Object Motion Rendering with the String-based Haptic Interface SPIDAR for Enhanced Video Viewing Experience

Anusha Jayasiri, Katsuhito Akahane, Makoto Sato
Precision and Intelligence Laboratory, Tokyo Institute of Technology
4259, Nagatsuta-cho, Midori-ku, Yokohama-shi, 226-8503, Japan

{anusha, kakahane }@hi.pi.titech.ac.jp, msato@pi.titech.ac.jp

***Abstract*—With the invention of immersive displays and high technology multimedia systems, there is an interest among viewers for more realistic and interactive experience with video media. Their viewing experience can be further enhanced by letting them to feel the motion of objects in a video through haptic interface, as it becomes an additional sensation over seeing and hearing it. The objective of this research is to use the string based haptic interface, SPIDAR to interact with the video and enable the viewers to feel the motion of objects in it, beyond passive seeing and hearing. We propose two methods for object motion rendering, one using a linear gain controller and another using a nonlinear gain controller. Furthermore, we evaluated those two methods with the participation of real users, with the objective of identifying the better method. We can conclude that the method using a nonlinear gain controller is more effective than the method using a linear gain controller for object motion rendering since it enables the user to get a continuous feeling of the movement of objects in the video. Feedbacks of real users further enable us to conclude that haptic motion rendering of a video sequence enhances the viewing experience of viewers.**

***Index Terms*— Video Processing, Multimedia applications, Virtual Reality, Haptic Interface, SPIDAR (Space Interface Device for Artificial Reality), Optical Flow, Haptic Motion Rendering**

I. Introduction

Nowadays, there is a growing interest among viewers for more realistic and interactive experience with video media. With the invention of high technology multimedia systems and immersive displays, significance of exploring new ways of interacting with video media has grown up. For example, in three-dimensional television (3D-TV), viewers can see objects in true dimensions and in their natural color, providing a natural viewing experience with advanced audio technologies. The Viewers` experience can be further enhanced if they can feel the movement of the objects in the video through haptic interface, as it is an additional sensation to seeing and hearing.

Today, there is a vast development and significant involvement of haptic interfaces in the world. Unlike traditional interfaces that provide visual and auditory information, haptic interfaces can generate mechanical signals, which stimulate human kinaesthetic and touch channels [1]. As a result it enables the users to touch and manipulate objects in the scene and can enhance human senses in a virtual world. Haptic Interfaces are being used in different kinds of application areas such as in training [2] [3], education [4], entertainment and 3D interaction [5]. However, the incorporation of haptic interface technology into video media is still in its infancy.

As Dinder et al [6] discusses, there are three types of haptic effects which are cooperating with haptic interaction of video media. Those are haptic structure, haptic texture and haptic motion. Haptic structure refers to the touching or getting the feeling of the geometry of an object in the video scene. Haptic texture refers to the rendering of surface properties such as roughness of various objects in the video scene. Haptic motion refers to the rendering of forces related to the moving objects in the scene. In this research we address the effect of haptic motion. Consequently, this research is an attempt to use haptic technology to interact with a video with the objective of enabling the viewers to feel the motion of objects in the video beyond passive seeing and hearing.

Videos are generally made up of sequences of individual images called frames. We identify feature points of each image frame and do the tracking of those points using basic computer vision algorithms. We calculate the motion of an object in the video by using the velocity information of feature points. The most interesting and the novel feature of this research is haptic motion rendering. For that, we propose a method by evaluating two candidate methods; one using a linear gain controller and another using a nonlinear gain controller. We use the haptic device SPIDAR, which is developed by the Sato laboratory of Tokyo Institute of Technology.

SPIDAR, which stands for `SPace Interface Device for Artificial Reality`, is a string-based haptic interface and can be used in various types of Virtual Reality applications ranging from desktop, workbench, human scale and networked environments. SPIDAR has three distinguishable features namely scalability, string based and transparency. Scalability means, the ability of SPIDAR to fit into different working spaces such as desktop, workbench, or human-scale with simple modifications on its structural layout. String based technology gives the user the ability to display position, orientation and force feedback information, providing an effective means of pointing and controlling in a virtual environment. Transparency is based on the string-based technology, because SPIDAR keeps the working space transparent without obscuring the visual display. The first proposal of SPIDAR was presented by Professor Makoto Sato in 1989 and until now different versions of SPIDAR systems are available from simple `pick` and `place` tasks to more complicated physical interactions in virtual worlds [7].

Some of new inventions of SPIDAR systems are namely SPIDAR-H, SPIDAR-G, SPIDAR-8, SPIDAR-mouse and SPIDAR-I. Among them SPIDAR-H and SPIDAR-8 are large scale devices and SPIDAR-G, SPIDAR-mouse and SPIDAR-I are desktop versions. SPIDAR-H can provide force feedback sensation to users within human scale virtual environments. SPIDAR-G works as a three dimensional interface device for 3D virtual environment interactions. SPIDAR-8 is a two-handed multi-finger version, which allows a user to use thumb, index, middle, and ring fingers of both left and right hands to manipulate virtual objects in a simulated virtual world. The user can perform the corporative work using both hands and perceive force feedback at eight fingertips, while manipulating the virtual objects. SPIDAR-mouse is suitable for interactions in 2D virtual environments and SPIDAR-I is an inner-string haptic interface, which is still in its development stage.

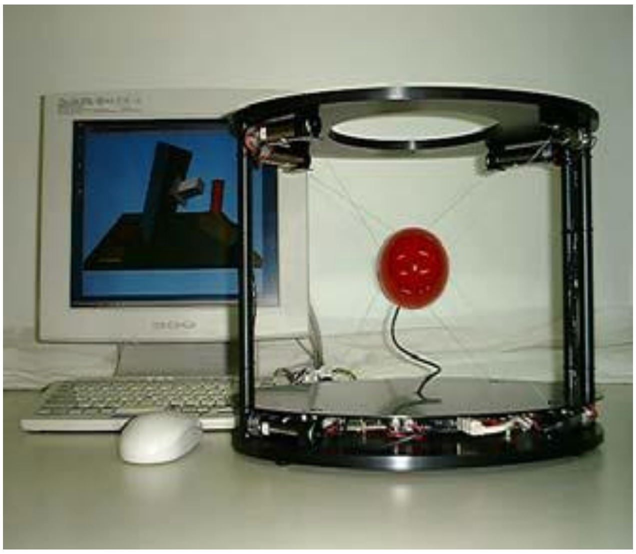

Fig. 1. SPIDAR-G haptic device

Even though the SPIDAR system is used in various types of virtual reality applications, it has not yet been used in the context of video media. From the above family of SPIDAR haptic interfaces, for this research we use the SPIDAR-G haptic interface, which is shown in Fig. 1. SPIDAR-G is a grip type, tension based, 6 degrees of freedom (3 degrees of freedom for translation, 3 degrees of freedom for rotation) and grasp enabled force-feedback device. This device has a grip and it is used to grasp objects in the virtual world. This grip is attached to 8 strings. Each string is connected to a motor and an encoder at one end and to the grip at the other end. The feedback force is determined by the tension of each string generated by the motor, which is transformed to the users hand through the grip. By connecting this device to a personal computer, it provides a high definition force feedback sensation to the user`s hand [8] [9].

