigma_{x_m}^2 & \sigma_{xy_m} \\ \sigma_{xy_m} & \sigma_{y_m}^2 \end{bmatrix} \qquad t_2 = t_{m_2}$$

$$x_2 = \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} x_{m_2} \\ y_{m_2} \\ \frac{x_{m_2} - x_{m_1}}{\Delta T} \\ \frac{y_{m_2} - y_{m_1}}{\Delta T} \end{bmatrix} \qquad T_2 = t_{m_2} \quad \Delta T = T_2 - T_1$$

$$P_2 = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} & \sigma_{x\dot{x}} & \sigma_{x\dot{y}} \\ \sigma_{xy} & \sigma_y^2 & \sigma_{y\dot{x}} & \sigma_{y\dot{y}} \\ \sigma_{x\dot{x}} & \sigma_{y\dot{x}} & \sigma_{\dot{x}}^2 & \sigma_{\dot{x}\dot{y}} \\ \sigma_{x\dot{y}} & \sigma_{y\dot{y}} & \sigma_{\dot{x}\dot{y}} & \sigma_{\dot{y}}^2 \end{bmatrix} = \begin{bmatrix} \sigma_{x_m}^2 & \sigma_{xy_m} & 0 & 0 \\ \sigma_{xy_m} & \sigma_{y_m}^2 & 0 & 0 \\ 0 & 0 & 10^4 & 0 \\ 0 & 0 & 0 & 10^4 \end{bmatrix}$$

Step 3: Predict System State Estimate

When the third measurement is received, the system state estimate is propagated forward to time align it with the measurement. This alignment is done so that the measurement and state estimate can be combined.

xp = Axk-1	Eqn. 3-1
Pp = APk-1AT + Q	Eqn. 3-2

The system model is used to perform this prediction. In this example, a constant velocity linear motion model is used to approximate the objects position change over a time interval. Note that a constant velocity model does assume zero acceleration. Remember this because it will resurface later.

The constant velocity linear motion model is something you may also remember from your high school physics class. The equation states that the position of an object is equal to its initial position plus its displacement over a specified time period assuming a constant velocity.

A state transition matrix represents these equations. This matrix is used to propagate the state estimate and state error covariance matrix appropriately. You may be wondering why the state error covariance matrix is propagated.

The reason for this is because when a state estimate is propagated in time, the uncertainty about its state at this future time step is inherently uncertain, so the error covariance grows.

$$z_3 = \begin{bmatrix} x_{m_3} \\ y_{m_3} \end{bmatrix} \quad R_2 = \begin{bmatrix} \sigma_{x_m}^2 & \sigma_{xy_m} \\ \sigma_{xy_m} & \sigma_{y_m}^2 \end{bmatrix} \quad t_3 = t_{m_3}$$

$$T_3 = t_{m_3} \quad \Delta T = T_3 - T_2$$

$$A = \begin{bmatrix} 1 & 0 & \Delta T & 0 \\ 0 & 1 & 0 & \Delta T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad Q = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 25 & 0 \\ 0 & 0 & 0 & 25 \end{bmatrix}$$

$$x_p = Ax_2 = \begin{bmatrix} x + \dot{x}\Delta T \\ y + \dot{y}\Delta T \\ \dot{x} \\ \dot{y} \end{bmatrix}$$

$$P_p = AP_2A^T + Q = \begin{bmatrix} \sigma_x^2 + 10000 \cdot \Delta T^2 + 10 & \sigma_{xy} & 10000 \cdot \Delta T & 0 \\ \sigma_{xy} & \sigma_y^2 + 10000 \cdot \Delta T^2 + 10 & 0 & 10000 \cdot \Delta T \\ 10000 \cdot \Delta T & 0 & 10025 & 0 \\ 0 & 10000 \cdot \Delta T & 0 & 10025 \end{bmatrix}$$

Step 4: Compute the Kalman Gain.

The Kalman Filter computes a Kalman Gain for each new measurement that determines how much the input measurement will influence the system state estimate. In other words, when a really noisy measurement comes in to update the system state, the Kalman Gain will trust its current state estimate more than this new inaccurate information.

