Journal of Theoretical and Applied Information Technology
10th March 2013. Vol. 49 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

REAL TIME SKELETAL ANIMATION WITH DUAL QUATERNION

[1]XIANG FENG, [2]WANGGEN WAN

[1]Master Student, College of Communication and Information Engineering, Shanghai University, Shanghai, China

[2]Prof., Institute of Smart City, Shanghai University, Shanghai, China

E-mail：[1]fengxiang0727@126.com , [2]wanwg@staff.shu.edu.cn

ABSTRACT

Though Combination of Quaternions and matrix has been a popular tool in skeletal animation for more than 20 years, classical quaternions are restricted to the representation of rotations. In skeletal animation and many other applications of 3D computer graphics, we actually deal with rigid transformation including both rotation and translation. Dual quaternions represent rigid transformations nearly in the same way as quaternions represent rotations. In fact, Algorithms based on dual quaternions show better properties than those based on quaternions, including increased computational efficiency, reduced overhead and coordinate invariance. In this paper we show how to generalize established techniques for quaternions to dual quaternions to include all rigid transformations, and implement real-time character animation with dual quaternions under the platform of OpenGL in order to demonstrate the superiority and effectiveness of dual quaternions.

Keywords: *Skeletal animation, Dual quaternion, Rigid transformation, Real time*

1. INTRODUCTION

Animation of characters in 3D can be done in several ways. When the animations become more complex it poses problems in matter of memory usage we usually employ skeletal animation system. We build a skeleton and joints inside the meshes we wish to animate, animate the skeleton within instead of animating the mesh itself.

Every vertex is tied to at least one joint through an influence. An influence stores the vertex, the joint index and a weight which specify how much influence the joint has over the vertex [1]. A vertex can have several influences but only one for each joint. The sum of all the weights in a vertex' influences should always be one. This is process is often called skinning because the mesh can be seen as a virtual skin over the skeleton.

It is well known that quaternions are an advantageous representation of 3D rotations, in many aspects better than rotation matrices [2]. But rigid objects not only rotate, but also translate. A rotation composed with translation is called a rigid transformation, and any displacement of a rigid object in 3D space can be described by a rigid transformation. In this paper, we advocate that dual quaternions are a better representation of rigid transformations than those treating rotation and translation components independently. We combine dual quaternions to implement real time skeletal animation.

2. RELATED WORK

The dual quaternion has been around since 1882 but has gained less attention compared to quaternions alone [3]. Comparable to quaternions the dual quaternion has had a taboo associated with them, whereby students avoid quaternion and hence dual-quaternions. While the research community in robotics has started to adopt dual-quaternions in recent years, the research community in computer graphics such as character animation has not embraced them as whole-heartedly.

Kuang presented a strategy for creating real time animation of clothed body movement [4]. Vasilakis discussed skeleton based rigid skinning for character animation [5]. Selig examined the problem of solving the equations of motion in real-time, put forward how dual quaternions gave a very neat and succinct way to represent rigid-body transformations, and addressed the key problem in computer games [6].

Journal of Theoretical and Applied Information Technology
10th March 2013. Vol. 49 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Schilling used dual quaternion with a mean of multiple computational model to model bodies [7]. Pham used Jacobian matrix in the dual quaternion space to solve linked chain inverse kinematic problems [8]. Yang used dual quaternions to calculate the relative orientation [9].

3. SKELETAL ANIMATION

In the skeleton, every joint has one or more degrees of freedom (DOFS), which define its possible range of motion, and a detailed character may have more than a hundred DOFs in the entire skeleton [10]. Specifying values for these DOFs poses the skeleton, and changing these values over time results in movement, and is the essence of the animation process. A simple character skeleton [15]and the hierarchical topological graph of skeleton joints [16]is shown in Figure 1 and Figure 2 respectively.