When used in a context of a video, SPIDAR-G enables the viewer to feel the movement of the objects in the video through the forces related to the movement into his/her hand by grasping the grip as illustrated in Fig. 2.

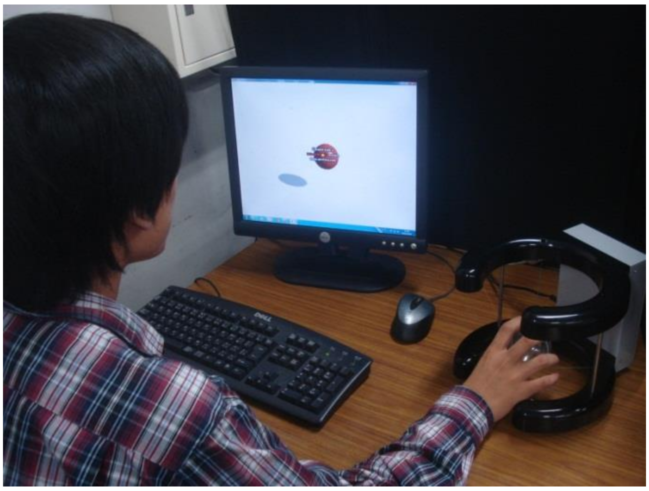

Fig. 2. Haptic interaction scenario

This paper is organized as follows. Section 2 presents related work in haptic rendering with video media and using SPIDAR for image haptization. Section 3 presents the methods adopted for feature point selection, feature point tracking, motion estimation and haptic motion rendering. We further elaborate on our proposed approach for haptic motion rendering using two methods; a linear gain controller and a nonlinear gain controller. Section 4 presents the results of the experimental evaluation using above two methods to identify the better method. It analyses the users’ feedbacks regarding the viewing experience of video with and without haptic feedback. Section 5 provides the concluding remarks and future work.

II. Related Work

Incorporation of haptic technology into video media is not adequately researched in the haptic rendering field. This section summarizes few of them.

Dinder et al [6] have introduced the concept of haptic motion for the first time and they discussed a method to compute haptic structure and motion signals for 2D video – plus-depth representation, which enables the viewer to navigate the scene and get the experience of the geometry of objects in the scene as well as the forces related to moving objects in the scene, using PHANToM haptic interface. They model the total force as the sum of static and dynamic forces. While the static force is related to the geometry, material and surface properties of an object, dynamic force relates to the relative motion between the object and haptic interaction point.

O`Modhrain et al [10] have discussed how haptic interaction can enhance and enrich the viewer's experience in broadcast content. They proposed a touch TV project with the use of Gravis Xterminator Force and remote control handset to generate haptic cues for cartoons and live sports broadcastings, which adds greater sense of immersion. They believe that the interactive nature of touch media has the potential to greatly enrich interactive TV by physically engaging the viewer in the programmed experience.

Cha et al [11] [12] [13] have proposed a touchable 3D video system, which provides haptic interaction with objects in a video scene. As a result it enables the viewers to actively touch a video scene through a PHANToM force feedback device. It enables to physically explore the video content and feel various haptic properties such as texture, height map and stiffness of the scene. They introduced Depth Image-Based Haptic Representation (DIBHR) method to add haptic surface properties of the video media.

Kim et al [14] have proposed a 3DTV system, which enables not only enjoying a high-quality 3D video in real time but experiencing various user-friendly interactions such as free viewpoint changing, composition of computer graphics and haptic display. They created a new representation of dynamic 3D scene, called 3D depth video, in which the viewer can touch the shape of it by wearing a haptic device using a haptic rendering algorithm.

Because of the distinguishing features of scalability, string based and transparency of the SPIDAR system, it`s interface is used in various types of virtual reality systems [7]. Even though SPIDAR has not been used for haptization in video media, recently SPIDAR-G haptic interface has been used in the context of images. Liu et al [15] have proposed a 2D image haptization system which provides the users with sense of touch on an image with local deformations using SPIDAR haptic interface. He further extends his research to 3D image haptization with local deformations by using depth representation of images [16].

III. Methodology

The proposed method for this research can be illustrated using a block diagram as in Fig. 3.

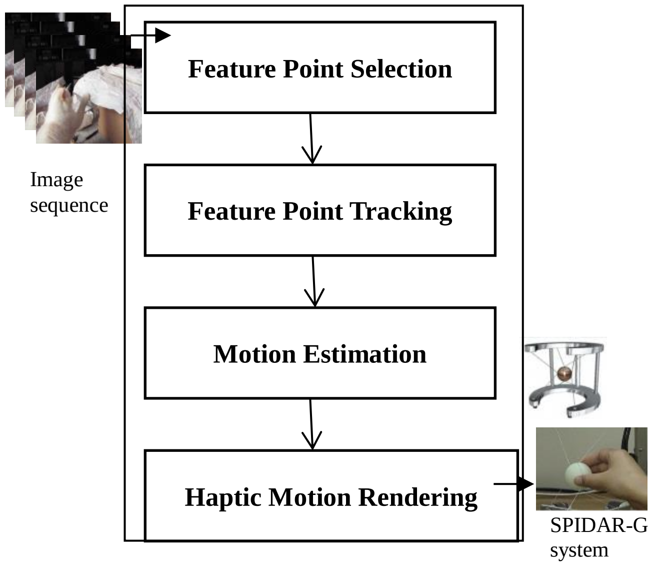

Fig. 3. Block diagram of the proposed approach

As shown in Fig. 3, the proposed approach in this research has four major parts namely feature point selection, feature point tracking, motion estimation and haptic motion rendering. The first step of this sequence of steps is feature point selection. In that step, it identifies good features of each image frame of a video. The next step, feature point tracking involves the tracking of those feature points from frame to frame. In the motion estimation, it calculates the motion of an object in the image frame by getting the average motion of each feature point throughout the video. In this research our main contribution is in the final step, i.e. haptic motion rendering. There, we intend to find a method to associate haptic signals with video media to generate motion feedback to users through the haptic interface SPIDAR. For that we propose two candidate methods with the intention of selecting the better one. The following sections broadly describe the above parts.

A. Feature Point Selection

Feature point selection is an important task of any computer vision and image processing application. Since feature point selection is the starting point of many computer vision algorithms, the performance of the subsequent algorithm as well as the overall performance of the process basically depends on it.