This concept is the root of the Kalman Filter algorithm and why it works. It can recognize how to properly weight its current estimate and the new measurement information to form an optimal estimate.

K = PPHT (HPPHT + R)-1	Eqn. 4-1

Step 5: Estimate System State and System State Error Covariance Matrix.

The Kalman Filter uses the Kalman Gain to estimate the system state and error covariance matrix for the time of the input measurement. After the Kalman Gain is computed, it is used to weight the measurement appropriately in two computations.

The first computation is the new system state estimate. The second computation is the system state error covariance.

xk = xp + K(zk – Hxp)	Eqn. 5-1
Pk = PP – KHPP	Eqn. 5-2

Kalman Filter Estimation Equations

The state estimate computed above is the only state history the Kalman Filter retains. As a result, Kalman Filters can be implemented on machines with low memory restrictions.

Advantages of motion capture

- In entertainment applications this can reduce the costs of keyframe-based animation.
- Complex movement and realistic physical interactions such as secondary motions, weight and exchange of forces can be easily recreated in a physically accurate manner.
- Potential for free software and third party solutions reducing its costs.

Disadvantages of motion capture

- Specific hardware and special software programs are required to obtain and process the data. The initial results are limited to what can be performed within the capture volume without extra editing of the data.
- Movement that does not follow the laws of physics cannot be captured.
- If the computer model has different proportions from the capture subject, artifacts may occur. For example, if a cartoon character has large, oversized hands, these may intersect the character's body if the human performer is not careful with their physical motion.

Applications of motion capture

- Advertising
- Entertainment
- Health
- Sports
- Robotics

The motion capture technology allow us to capture the real world movement which we can use later. The motion capture technology reduce the efforts of animation and provide facility to make smooth and accurate animation. The capture data form the motion capture technology can be use to teach our machine for taking physical action. The most of robotics training can be acquire by motion capture technology such as digital image processing and train to the hardware robots. Improvement in optical Motion capture system i.e. markup techniques has to be improve. Transference of motion captured animations between models such as applying animation of normal human cannot be transferred to giant as physical law they moves in very different manners. Improvement in actual tracking techniques: manipulating of the occlusion from capture data is make the animation worse. Improvement in tracking tech can eliminate this problem.

Conclusion:

This paper provides a detailed overview of motion capture system (Mocap) and their background. from a video stream is a growing and challenging area. Motion capture system (Mocap) is widely used in today's realistic world including sports, entertainment, video gaming and many more. In this paper, firstly we provide an overview of different types of Mocap on the basis of Marker-based Mocap and Markerless

Mocap along with their merits and demerits. Secondly, we discuss about the Applications of Mocap and algorithms for real-time motion capture.

REFERENCES

[1] M. Gliecher, "Animation from Observation: Motion Capture and Motion Editing", ACM SG Graph Computer graphics, Vol 33 Issue 4, PP 51-54, Nov 2000.

[2] Z. Manyu, "Application of Performance Motion Capture Technology in Film and Telivision Performance Animation", Atlantis Press, Paris, France, 2013.

[3] F.Kyle, "The Use of Motion Capture and 2D Animation in the Making of "If a First...",Department of Professional Media Management and Media Studies, pp. 268-269, 2012 Motion Capture Fundamentals. Available on:https://paginas.fe.up.pt/~prodei/dsie12/papers/paper_ 7.pdf [accessed on 27-Sep2018]

[4] S.li, M Okuda, S. Takahashi, "Embedded key- Frame Extraction for CG Animation by Frame Decimation", IEEE International Conference on Multimedia and Expo, pp 1243 - 1256, 2005.

[5] F. Lamberti, G. Paravati, V. Gatteschi, "Virtual Character Animation Based on Affordable Motion Capture and Reconfiguration Tangible Interfaces", IEEE Transactions on Visualization and Computer Graphics Vol. 24, Issue: 5, May 1 2018.

[6] J. Pan and J. J. Zhang, "Sketch-based Skeleton- Driven 2D animation and motion capture", Lecture Notes in Computer Science book series: Transactions on Edutainment VI, vol. 6758, pp 164-181, 2011.