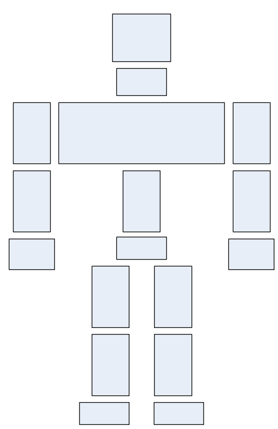


Figure 1: A simple character skeleton

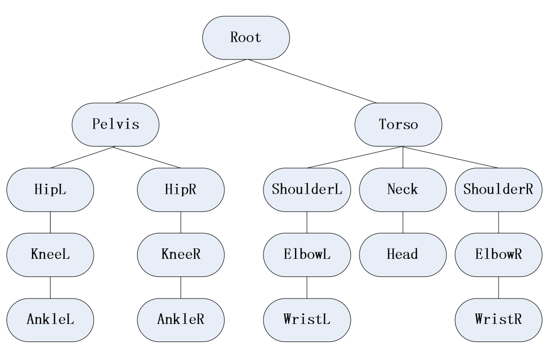


Figure 2: Hierarchical topological graph of skeleton joints

Given a set of specified DOF values, a joint local matrix can be constructed for each joint. These matrices define the position and orientation of each joint relative to the joint above it in the tree hierarchy. The local matrices can then be used to compute the world space matrices for all of the joints using the process of forward kinematics. These world space matrices are what ultimately place the virtual character into the world, and can be used for skinning, rendering, collision detection, or other purposes. A vector is transformed by a matrix in the following manner:

$$V' = V \cdot M \tag{1}$$

Where V' is the resulting transformed vector. If V is a vertex in an object's local coordinate system and M is a matrix placing the object in world space, then V' will be the vertex's location in world space. The inverse of this transformation is written as:

$$V = V' \cdot M^{-1} \tag{2}$$

Where M^{-1} is the matrix inverse of M. If M is a matrix that transforms from local to world space, then M^{-1} will transform from world space to local space.

4. DUAL QUATERNION

4.1 Quaternions and Dual Quaternion

Quaternions have been a popular tool in 3D computer graphics for more than 20 years[13], they are four terms real numbers $(q_r\ q_x\ q_y\ q_z)$ which include a three-term vector with components q_x q_y and q_z. Quaternions are usually represented in the form

$$Q = q_r + q_x\vec{i} + q_y\vec{j} + q_z\vec{k} = q_r + \vec{q} \tag{3}$$

Where q_r and $\vec{q}$ are the real and vector parts, respectively, and $\vec{i}$, $\vec{j}$ and $\vec{k}$ are the unit vectors associated with the axes of a Cartesian coordinate system. A dual quaternion can be used to define a rigid body rotation of an angle θ about an axis $\vec{u}$ through the origin

$$Q = \cos\frac{\theta}{2} + \vec{u}\sin\frac{\theta}{2} \tag{4}$$

Classical quaternions are restricted to the representation of rotations, whereas in graphical applications we typically work with rotation composed with translation, i.e. rigid transformations. Dual quaternions are mathematical entities[14] whose four components are dual numbers，they can be expressed as follows:

Journal of Theoretical and Applied Information Technology
10th March 2013. Vol. 49 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$Q = Q + \varepsilon Q_0 \qquad (5)$$

Where Q and Q_0 are both quaternions, and ε is dual unit. Dual quaternion can formulate a problem more concisely, solve it more rapidly and in fewer steps, present the result more plainly to others, be put into practice with fewer lines of code and debugged effortlessly. Dual-quaternion has a unified representation of translation and rotation as follows:

$$Q = r\,;\, Q_0 = \frac{1}{2} t \cdot r \qquad (6)$$

Where r is a unit quaternion representing the rotation and t is the quaternion describing the translation represented by the vector t, as we can see from Figure 3.

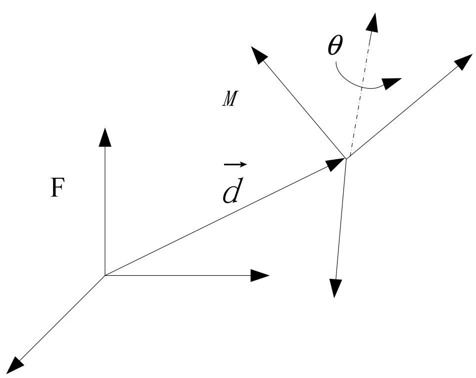


Figure 3: Rigid transformation with dual quaternions

Dual quaternions represent rigid transformations in the same way as classical quaternions represent rotations. The dual quaternion blending algorithms [11] could be applied in motion blending, motion analysis, spatial key framing, computer vision, graphics hardware.