Feature point selection finds which points are good to track. For example corners or good textures may be good feature points of an image. There are various methods existing for feature selection such as Harris, Canny, Sobel etc. Among those methods Shi & Thomasi algorithm gives better results than the others [17]. Besides, this algorithm is more efficient in feature point detection and hence, the processing time of the overall process becomes less. Therefore we use Shi & Thomasi algorithm for feature point selection in the image sequence. This algorithm is based on the assumption that the brightness of a pixel does not change from frame to frame.

Feature points are chosen by first selecting a neighbourhood $\boldsymbol{N}$ of $n \times n$ pixels around each pixel in the image. Due to the image motion, the horizontal and vertical displacements of the point at (x, y), ∂x and ∂y, are calculated and the derivatives $\partial I/_{\partial x}$ and $\partial I/_{\partial y}$ are calculated with a Sobel operator for all pixels in the neighbourhood N [17]. For each pixel in the neighbourhood the eigenvalues λ is of matrix A, which represented by equation (1), are calculated.

$$A = \begin{bmatrix} \sum_{Neighbourhood}\left(\frac{\partial I}{\partial x}\right)^2 & \sum_{Neighbourhood}\left(\frac{\partial^2 I}{\partial x \partial y}\right) \\ \sum_{Neighbourhood}\left(\frac{\partial^2 I}{\partial x \partial y}\right) & \sum_{Neighbourhood}\left(\frac{\partial I}{\partial y}\right)^2 \end{bmatrix} \quad (1)$$

In case of a feature point, the matrix A should have two large eigenvalues. Therefore, the pixels as feature points with the large value of λ are then selected by thresholding. Based on the magnitude of the eigenvalues, the following inferences can be made [17].

1. If λ1 and λ2 are too small, the pixel (x, y) has no features of interest.
2. If λ1 is too small and λ2 has a large positive value, an edge is found.
3. If λ1 and λ2 have large positive values, a corner is found.

This method, it recognizes the corners as more stable for tracking, by using the condition [**min** $(\lambda_1, \lambda_2) > \lambda$**]**, where λ is a predefined threshold. Furthermore, this method accepts feature points only if two stronger feature points are distanced enough from each other.

B. Feature Point Tracking

Feature point tracking involves identifying above features reliably from frame to frame. These feature points are then used to measure the motion of the objects between two frames in an image sequence. In this research, we use the optical flow technique for feature point tracking [18].

Optical Flow is the distribution of apparent velocities of movement of brightness patterns in an image. Optical flow arises from relative motion of objects and the viewer [19]. The way an object moves when it is seen or followed in a video or sequence of images is known as optical flow [20]. There are two types of optical flow methods namely dense optical flow and sparse optical flow. In dense optical flow methods, it associates velocity with every pixel in an image. 'Horn-Schunck method' and 'Block matching method' are examples for this type of optical flow [18]. In practice, calculating dense optical flow is not easy because of the high computational cost. Alternatively, sparse optical flow techniques calculate velocities only on the points which have certain desirable properties. Suppose there is a pixel point ***(x, y, t)*** of an object in the image at time t with intensity I(x, y, t), which moves to (△x, △y, △t) in the next image frame.

According to the image brightness constraint equation in equation (2), the brightness of a pixel does not change as it moves from frame to frame.

$$I(x, y, t) = I(x+\Delta x, y+\Delta y, t+\Delta t) \quad (2)$$

Assuming the displacement between two consecutive frames to be small, using Taylor series for brightness can obtain,

$$I(x+\Delta x, y+\Delta y, t+\Delta t) = I(x, y, t) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t + Higher\ Order\ Trems \quad (3)$$

However according to the image brightness constraint in equation (2),

$$\frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t = 0 \quad (4)$$

Dividing equation (4) by △t gives ,

$$\frac{\partial I}{\partial x}\frac{\Delta x}{\Delta t} + \frac{\partial I}{\partial y}\frac{\Delta y}{\Delta t} + \frac{\partial I}{\partial t}\frac{\Delta t}{\Delta t} = 0 \quad (5)$$

$$\frac{\partial I}{\partial x}V_x + \frac{\partial I}{\partial y}V_y = -\frac{\partial I}{\partial t} \quad (6)$$

$$I_x V_x + I_y V_y = -I_t \quad (7)$$

To overcome the difficulties of solving this equation with two unknowns, it needs some additional constraint. There are lots of methods for determining optical flow, which address the above additional constraint for estimating the actual flow. In this research, we use Lucas-Kanade method.

Lucas-Kanade Method is a widely used differential method for optical flow estimation. In this method it assumes that the spatial coherence constraint, which is the set of neighbouring points of the pixel under consideration, have similar motion. Thus the optical flow equation can be assumed representing the entire neighbourhood $\boldsymbol{N}$ of pixels points (p_1, p_2, p_3… P_N) , within the window centred at the pixel p. The velocity vector (V_x ,V_y) is then determined through a calculation of least squares.

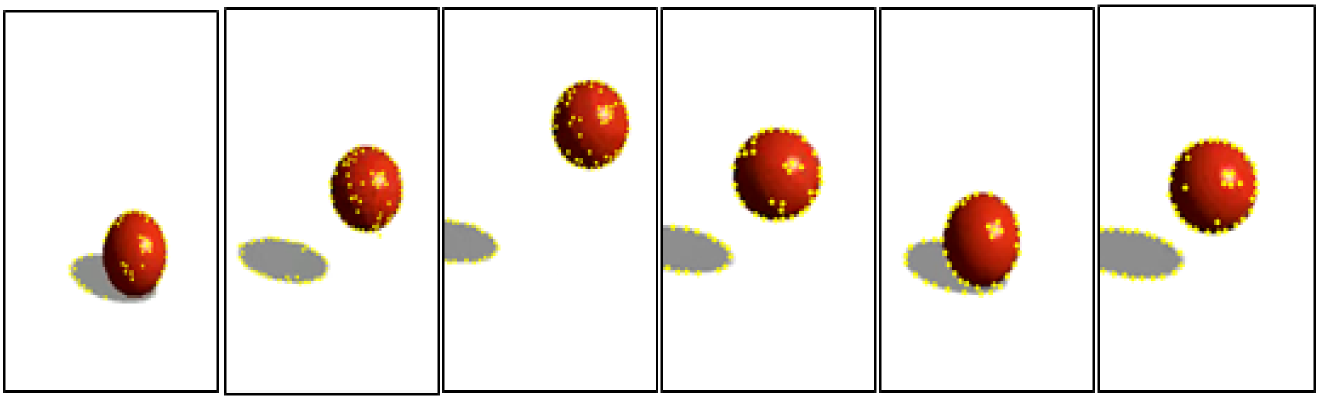

Fig. 4. Feature point selection with the Shi & Thomasi method and tracking of those points in the subsequent frames using the Pyramid Lucas-Kanade feature tracker