[7] S. Yabukami, H. Kikuchi, M. Yamaguchi, "Motion capture system of magnetic markers using three-axial magnetic field sensor". IEEE Transactions on Magnetics, Vol. 36, Issue: 5, pp: 3646-3648. 2000

[8] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, "Real-Time Human Pose Recognition in Parts from Single Depth Images", Computer Vision and Patter Recognition, 2011.

[9] A. J. Davison, J. Deutscher, I. D. Reid. "Markerless motion capture of complex full-body movement for character animation", Proceedings of the Eurographic workshop on computer animation and simu lation, 2001.

[10] C. Chu, O. C. Jenkins, "Markerless Kinematic Model and Motion Capture from Volume Sequen ces". Proc. of IEEE Computer Vision and Pattern Recognition, Vol. 2, pp. 475-482. 2003.

[11] B. Rosenhahn, T. Brox, H. Seidel, "Scaled Motion Dynamics for Markerless Motion Capture". IEEE Conference on Computer Vision and Pattern Recognition, 2007.

[12] Faceshift Markeless motion capture. Available on:https://www.google.co.in/search?q=Markerless+based+motion+capture&source=lnms&tbm=isch&sa=X&ved=0ahUKEwis77Kh69fdAhVFu48KHR_zDtAQ_AUIDigB&biw=1209&bih=474&dpr=1.13#imgrc=j MsUgrVzy1nnYM [accessed on 26-sep-2018]

[13] Market and Market report. Available on:https://www.marketsandmarkets.com/PressReleases/ 3d-motioncapture-system.asp [accessed on 26-sep- 2018]

[14] B. Feng, H. Zhao,"The Research of Motion Techology Based on Inertial Measurement", IEEE11th International Conference on Dependable Autonomic and Secure Computing, 21-23 Dec. 2013.

[15] P. Rahima, J. K. Kearney, "OptimalCamera Placement for Motion Capture System", IEEE Transact ions on Visualization and Computer Graphics, Vol. 23 Issue: 3, 1 March 2017

[16] D. Dong, L. Wang,"Experimental Research on Human Body Motion Simulation Based on te Motion Capture Technology", International Conference on Digital Human Modeling, Vol. 4561 pp.42 – 47, 2007.

[17] G. Colombo, D. Regazzoni, "Markerless Motion Capture Integrated with Human Modeling for Virtual Ergonomics", Applications in Health Safety, Ergonomics, and Risk Management, DHM 2013, Vol. 8026 pp.314 – 323, 2013.

[18] Optical Systems (motion of face captured using reflectors). Available on: http://pirro.mur.at/traces.html. (accessed on 26-sep- 2018)

[19] Business insights.Available on: https://www.businessinsider.com/james-spader- ultron-without-visual-effects 2015-9?IR=T (accessed on 26-sep-2018)

[20] Yun, X.E., R. Bachmann. Design, implementation, and experimental results of a quaternion-based Kalman filter for human body motion tracking. Robotics, IEEE Transactions on, 2006. 22(6): p. 1216-1227.

[21] Jáuregui, D.A.G., P. Horain, M.K. Rajagopal, S.S.K. Karri. Real-time particle filtering with heuristics for 3D motion capture by monocular vision, in Multimedia Signal Processing (MMSP), IEEE International Workshop on. 2010.

[22] Hu, Y., W. Jin, F. Ni. An efficient wireless sensor network for realtime multiuser motion capture system, in Communication Technology (ICCT), IEEE 14th International Conference on. 2012.

[23] Jost, C., P. De Loor, L. Nedelec, E. Bevacqua, I. Stankovic. Real-time gesture recognition based on motion quality analysis, in Intelligent Technologies for Interactive Entertainment (INTETAIN), 7th International Conference on. 2015.

[24] Han, H., J. Kwon, J. Lee, R. Destenay, B.-J. You. Real-time optimization for the high-fidelity of human motion imitation, in Ubiquitous Robots and Ambient Intelligence (URAI), 11th International Conference on. 2014.