# Real-Time Stereoscopic Conversion with Adaptable Viewing Distance at Personal Stereoscopic Viewing Devices

Ilkwon Park and Hyeran Byun

Department of Computer Science, Yonsei University, Seoul, 120-749, South Korea
{ikheart,hrbyun}@yonsei.ac.kr

**Abstract.** This paper proposes the real-time stereoscopic conversion method with adaptable viewing distance by the viewer's eye detection at personal stereoscopic viewing devices. The distance between a screen and viewer is often changed by moving a personal display or moving head position. It causes a visual discomfort and low quality stereoscopy. Our proposed method is divided into two parts; eye detection part to acquire viewer's eye positions and stereoscopic generation part in real-time. In the first part, we can measure an interocular distance of a single viewer by the proposed eye detection method and also calculate the distance between a screen and a viewer by using the principle of stereoscopic generation. In the other part, we propose an efficient motion-based approach for conversion from 2D video to stereoscopic video. The disparity between left image and right image is adaptively adjusted by viewer's interocular distance. Adaptation time for a 3D immersion during 3D stereoscopic viewing is remarkably reduced and stereoscopy is personally optimized. Therefore, our proposed system can offer optimized stereoscopic conversion and more freedom of viewer's head considering the distance between a screen and a viewer through the proposed eye detection.

**Keywords:** Mobile stereoscopic system, Adaptable stereoscopic conversion, Viewer adaptable 3D.

## 1 Introduction

Within the past decade, many efforts have been made to convert a monoscopic video to a stereoscopic video by the aid of video device and software [1-2]. Generally, stereoscopic video is mainly applied to scientific visualization, simulation and entertainment [3]. However, the stereoscopic displays are getting popular in these days due to low cost by hardware improvements and consumer's visual desire. Moreover, we can find easily personal stereoscopic viewing devices in everywhere according to supply mobile phone that is available with multimedia data, personal digital assistant (PDA), and laptop computer as well as personal computer.

However, stereoscopic contents are still insufficient in spite of many efforts [1-3]. There are two approaches for a stereoscopic video generation. First approach is generated by using stereoscopic camera. This approach generates high quality

M.J. Smith, G. Salvendy (Eds.): Human Interface, Part I, HCII 2007, LNCS 4557, pp. 738–746, 2007.

stereoscopic contents though this camera is too expensive and these stereoscopic contents are only generated by newly recording real scenes without reproducing conventional two dimensional video. The other one is generated by converting a conventional two dimensional video into three dimensional stereoscopic video. Our proposed method is also stereoscopic conversion approach. It has a lot of advantages such as low cost and using a conventional video that is uniquely existed though we have still faced with eyestrain and low-quality stereoscopy as disadvantages [3].

Our proposed system extends fundamentally motion-based stereoscopic conversion system according to motion types. This method converts motion information in conventional 2D video to 3D disparity to generate stereoscopic video. Extracted motion vectors are used to decide motion types and the amount of disparity. Proposed system consists of three motion types such as horizontal, vertical, and static. Motion types follow the decision rule from [3]. For decision of 3D disparity, we adapt roughly the distance between screen and viewer calculated by obtaining user's interocular distance and finely an amount of motion from 2D video.

Based on the principle of stereoscopic generation, the distance between pupils of viewers and the distance between screen and viewer's pupils are important clues to generate optimal 3D stereoscopy. If we generate stereoscopic video by only motion vector without certain information of these distances, viewers must always have kept the approximate distance from the screen uncomfortably as well as struggled for finding the optimal distance each time. To solve this problem, we apply interocular distance based on a viewer's eye positions to generate stereoscopic contents adaptively. This proposed method can make a more freedom of viewer's head and optimal 3D stereoscopy for single viewer.

For detecting viewer's eye positions, we propose robust real-time eye detection method under various illuminations. In our system, we can consider not only personal computer that is fixed but also mobile phone or PDA that is moving in outdoor environments. Furthermore, we can also operate our proposed system in dark circumstance such as night time and home under dim light because of using the infrared illuminators. For real-time processing, we combine the eye detection by using Iris mask for open eye detector with the one by using eye corner filter for closed eye detector [5], [7]. Both methods are very simple and fast. However, each method has relatively low accuracy. Therefore, we highly boost this detection rate by using pattern recognition approach as eye verification step. For real time processing, we configure an efficient formation of the cascaded form of SVM as two level cascaded SVM. This method highly kept the accuracy while speed up the processing time in its test phase.