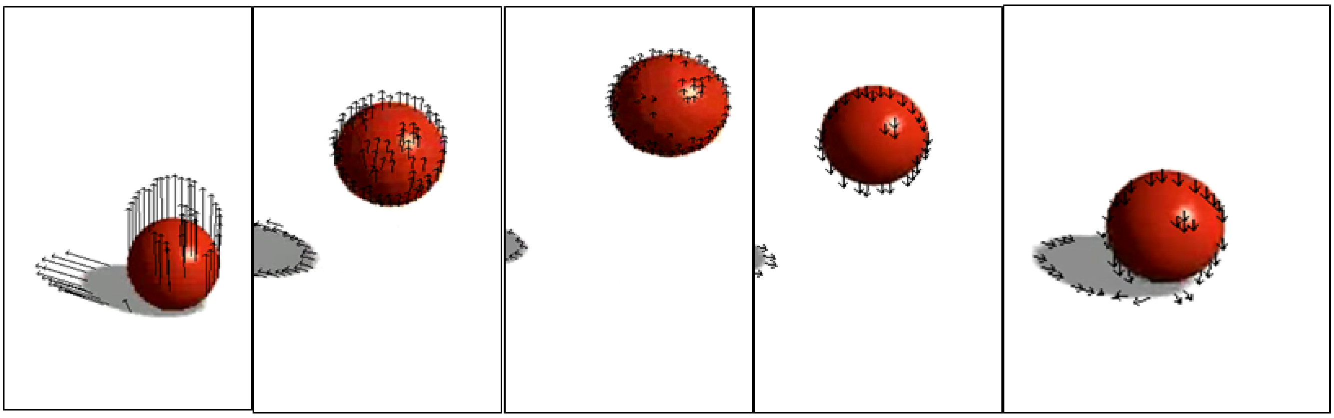

Fig.5. Results of optical flow

$$\begin{pmatrix} \frac{\partial I}{\partial x}(\mathrm{P}_1) & \frac{\partial I}{\partial y}(\mathrm{P}_1) \\ \vdots & \vdots \\ \frac{\partial I}{\partial x}(\mathrm{P}_i) & \frac{\partial I}{\partial y}(\mathrm{P}_i) \\ \vdots & \vdots \\ \frac{\partial I}{\partial x}(\mathrm{P}_n) & \frac{\partial I}{\partial y}(\mathrm{P}_n) \end{pmatrix} \begin{pmatrix} \mathrm{V}_x \\ \mathrm{V}_y \end{pmatrix} = \begin{pmatrix} -\frac{\partial I}{\partial t}(\mathrm{P}_1) \\ \vdots \\ -\frac{\partial I}{\partial t}(\mathrm{P}_i) \\ \vdots \\ -\frac{\partial I}{\partial t}(\mathrm{P}_n) \end{pmatrix} \quad (8)$$

We used Pyramid Lucas-Kanade Algorithm, which is a pyramidal implementation of the Lucas-Kanade feature tracker [21]. At first in this technique, it solves optical flow at the top layer of the pyramid and then use the resulting motion estimates as the starting point for the next layer down. It continues going down the pyramid in this manner until it reaches the lowest level. Therefore by using this method, it can track faster and longer motions [18]. This has less computation and therefore it could be easily adapted for real time applications. [17]

Fig. 4 shows the obtained results for feature point selection and tracking with the use of Shi & Thomasi method and the Pyramid Lucas-Kanade method for the video sequence of a bouncing ball. Fig. 5 shows the optical flows of that image sequences. In that figure, the direction of each arrow represents the direction of optical flow and the length of each arrow represents the magnitude of the optical flow.

C. *Motion Estimation*

This section explains how we calculate the motion of an object in the image frame.

We use velocity of a feature point to estimate the motion. As shown in Fig. 6, the position of a feature point in two subsequent frames at time t and $t + \Delta t$ can be represented as $p_i(t)$ and $p_i(t + \Delta t)$.

The velocity of the feature point is calculated using the equation (9).

$$\overrightarrow{v_\iota(t)} = \frac{p_i(t+\Delta t)-p_i(t)}{\Delta t} \quad (9)$$

Here, if there are N feature points in a frame, then the velocity of each frame is given by the average velocity of feature points in the image frame, as shown in equation (10).

$$\overrightarrow{v_f(t)} = \frac{1}{N}\sum_{i=1}^{N} v_i(t) \quad (10)$$

To explain the result of this step, we used the image sequence of a bouncing ball shown in Fig.5. The Fig.6 shows the changes of the position (6(a)), velocity (6(b)) and acceleration (6(c)) of the ball in each frame during each run.

Bouncing ball is a good example for evaluating position, velocity and acceleration change against time because these graphs convey lots of information about speeding up, slowing down, rising or falling of the ball.

According to the Fig. 6(a), we can easily recognize the occasions when the ball is falling and rising. As shown in Fig. 6(b), when the ball is falling, the velocities on the graph are shown as negative values and the velocity is increasing. On the other hand, when the ball is rising, the velocities on the graph are shown as positive values and the velocity is decreasing. At the top of each bounce the velocity is zero because the ball changes its moving direction.

As the ball falls towards the floor its velocity increases and just before it hits the floor, its velocity becomes maximum.. Immediately after leaving the floor, i.e. at the start of the upward journey, its velocity is maximum and it is in the upward direction. As the ball rises towards its highest position, its velocity approaches zero.

The Fig. 6 complies with the physics of a bouncing ball [22] and hence we can conclude that our proposed method is accurate.

D. Haptic Motion Rendering

Haptic motion rendering means rendering of forces related to the moving objects in the scene. In this section we explain how we calculated forces based on the above velocity changes in the video.

We used SPIDAR-G haptic device shown in fig. 1. It generates six degrees of freedom force feedback by controlling the tension of each string in the system.

However, the high velocities produced by high force and low velocities produced by low force lead to an unrealistic sensation. To overcome this problem of haptic jitter and to get a realistic sensation to the user we need to reduce force for high velocities and increase the force for low velocities. For this purpose, we evaluated two alternative methods, a method using a linear gain controller and a method using a nonlinear gain controller with the objective of identifying the better method..