4.2 Dual quaternion Linear Blending

After we convert our skinning transformations to unit dual quaternions $\hat{q}_1, \ldots, \hat{q}_n$, what we need to do next is to compute a blended unit dual quaternion $\hat{q}$ with the given convex weights

$$w = (w_1, \cdots, w_n) \qquad (7)$$

We achieve this by taking their linear combination followed by the normalization to get a unit dual quaternion, which is called Dual quaternion Linear Blending (DLB)[17]:

$$DLB(w; \hat{q}_1, \ldots, \hat{q}_n) = \frac{w_1 \hat{q}_1 + \ldots + w_1 \hat{q}_n}{\| w_1 \hat{q}_1 + \ldots + w_1 \hat{q}_n \|} \qquad (8)$$

DLB has many excellent properties required in skinning. DLB computes a unit dual quaternion which can be subsequently converted to a rigid transformation matrix. As a result, DLB always returns a rigid transformation. In order to demonstrate the fact that DLB is coordinate-invariant, we verify that for any unit dual quaternion $\hat{q}$, the following formula is true:

$$DLB(w; \hat{r}\hat{q}_1\hat{r}^*, \ldots, \hat{r}\hat{q}_n\hat{r}^*) = \hat{r} DLB(w; \hat{q}_1, \ldots, \hat{q}_n)\hat{r}^* \qquad (9)$$

In fact, it breaks down to verifying two similar properties which are called left and right invariance. Left invariance requires what follows:

$$DLB(w; \hat{r}\hat{q}_1, \ldots, \hat{r}\hat{q}_n) = \hat{r} DLB(w; \hat{q}_1, \ldots, \hat{q}_n) \qquad (10)$$

Right invariance requires what follows:

$$DLB(w; \hat{q}_1\hat{r}, \ldots, \hat{q}_n\hat{r}) = DLB(w; \hat{q}_1, \ldots, \hat{q}_n)\hat{r} \qquad (11)$$

It should be clear that coordinate-invariance includes both left invariance and right invariance. To prove left invariance, we apply the distributive property of dual quaternions and the multiplicative property of the norm, according to our assumption:

$$\| \hat{r} \| = 1 \qquad (12)$$

$$DLB(w; \hat{r}\hat{q}_1, \ldots, \hat{r}\hat{q}_n) = \frac{w_1 \hat{q}_1 + \ldots + w_n \hat{q}_n}{\| w_1 \hat{q}_1 + \ldots + w_1 \hat{q}_n \|} =$$

$$\hat{r} \frac{w_1 \hat{q}_1 + \ldots + w_n \hat{q}_n}{\| \hat{r} \| \, \| w_1 \hat{q}_1 + \ldots + w_n \hat{q}_n \|} = \hat{r} DLB(w; \hat{q}_1, \ldots, \hat{q}_n) \qquad (13)$$

Demonstration of the right invariance proceeds along the same lines and therefore DLB is indeed coordinate-invariant. When DLB is applied to two rigid transformations, it interpolates them along the shortest trajectory. Let therefore p, $\hat{q}$ be two unit dual quaternions. We denote their blending as $DLB(t; p, \hat{q})$. Due to the left invariance of DLB we can write as follows:

$$DLB(t; p, \hat{q}) = p p^* DLB(t; p, \hat{q}) = p\, DLB(t; 1, p^* \hat{q}) \qquad (14)$$

It is therefore sufficient to show that the path between 1 and $p^* \hat{q}$ will be the shortest one which is given by the screw corresponding to $p^* \hat{q}$. Since p, $\hat{q}$ are unit dual quaternions, so is $p^* \hat{q}$.

Journal of Theoretical and Applied Information Technology
10th March 2013. Vol. 49 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

There exists a unit dual quaternion with zero scalar part $\hat{n}$ and a dual scalar α such that

$$\overset{*}{p}\hat{q} = \cos\frac{\alpha}{2} + \hat{n}\sin\frac{\alpha}{2} \quad (15)$$

Therefore, $DLB(t; p, \hat{q})$ can be re-written as follows:

$$DLB(t;1,\overset{*}{p}\hat{q}) = \frac{1-t+t\overset{*}{p}\hat{q}}{\|1-t+t\overset{*}{p}\hat{q}\|} = \frac{1-t+\cos(\frac{\alpha}{2})+\hat{n}t\sin(\frac{\alpha}{2})}{\|1-t+t\overset{*}{p}\hat{q}\|} \quad (16)$$

It means that the screw axis of $DLB(t; p, \hat{q})$ is the same for all $t \in [0,1]$ and is given by $\hat{n}$. The only thing that varies is amount of translation and the angle of rotation and, encoded in

$$\frac{1-t+t\cos(\frac{\alpha}{2})}{\|1-t+t\overset{*}{p}\hat{q}\|} \quad (17)$$

$$\frac{t\sin(\frac{\alpha}{2})}{\|1-t+t\overset{*}{p}\hat{q}\|} \quad (18)$$

In other words, $DLB(t;1,\overset{*}{p}\hat{q})$ produces a shortest path screw motion. This screw motion does not have a constant speed, which is similar to linear interpolation of regular quaternions. However, the velocity of this motion is actually not far from constant, which explains why this issue does not present any visible drawbacks in skinning. The procedure of Simple Dual quaternion Linear Blending is shown in Table 1.