The remainder of this paper is organized as follows. In Section 2, we describe the proposed system architecture. In Section 3, we represent the proposed eye detection method and the measure of interocular distance. In Section 4, we represent motion detection method in 2D video contents and adaptable stereoscopic conversion method according to viewer's eye position and motion information detected from 2D video contents. Experimental results and the estimation of our system are provided in Section 5, and finally we make a conclusion in Section 6.

## 2 The Proposed System Architecture

Our proposed system mainly consists of two parts; User Adaptation Engine(UAE) and Video Conversion Engine(VCE). UAE includes the process acquiring viewer's eye position and extracting interocular distance of a single viewer. VCE is the process of motion analysis in 2D video contents and the process of stereoscopic video generation based on viewer's information.

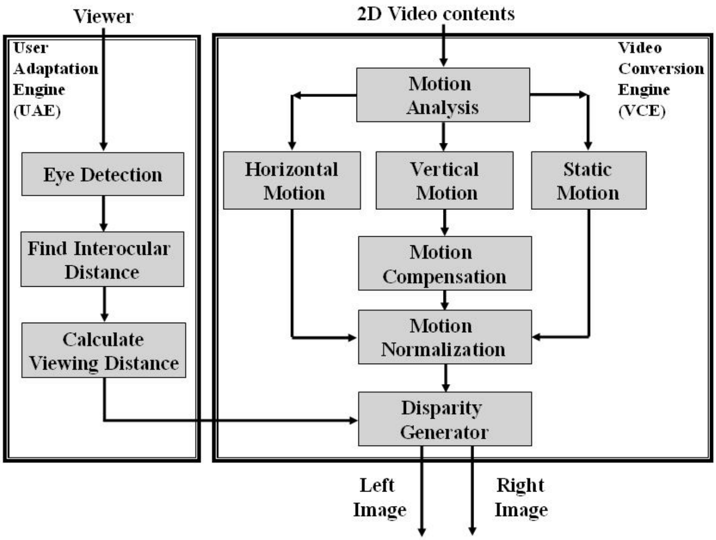



**Fig. 1.** The overview of the proposed system architecture

Fig.1 shows the overview of the proposed stereoscopic conversion system with viewer adaptable function. We assume that this system is used by only single viewer. The our proposed system is operating on a personal 3D display, such as mobile phone, PDA, and laptop computer as well as personal computer.

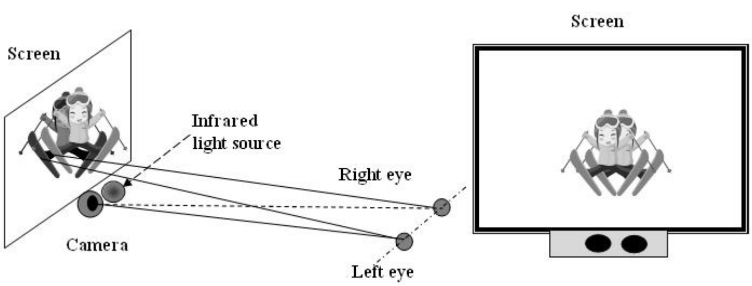



**Fig. 2.** The installation of personal 3D display and a CMOS camera

UAE is the process module based on computer vision technique. This processing module plays an important role in adaptable stereoscopic generation and it has three parts; eye detection, interocular distance extraction and calculate the distance between the screen and viewer's eye position.

On the other hand, VCE is the processing module that is motion-based stereoscopic converting module with user adaptable modules, while typical systems convert a simply 2D to 3D stereoscopic video based on motion information. Single CMOS camera and infrared light source are installed in front of personal 3D display to acquire viewer's eye position as Fig. 2.

## 3 User Adaptation Engine

In this paper, UAE extracts user information based on viewer's eye position and applies it to VCE in real-time. Our proposed system must operate not only in bright circumstance but also in dark one. Therefore, infrared illuminators are used with camera as invisible illuminator. To acquire viewer's eye position, we deign efficient eye detection algorithm. This approach consists of three parts; open eye detector, closed eye detector and eye verifier as Fig. 3. This technique gives us fast and accurate performance for finding eye position in spite of combining with a pattern recognition method on eye verification.