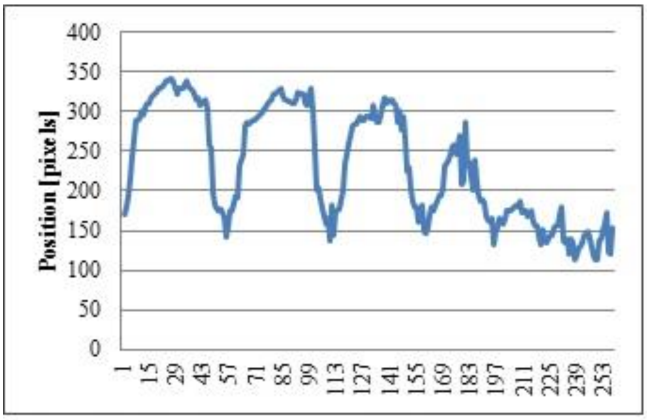

Fig. 6(a). Position changes in bouncing ball

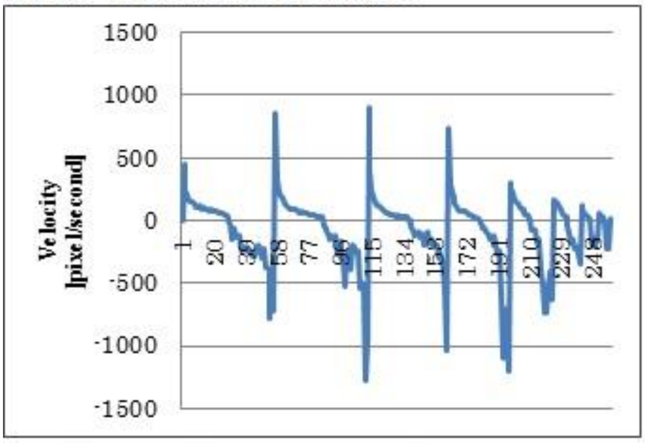

Fig. 6(b). Velocity changes in bouncing ball

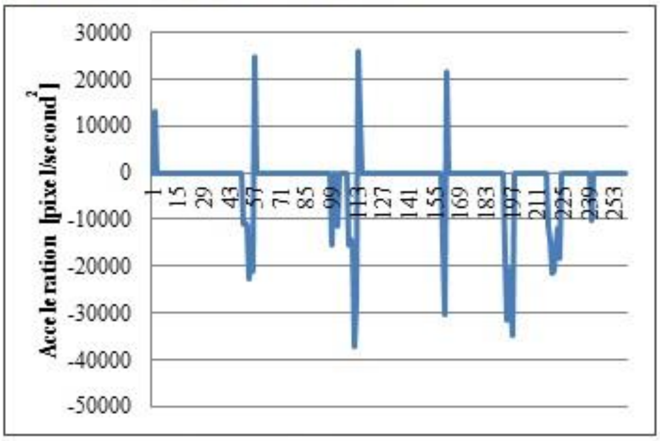

Fig. 6(c). Acceleration changes in bouncing ball

Fig.6. Motion Estimation of bouncing ball

I) Method Using a Linear Gain Controller:

Automatic gain controller is a feature found on many electric circuits that automatically controls the gain of a signal.

We used this concept to control the force of the haptic device.

Using the linear gain controller method, the feedback force is calculated from equation (11). This enables user to get the feeling of the movement of the object.

$$\overrightarrow{F(t)} = k * \overrightarrow{v(t)} \qquad (11)$$

Here k is a gain controller.

Calculation of k is done as in equation (12) to control the feedback force within a sensible region for all velocity levels. In other words, the purpose of the gain controller k is to increase the feedback force for weak changes in velocity and decrease the feedback force for strong changes in velocity.

$$k = \frac{F_{max}}{V_{max}(T)} \qquad (12)$$

Here F_{max} is the maximum force output level of the SPIDAR-G for better sensation for this application. $V_{max}(T)$ is the maximum velocity of the video frame at a time T, which can be expressed as in equation (13).

$$V_{max}(T) = \{ V_{max}(t) \quad 0 \leq t \leq T \qquad (13)$$

We analysed the pattern of k value using the previously mentioned image sequence of the bouncing ball. Fig. 7 shows the results of the changing k values for the image sequence.

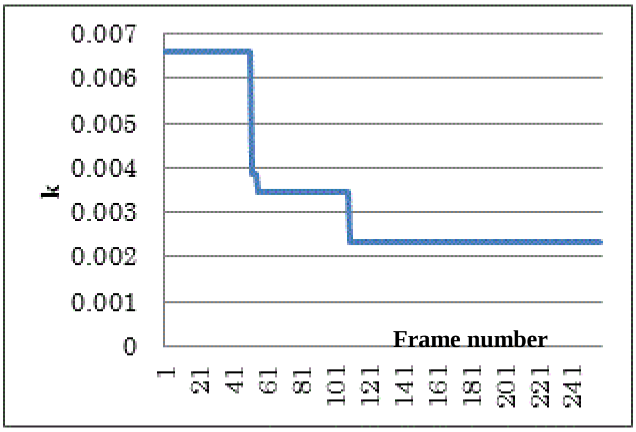

Fig.7. Performance of the k value

Resulting force feedback of the SPIDAR for the image sequence, calculated using equation (11), is shown in Fig 8. Notably, resulting force is within the sensible region of the SPIDAR-G and hence user can smoothly feel the movement of the objects in the video.

II) Method Using a Nonlinear Gain Controller

Similar to the previous method, the purpose of using a nonlinear gain controller is to maintain the feedback force in the sensible region by decreasing the feedback force for high velocities and increasing the feedback force for low velocities. Fig. 9 illustrates this idea in a graphical form.

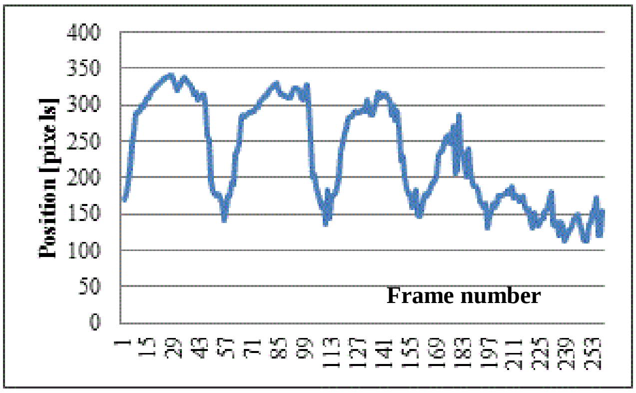

Fig. 8(a). Position changes of a bouncing ball

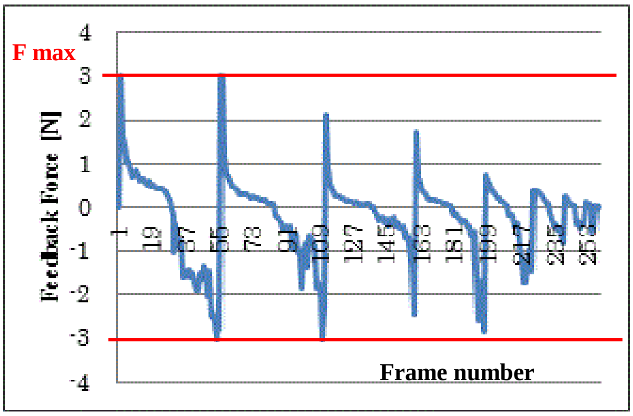

Fig. 8(b). Force changes of a bouncing ball

Fig.8. Feedback Force generated from SPIDAR for the bouncing ball using the linear gain controller method