Table 1 Simple Dual quaternion Linear Blending

Algorithm 1 sDLB: Simple Dual quaternion Linear Blending

procedure $SDLB(\hat{q}_1, \hat{q}_2, t)$

$$SDLB(t; \hat{q}_1, \hat{q}_2) = \frac{(1-t)\hat{q}_1 + t\hat{q}_2}{\|(1-t)\hat{q}_1 + t\hat{q}_2\|}$$

end procedure

4.3 Screw linear interpolation

This is a generalization of the well-known Spherical Linear Interpolation (SLERP) [12]scheme. Let denote by $\hat{q}_1$ and $\hat{q}_1$ two dual quaternions expressing the initial and final pose of a rigid body, respectively.

The ScLERP function (Screw Linear Interpolation) [18] is defined as follows

$$ScLERRP(t; \hat{q}_1, \hat{q}_2) = \hat{q}_1 * (\hat{q}_1 * \hat{q}_2)^t \ , t \in [0,1] \quad (19)$$

Since $\hat{q}_1^{-1} * \hat{q}_2$ represents the finite screw motion between the initial and final pose of the rigid body, the product

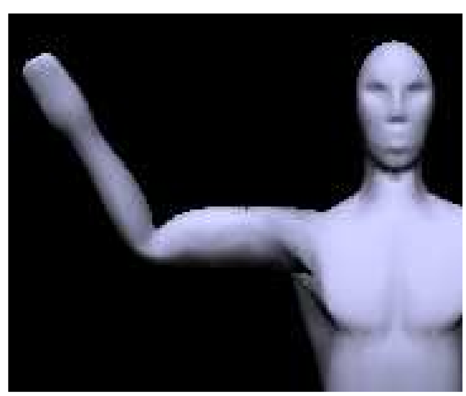
$$(\hat{q}_1^{-1} * \hat{q}_2)^t = \cos(t\frac{\hat{\theta}}{2}) + \sin(t\frac{\hat{\theta}}{2})\hat{u} \quad (20)$$

defines a screw motion of a dual angle tbθ along the screw axis.

5. IMPLEMENTATION

The dual-quaternion unifies the translation and rotation into a single state variable. This single state variable offers a robust, unambiguous, computationally efficient way of representing rigid transform. We combine skeletal animation and dual quaternions to implement character animation with OpenGL. We demonstrate the real time result in Figure 4 and Figure 5.

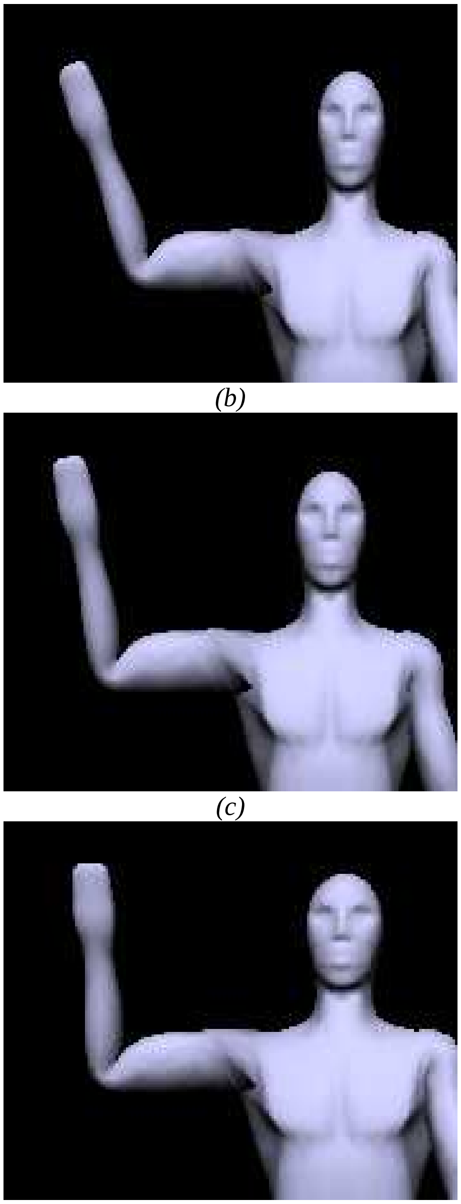
Dual quaternion model is an accurate, computationally efficient, robust, and flexible method of representing rigid transforms, which should not be overlooked. It enables the creation of more elegant and clearer computer programs that are easier to work with and control when we implement pre-programmed dual quaternion modules including multiplication and normalization.