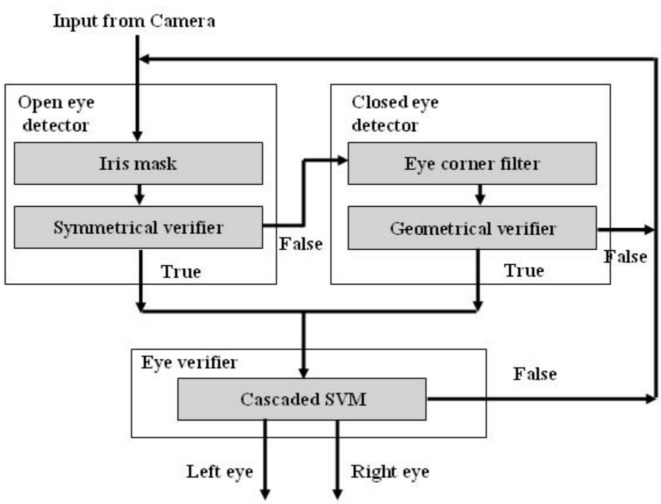



**Fig. 3.** Proposed eye detection system for real time operation

### 3.1 Open Eye Detector

Open eye detector is based on geometrical mask as iris shape proposed in [5]. This algorithm is very useful for detecting open eyes or partially open eyes. Through the iris mask, they can directly extract eye candidates in the whole image without face segmentation or limited face region. In [5], search step of similar regions decided eye candidates by using the similarity of both shapes after comparing left eye with right eye. We separate this step as symmetrical verifier. Furthermore, we modified this algorithm to align each candidate patches by the similarity of edge distribution before symmetrical verifier. Although it has a lot of advantages, we can not be satisfied with the results for partially open eyes and closed eyes. Therefore, we use closed eye detector based on eye corner filter for detecting closed eyes and partially open eyes.

### 3.2 Closed Eye Detector

Eye corner filter is a fast and reliable preprocessing method for eye detection [9]. In this paper, the eye corner filter proposed in [7] is partially used to detect suitable candidates of eye region as closed eyes and partially open eyes. Eye corner filter consists of two shapes: left filter and right filter shapes like eye corners. As input frame is convoluted with the filter, we can detect the positive areas over the threshold. The threshold is determined to take 10% from high rank of the result by a convolution with total eye area. Positive areas by both filters are merged by morphological operation. Each positive area detected by eye corner filter is cropped with 5 pixels margins from the boundary of positive area as eye candidates if each positive area has a reasonable ratio of horizontal length to vertical length. Cropped eye candidates are normalized the size of 41 by 21 for the eye verification.

### 3.3 Eye Detection by Using Cascaded SVM

We verify the filtered eye candidates using the support vector machine (SVM). For computational efficiency, we evaluate the SVM by the sequential evaluation algorithm [4] that we called the cascaded SVM. In general, eye verification deals with the two-class problem of eye vs. noneye. However, since variation reduction of the eye class can improve the verification performance, we consider three classes of open eye, closed eye and noneye. Then we design the verification procedure with the cascaded SVM [7]. Fig. 4 shows the schematic description of the proposed eye detector. The classifiers are determined as similar to [4], but we choose their thresholds in the different way in order to reduce the false negative rates.

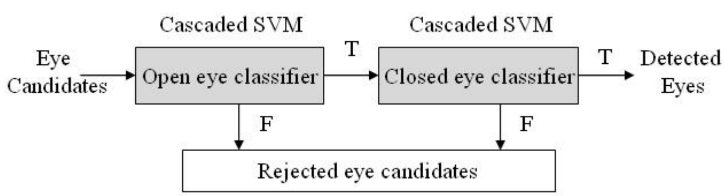



**Fig. 4.** Schematic description of the proposed eye detector using the cascade SVM

### 3.4 Calculate Interocular Distance

In the previous section, we found viewer's eye positions by using our proposed method. Based on viewer's eye position, we can calculate Interocular distance in real-time. This is the key idea of this paper that interocular distance reflects not only various personal Interocular distances between two eyes but also the distance between a screen and a viewer. For example, the farther interocular distance is, the closer the distance between a screen and a viewer is. In opposite case, the farther the distance between a screen and a viewer is, the closer interocular distance is. Interocular distance is represented by the number of pixels in the captured image from a camera.

# 4 Video Conversion Engine

In this section, we represent motion-based 2D to 3D stereoscopic conversion and also we represent how to viewer's information is applied in our proposed system as user adaptable system.