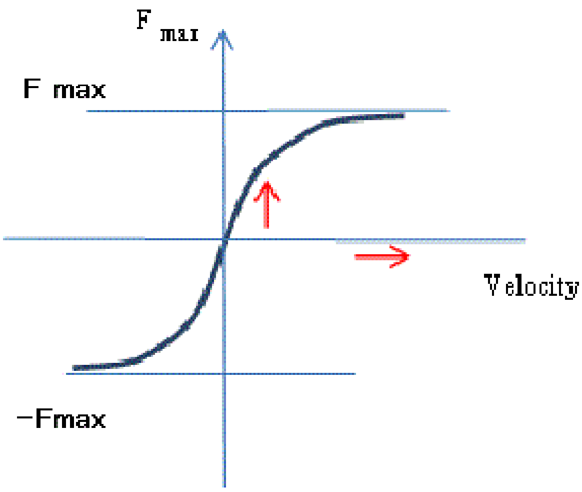

Fig.9. Purpose of the non-linear function

In the non-linear gain controller method we used a nonlinear function to map the velocity into force and the resulting feedback force to sense the motion of objects is shown in equation (14).

$$F(t) = f(v(t)) \quad (14)$$

Our requirement in selecting a nonlinear function was the ability of bringing down the feedback force into the sensible region of SPIDAR-G. As shown in Fig. 9, due to the S-shape behaviour, the sigmoid function proved to be a good candidate. However, as the velocity needs to be zero when changing the moving direction, we got an additional requirement such that the selected sigmoid function needs to go through the origin. Therefore, we selected the inverse tangent function, of which the corresponding candidate as shown in equation (15).

$$F(t) = \frac{2\,F_{max}}{\pi}\tan^{-1}(\alpha\ * V(t)) \quad (15)$$

Here α is chosen arbitrary as 0.01.

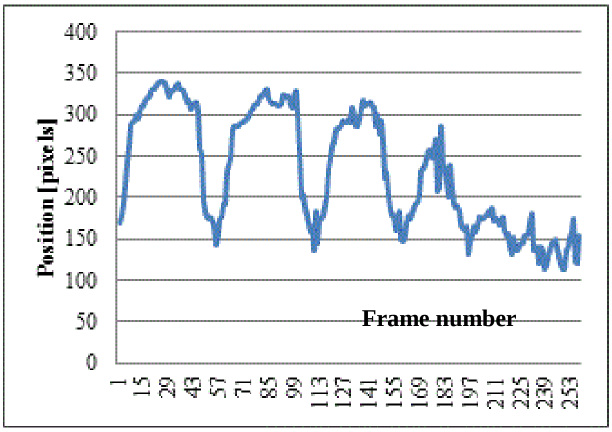

Fig. 10(a) Position changes of a bouncing ball

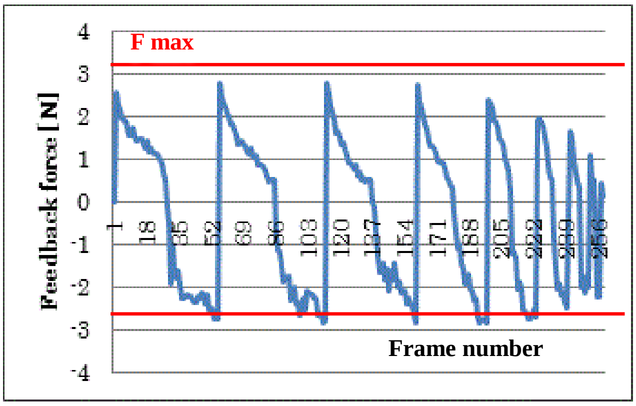

Fig. 10(b) Force changes of a bouncing ball

Fig.10. Feedback Force generated from SPIDAR for the bouncing ball using a nonlinear function

Resulting force feedback of the SPIDAR for the image sequence calculated using equation (15), is shown in Fig. 10. Notably, resulting force is within the sensible region of the SPIDAR-G and hence user can smoothly feel the movement of the objects in the video.

We can compare the resulting feedback force using two methods, the method using linear gain controller and the method using nonlinear gain controller with respect to the position and velocity changes by using Fig. 8 and Fig. 10. It is clear from Fig. 10 that nonlinear gain controller method outperforms the linear gain controller method, since it increases the feedback force for low velocities than the gain controller method and decreases the feedback force for high velocities reasonably within the sensible region of the SPIDAR-G haptic device. In other words, in the method using the nonlinear gain controller user can feel the movement of the objects in the video even for smaller changes of the velocity. As a result user can get a continuous feeling of the movement of the ball even when the ball is in the air.

IV. Results of Experimental Evaluation

In order to further evaluate which method gives better feeling about the movement of the objects in the video, we performed an experiment by getting the involvement of real users. We used different video sequences such as a bouncing ball, waterfall, a blowing tree and a cat moving its hand as test beds. Those videos are 2D image sequences, which includes only one moving object at a time. Moreover, the camera is not moving in those videos.

The users are experienced users of SPIDAR system. For each user we assigned three videos randomly from above and conducted the experiment using the two methods, i.e linear gain controller method and the nonlinear gain controller method. After the experience with the three image sequence, we asked the users to rate each method based on their feeling using a questionnaire.

As the first thing, we asked them to rate the two methods in a scale of \`bad\`, \`poor\`, \`average\`, \`good\` and \`very good\` based on their feeling of movement of the object in the video. The resulting responses are shown graphically in fig. 11.

However to simplify the analysis, we generally considered the responses of \`poor\` and \`bad\` as negative responses such that the feeling of the movement is not satisfactory for the respective method. On the other hand, the responses of \`average\`, \`good\` and \`very good\` have considered as positive responses such that the feeling of the movement is satisfactory for the respective method. These two categories of negative and positive responses are denoted by \`Bad\` and \`Good\` respectively. Here we took \`average\` also as a positive response, because we believe that the proposed method is successful even if that method can give at least some kind of feeling to users.