(a)

Journal of Theoretical and Applied Information Technology
10th March 2013. Vol. 49 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195



(b)

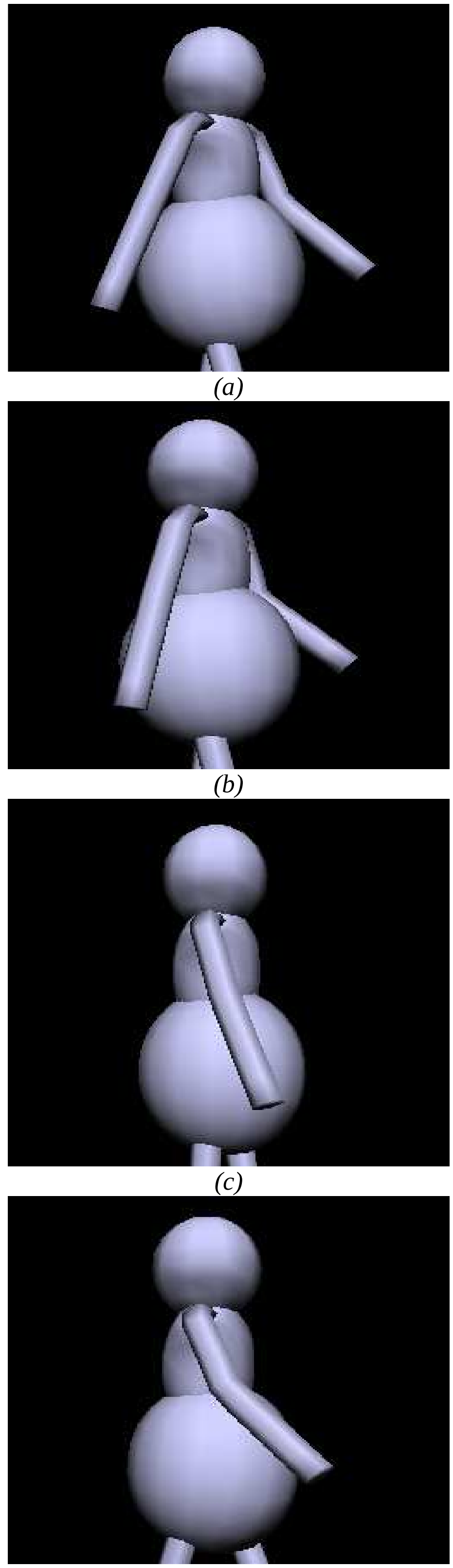
(c)

(d)

Figure 4: Character Animation With Dual Quaternions, Frame By Frame, The Subfigures In (A) (B) (C) (D) Show Four Different Frames.

The Computational Cost Of Combining Matrices And Dual-Quaternions:

Matrix4x4 : 64mult + 48adds
Matrix4x3 : 48mult + 32adds
Dual Quaternion : 42mult + 38adds



(a)

(b)

(c)

(d)

Figure 5: Cartoon Girl Animation With Dual Quaternions, Frame By Frame, The Subfigures In (A) (B) (C) (D) Show Four Different Frames.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

6. CONCLUSION

Skeletons in character animation are typically built from a hierarchy of rigid bones connected by articulated joints. Rigid transformation blending of bones based on dual quaternions exhibit advantageous properties, and fast execution time. In this paper, we introduce dual quaternion and take the advantage of it. We make use of dual quaternion to represent the translation and rotation, and have implemented real-time character animation with OpenGL.

In many ways, a digital character's skeleton is analogous to the skeleton of a real animal [19]. Real world animals with true bones are called vertebrates, which includes humans, mammals, reptiles, fish, and birds. The use of a virtual skeleton to animate these creatures should make perfect sense while digital bones don't necessarily have to correspond to actual bones [20]. In addition to animating rigid movement, they can be used to animate facial expressions, soft tissues and mechanical parts, such as fat muscles, wheels, and even clothing. Skeletons can be used to animate, cartoon characters, aliens, robots, plants, humans, vehicles, furniture, and others.