## 4.1 Motion Detection

In this paper, Optical flow approach that is fast and simple is used for real-time motion estimation. Estimated motion information is represented by orientation and magnitude which is based on the position of each feature point. Each orientation and magnitude is computed by equation 1 and equation 2. Motion vector MV is (MVx, MVy), where MVx and MVy are horizontal and vertical components, respectively.

$$Orientation(\theta) = \tan^{-1}\left(\frac{MV_y}{MV_x}\right). \tag{1}$$

$$Magnitude\ (M) = \left|\sqrt{MV_x^{\ 2} + MV_y^{\ 2}}\right|. \tag{2}$$

To generate a disparity between left and right image from single image, we use only the average motion vector (AMV) of the current frame. If current AMV is larger than threshold ( $\theta$ ) that is experimentally acquired, this frame has enough motion information as horizontal or vertical motion. Otherwise, it classifies static motion. In motion analysis, vertical motion is separated with horizontal motion by empirically acquired threshold and it must be compensated because vertical disparity makes viewer's eye uncomfortable. In the case of static motion, previous AMV is used as the AMV in current frame. Acquired motion magnitude of AMV is too small or large for using stereoscopic disparity. Therefore, it must be normalized within maximum disparity for a viewer. However, our proposed system uses both the disparity by AMV and interocular distance. Therefore, the sum of disparity by AMV and interocular distance is lower than maximum disparity for a viewer. In this paper, the magnitude of AMV is normalized within 10 pixels, this is empirically set.

## 4.2 Generation of Disparity

To generate disparity between left image and right image from a single image, the magnitude of disparity and shift direction are necessary. Shift direction is represented by only horizontal direction. Disparity assignment is firstly given by viewer's interocular distance. And Magnitude of AMV is applied as a variation from given disparity. Equation 3 represents the disparity assignment with magnitude of AMV and interocular distance. In Eq. 3, *I_disparity* represents the normalized disparity from interocular distance. *nAMV* represents the normalized disparity from the magnitude of AMV and *max_AMV* is a maximum scale of normalized disparity.

$$Disparity = I\_disparity + (nAMV - \max\_AMV/2)\,. \tag{3}$$

Fig. 5 represents a disparity compensation for the same depth unrelated with the viewing distance. As viewer's eye position is moved from distance 1 to distance 2, depth 1 and converging point 1 are also moved depth 2 and converging point 2. Therefore, viewer watches the video with the different depth although the depth is a physically the same. To solve this problem, the disparity is compensated to be the same converging point as the viewing distance is changed. Finally, for disparity variation, compensated disparity and motion disparity are integrated as Eq. 3.

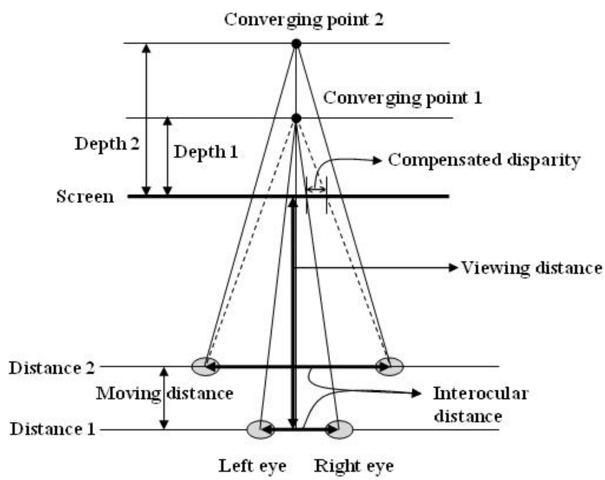



**Fig. 5.** Disparity compensation for the same depth unrelated with the distance between screen and viewer's eye position

## 5 Experiments

In this section, we experiments the acquisition process of viewer's information from UAE and the adaptation process of stereoscopic conversion from VCE. To measure viewer's interocular distance, we must define initial distance between screen or camera and viewer's eye position as 70cm in our laptop computer by many experiments and then adjust initial shift distance to watch an optimized stereoscopy.

After initialization for a viewer, the UAE adaptively calculates shift distance followed by viewer's interocular distance. That is, interocular distance reflects the various distances between screen and viewer's eye position. Viewing distance is for minimum 45cm to maximum 120cm for stable watching by characteristics of stereoscopic monitor. Through our experiments, viewing distance is roughly in inverse proportion to the disparity between left image and right image.