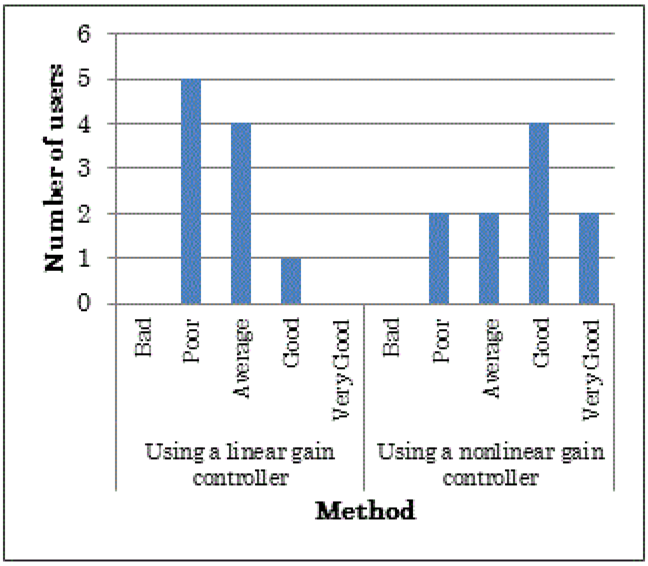

Fig.12. Distribution of responses for using a nonlinear gain controller method

The result of this new rating is shown in Table 1 and also shown graphically in Fig. 12. This graph shows how users rate each method based on their feeling as `Good ` or `Bad`. We tested with the involvement of ten numbers of real participants and it is clear that more than 70% of users responded that using the nonlinear gain controller method is better than using a linear gain controller method.

Furthermore, we evaluated the usefulness of haptization with our proposed method to users by evaluating the system with and without haptic feedback. We assumed that users are good at the SPIDAR system and also used the videos which were used in the previous experiments. Moreover, for haptic feedback, we used the nonlinear gain controller method, which most of the users rated as the better method from the previous experiment. In the current experiment, we tested the system using three aspects; reality, interactivity and comfortability. Reality evaluates whether the system enables getting a real feeling of the scene. Interactivity evaluates whether the user feels him or herself as better involved in the scene as a part of it. Comfortability means whether the user feels more comfortable with the additional sensation provided with haptic feedback. For this purpose we used a questionnaire and in the questionnaire users were asked to rate their experience for each aspect in both situations (i.e. with and without haptic feedback) in a scale of `Bad`, ` Poor`, `Average`, `Good` and `Very good`. The results are shown in the Fig. 13.

To compare the difference between the two methods, we used the value of weighted average of users` responses for each method. When the reality was evaluated without haptic feedback, it`s value was 2.4 and it was 4.2 when evaluated with haptic feedback. When the aspect of interactivity was evaluated without haptic feedback, average user response value was 1.8 and it was 3.6 for the case with haptic feedback. When the aspect of comfortability was evaluated the average user response value was 3.2 both in the case of with and without haptic feedback. The above results are shown in the table 2.

Method	Good (User Percentage %)	Bad (User Percentage %)
Using a linear gain controller method	50	50
Using a nonlinear gain controller method	80	20

Table 1. User responses for each method

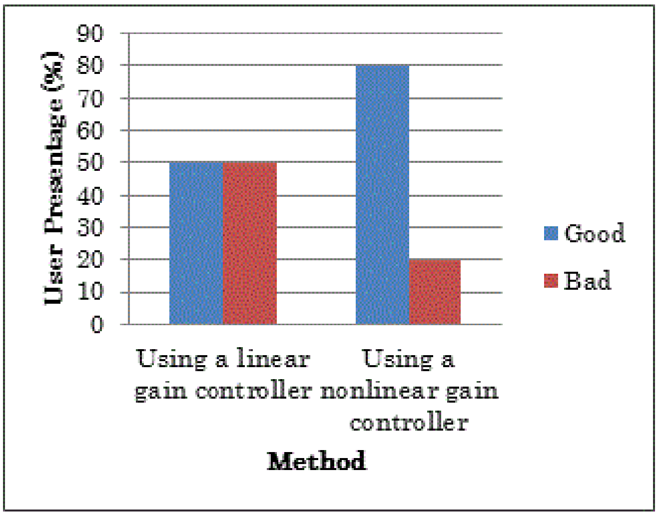

Fig.12. Result of responses of users for each method

Aspect	Without haptic feedback (Weighted Average Value)	With haptic feedback (Weighted Average Value)
Reality	2.4	4.2
Interactivity	1.8	3.6
Comfortability	3.2	3.2

Table 2. User's responses for the three aspects with and without haptic feedback

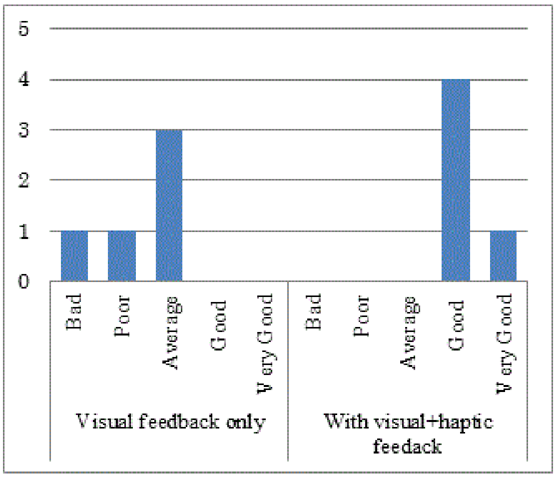

Fig. 13(a). Comparison of reality using with and without haptic feedback

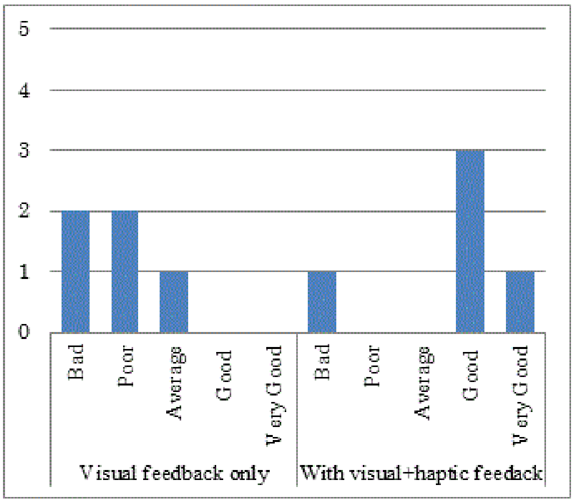

Fig. 13(b). Comparison of interactivity using with and without haptic feedback

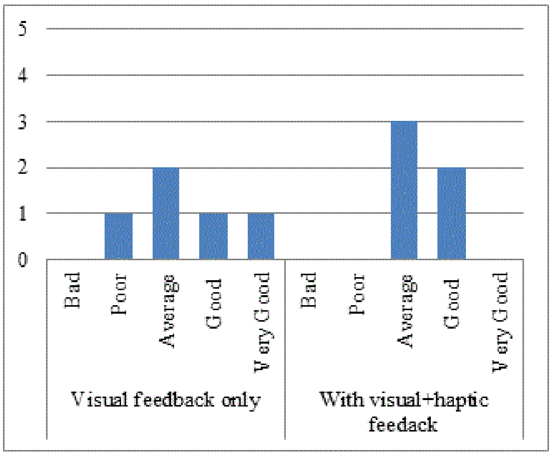

Fig. 13(c). Comparison of comfortability using with and without haptic feedback

Fig.13. Distribution of responses for using a nonlinear gain controller method

From the Table 2 it is clear that having haptic feedback enhances the user experience in the aspects of reality and interactivity. However, we got moderate feedback from the users regarding the comfortability aspect as some users concluded that having only the visual feedback is more comfortable than having both visual and haptic feedback.