ACKNOWLEDGEMENTS

This research is supported by the National Natura l Science Foundation of China (No.61103076).

REFERENCES:

[1] T.C.Xu , Mo Chen , Ming Xie and Enhua Wu. "A Skinning Method in Real-time Skeletal Character Animation". The International Journal of Virtual Reality, 2011, 10(3):25-31.

[2] J. McDonald, "Teaching Quaternions is not Complex," Computer Graphics Forum, vol. 29, no. 8, pp. 2447-2455, Dec. 2010.

[3] Ben Kenwright, "A Beginners Guide to Dual-Quaternions: What They Are, How They Work, and How to Use Them for 3D Character Hierarchies", The 20th International Conference on Computer Graphics, Visualization and Computer Vision, WSCG 2012 Communication Proceedings,pp.1-13.

[4] Y. Kuang, A. Mao, G. Li, and Y. Xiong, "A strategy of real-time animation of clothed body movement," in Multimedia Technology (ICMT), 2011 International Conference on, 2011, pp. 4793–4797.

[5] A. Vasilakis and I. Fudos, "Skeleton-based rigid skinning for character animation," in Proc. of the Fourth International Conference on Computer Graphics Theory and Applications, 2009, no. February, pp. 302–308.

[6] J. Selig, "Rational interpolation of rigid-body motions," Advances in the Theory of Control, Signals and Systems with Physical Modeling, pp. 213–224, 2011.

[7] M. Schilling, "Universally manipulable body models — dual quaternion representations in layered and dynamic MMCs," Autonomous Robots, 2011.

[8] H. L. Pham, V. Perdereau, B. V. Adorno, and P. Fraisse, "Position and orientation control of robot manipulators using dual quaternion feedback," in Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on, 2010, pp. 658–663.

[9] Y. Lin, H. Wang, and Y. Chiang, "Estimation of relative orientation using dual quaternion," System Science and, no. 2, pp. 413-416, 2010.

[10] Information on http://graphics.ucsd.edu/courses/cse169_w05.

[11] Ilie, M.D. Negrescu, C. Stanomir, D. "An efficient parametric model for real-time 3D tongue skeletal animation".2012 9th International Conference on Communications (COMM). pp. 129-132, 2012.

[12] CH Chen, IC Lin, MH Tsai, PH Lu. "Lattice-Based Skinning and Deformation for Real-Time Skeleton-Driven Animation". 2011 12th International Conference on Computer-Aided Design and Computer Graphics (CAD/Graphics), pp.306-312.2011.

[13] Yongmin Zhong. "A Quaternion-Based Method for SINS/SAR Integrated Navigation System". IEEE Transactions on Aerospace and Electronic Systems, vol.46, pp514-524. 2012.

[14] Jiang Feng, Wang Hui-nan, Huang Chao Song. "Algorithm for Relative Position and Attitude of Formation Flying Satellites Based on Dual Quaternion". Chinese Space Science and Technology , 2012,V32(3): 20-26.

[15] F. Z. Ivo and H. Ivo, "Spherical skinning with dual quaternions and QTangents," ACM SIGGRAPH 2011 Talks, vol. 27, p. 4503, 2011.

[16] Ladislav K, Steven C, Jiri,Z, Carol O'Sullivan. "Geometric skinning with approximate dual quaternion blending". ACM Transactions on Graphics, Volume 27 Issue 4, October 2008.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[17] A Perez Gracia. "Synthesis of spatial RPRP closed linkages for a given screw system". Journal of mechanisms and robotics. vol. 3, no. 2. 2011.

[18] Ettore Pennestrì , Pier Paolo Valentini. "Dual quaternions as a tool for rigid body motion analysis: a tutorial with an application to biomechanics". Archive of Mechanical Engineering, Volume LVII, Issue 2, Pages 187–205, October 2010

[19] M. C. Marco Antonio, González, M. C. Martha Rosa Cordero, Jorge Benjamín Silva González. "Web Virtual Ball Game Pok-Ta-Pok". Journal of Theoretical and Applied Information Technology, pp8 – 13,Vol 34. No. 1,2011.

[20] Mr. R.M.Noorullah and Dr.A.Damodaram. "Innovative Thinning And Gradient Algorithm For Edge Field And Categorization Skeleton Analysis Of Binary And Grey Tone Images". Journal of Theoretical and Applied Information Technology, pp 75– 80, Vol 5,No1, 2009.