Fig. 6 shows the some results of stereoscopic conversion for three video sequences using proposed adaptation methods. In Fig. 6, (a) is converted from typical mpeg 2D test video that called coastguard to stereoscopic video. (b) and (c) is converted from video sequence selected only one view from multiview sequence given by Microsoft Research []. *Disp* represents the disparity between left image and right image. *VD* represents the viewing distance between a screen and a viewer's eye. In Fig. 6, viewing distance is represented as three layers; 100, 75 and 50. Therefore, viewer can watch stereoscopic video with stable depth sense as applying different disparity according to viewing distance

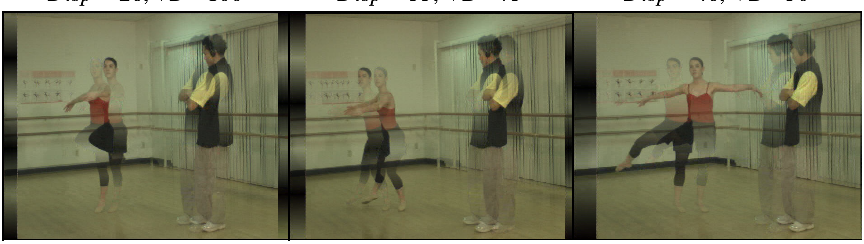
(a)

*Disp* = 19, *VD* =100 *Disp* = 25, *VD* =75 *Disp* = 36, *VD* =50

(b)

*Disp* = 26, *VD* =100 *Disp* = 33, VD =75 *Disp* = 46, VD =50

(c)



*Disp* = 30, *VD* =100 *Disp* = 36, *VD* =75 *Disp* = 48, *VD* =50

**Fig. 6.** Some results of stereoscopic conversion adapted by viewing distance

# 6 Conclusion

This paper proposes the adaptable stereoscopic conversion system by the viewer's eye position on personal stereoscopic viewing devices. For adoptable stereoscopic conversion for a viewer, we propose efficient eye detection method for acquiring viewer eye position. This method is a simple and fast for real-time process as well as accuracy by using cascaded SVM as a verification step. The disparity of left image and right image is continuously adjusted according to the viewing distance represented by acquired viewer's eye position. Therefore, our proposed system can offer personally optimized stereoscopic conversion unrelated with viewing distance.

**Acknowledgments.** This research was supported by the MIC, Korea, under the ITRC support program supervised by the IITA (IITA-2006-C1090-0603-0017).

## References

1. Meesters, L.M.J., IJsselsteijn, W.A., Seuntiëns, P.J.H.: A Survey of Perceptual Evaluations and Requirements of three-dimensional TV. IEEE Transctions on Circuits and systems for Video technology, vol. 14(3) (2004)
2. Battiato, S., Capra, A., Curti, S., La Cascia, M.: 3D Stereoscopic Image pairs by Depth-Map Generation. In: Proceeding of the 2nd International Symposium on 3D Data Processing, Visualization, and Transmission, pp. 124–131 (2004)
3. Kim, M., Park, S., Kim, H., Artem, I.: Automatic conversion of two-dimensional video into stereoscopic video. In: Proceeding of SPIE, November 15, 2005, vol. 6016 (2005)
4. Burges, C.J.C.: Simplified support vector decision rules. Intl. Conf. on Machine Learning 1, 71–77 (1996)
5. D'Orazio, T., Leo, M., Cicirelli, G., Distante, A.: An algorithm for real time eye detection in face images. In: Proc.17th Int. Conference on Pattern Recognition, vol. 3, pp. 278–281 (2004)
6. Liu, X., Xu, F., Fujimura, K.: Real-time eye detection and tracking for driver observation under various light conditions. IEEE Intelligent Vehicle Symposium 2, 344–351 (2002)
7. Park, I., Ahn, J.-H., Byun, H.: Efficient Measurement of Eye Blinking under Various Illumination Conditions for Drowsiness Detection Systems. 18th International Conference on Pattern Recognition, vol. 1, pp. 383–386 (2006)
8. Romdhani, S., Torr, P.H.S., Scholkopf, B., Blake, A.: Computationally efficient face detection. In: Proc. 8th Int. Conf. on Computer Vision, pp. 695–700 (2001)
9. Zhu, J., Yang, J.: Subpixel eye gaze tracking. In: Proc. 5th IEEE International Conference on Automatic Face and Gesture Recognition, pp. 124–129 (2002)