V. DISCUSSIONS

In this research we concern how to associate haptic signals with a video to feel the motion of objects. To achieve the above objective, we have experimentally evaluated two methods i.e. a linear gain controller method and a nonlinear gain controller method, with the objective of identifying the better one. We can conclude that using a nonlinear gain controller is more effective than using a linear gain controller because user can get a continuous feedback force and hence, can get the continuous feeling of the movement of the objects in the video. As a result, user can feel the movement of the objects in the video as additional information than seeing and hearing it.

We tested our method with different types of 2D image sequences and identified that with haptic feedback enhances the video viewing experience of the users than without haptic feedback.

The videos we used for our experiments had only one moving object. However, real videos are object rich environments with multiple moving objects. Therefore, it is highly necessary to research on how to interact with multiple objects in object rich environment. Further, real image sequences include lots of background noise, which affects the feedback force of the rendered object. Therefore, in the future, we expect to improve our method to eliminate such background noise and improve the output.

Furthermore, we believe that it would be interesting and highly necessary to improve this method to render 3D motion from 2D image sequence as 3D technologies will increasingly become popular in the future. In consequence, it will allow the users to get the 6 degrees of freedom force feedback based on rotation and translation of the motion of objects.

Moreover, since SPIDAR-G is a grip type haptic device, user can feel the movement of the object only to the gripping hand. However this is not adequate to a realistic sensation. Therefore, another possible direction of future work would be to improve the system by identification of a suitable device such that it enables the user to get the whole body sensation.

REFERENCES

[1] [1] Hayward V., Astley O. R., Cruz-Hernandez M., Grant D., Robles-De-La-Torre G. (2004). Haptic Interfaces and devices: *Journal of Sensor Review*, 24(1): 16-29.

[2] Basdogan C., De S., Kim J., Muniyandi M., Kim H., Srinivasan M.A. (2004). Haptics in Minimally Invasive

Surgical Simulation and Training: *Proceedings of the IEEE Computer Graphics and Applications*, 24(2): 56.

[3] Wu J., Wang D., Zhang Y. (2009). Virtual Fixture based Haptic Rendering of Handwriting, *IEEE International Conference on Virtual Environments, Human-Computer Interfaces, and Measurement Systems:* Hong Kong, China : 16-21.

[4] Sato M., Liu X., Murayama J., Akahane K., Isshiki M (2008). A Haptic Virtual Environment for Molecular Chemistry Education: *Transactions on Edutainment I*, LNCS 5080: 28-39.

[5] Morris D., Joshi N., Salisbury K.(2004). Haptic Battle Pong: High-Degree-of-Freedom Haptics in a Multiplayer Gaming Environment: *Proceedings of Experimental Gameplay Workshop at Game Developers Conference (GDC)'04*

[6] Dindar N., Tekalp A. M., Basdogan C. (2010). Immersive Haptic Interaction With Media: *Proceedings of the Visual Communications and Image Processing 2010*, Huangshan, China.

[7] Sato M. (2006). A String-based Haptic Interface SPIDAR: *ISUVR2006*, Yanjin, China, 191, 1-5.

[8] Akahane K., Hasegawa S., Koike Y., Sato M. (2006). A proposal of a High Definition Haptic Rendering for Stability and Fidelity: *Proceedings of the 16th International Conference on Artificial Reality and Telexistence Workshop*:162-167.

[9] Kim S., Koike Y., Sato M. (2002). Tension Based 7DOFs Force Feedback Device: SPIDAR-G: *Transactions on Control, Automation, and Systems Engineering,* 4(1): 9-16.

[10] O`Modhrain S., Oakley I. (2003). Touch TV: Adding Feeling to Broadcast Media : *Proceedings of the 1st European Conference on Interactive Television: from Viewers to Actors*, Brighton, UK.

[11] Cha J., Eid M., Saddik A.E. (2009). Touchable 3D Video System: *ACM Transactions on Multimedia Computing, Communications and Applications*, 5(4).

[12] Cha J., Kim S., Ho Y. (2006). 3D Video Player System with Haptic Interaction based on Depth Image-Based Representation: *IEEE Transaction on Consumer Electronics*, 52(2).

[13] Cha J., Kim S., Oakley I., Ryu J., Lee K. H. (2005). Haptic Interaction with Depth Video Media: *Advances in Multimedia Information Processing - PCM 2005, Lecture Notes in Computer Science*.

[14] Kim S., Cha J., Lee S., Ryu J., Ho Y. (2007). 3DTV System using Depth Image-Based Video in MPEG-4 Multimedia Framework: *3DTV Conference*.

[15] Liu X., Akahane K., Isshiki M., Sato M. (2010). Design and Implementation of an Image Haptization System: *3DSA2010 (International Conference on 3D Systems and Applications）* : 260-262.

[16] Liu X., Jayasiri A., Akahane K., Sato M. (2012). A proposal of a Hapitzation System for Depth Images with Local Deformation, *3DSA2012 (International Conference on 3D Systems and Applications）*, June.

[17] Abdat F., Maaoui C., Pruski A. (2008). Real Time Facial Feature Points Tracking with Pyramidal Lucas-Kanade Algorithm: *Proceedings of the 17th IEEE International Symposium on Robot and Human Interactive Communication*, Munich, Germany

[18] Bradski G. and Kaehler A. (2008). Learning OpenCV: *O`Reilly Media, Inc.*, 1005 Gravenstein Highway North, Sebastopol, CA 95472.

[19] Horn B. K. P., Schunck B.G. (1981) . Determining Optical Flow: *AI(17*), 1(3): 185-203.

[20] Sommer J., Arango R., Grest D. (2008). Sparse Optical Flow for 3D Objects.

[21] Bouguet J. (2000). Pyramidal Implementation of the Lucas Kanade Feature Tracker Description of the Algorithm: *Intel Corporation, Microprocessor Research Lab*.

[22] Cross R. (1999). The bounce of a ball: *American Journal of Physics*, 67(3): 222-227.