Video Saliency Modulation in the HSI Color Space for Drawing Gaze

Tao Shi and Akihiro Sugimoto

National Institute of Informatics,
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan
shitao.2011@my.bristol.ac.uk,
sugimoto@nii.ac.jp

Abstract. We propose a method for drawing gaze to a given target in videos, by modulating the value of pixels based on the saliency map. The change of pixel values is described by enhancement maps, which are weighted combination of center-surround difference maps of intensity channel and two color opponency channels. Enhancement maps are applied to each video frame in the HSI color space to increase saliency in the target region, and to decrease that in the background. The TLD tracker is employed for tracking the target over frames. Saliency map is used to control the strength of modulation. Moreover, a *pre-enhancement* step is introduced for accelerating computation, and a post-processing module helps to eliminate flicker. Experimental results show that this method is effective in drawing attention of subjects, but the problem of flicker may rise in minor cases.

Keywords: visual focus of attention, saliency, video modulation, gaze navigation.

1 Introduction

To understand the behavior of human visual attention is an important task in the study of neuroscience. Human gaze can be directed by the ability of learning, recall, or recognition. More frequently, the direction of gaze is controlled by our born ability to discriminate object appearances. To understand the principle of human vision, creating a computational model for visual attention is a primary task in the cross subject of neuroscience and computer science.

One promising way to estimate the visual focus of attention is to use a saliency map, which identifies image regions that draw more human attention. Koch and Ullman [9] firstly proposed a prototype of saliency map model. Itti et al. [6] summarized the previous work and proposed the basic bottom-up saliency map model. Afterwards, Itti et al. [7] extended the saliency map to deal with videos by addinsg flicker and motion detection. In later years, more models of saliency computation were proposed. The graph-based visual saliency [4] added an 'activation' step after the extraction of original features. The ability of feature selection was improved in this work, but the proposed algorithm is more complex.

R. Klette, M. Rivera, and S. Satoh (Eds.): PSIVT 2013, LNCS 8333, pp. 206–219, 2014.

Huang et al. [5] proposed a saliency model in HSV color space, for extracting regions of interest. Although their proposed saliency map is defined in a more human-perception oriented color space, key functions of map components are similar to those of Itti et al. 's bottom-up model.

One of the potential applications of saliency maps is the gaze-based interface where we need to draw user's visual attention. With the help of saliency map, it is possible to navigate the visual focus of attention by modulating features in the image. By this way, we can encourage the audience to watch the information we stress, without any aid of texts or overlapped graphics. In broadcasting of sports games, on the other hand, following the motion of a single player by image modulation will lead to a comfortable visual experience. Additionally, if we apply such image modulation to rear-view images displayed inside a car, the driver need not to read texts anymore while driving.

An early trial on gaze navigation by image modulation was to shift the hue and luminance to raise attraction, and then remove the modulation immediately when the subject's gaze has moved to the target [1]. A pixel-wise modulation for still images was, on the other hand, proposed by Hagiwara et al. [3]. In this method the gaze was drawn to a given target in the modulated image, but unnatural color was observed in the modulated image. Another algorithm to modulate an image and video was proposed [11,13]. In this work the saliency map was generated in L*a*b* space, and the map was applied to images for modulation. Although this approach produced a fine and neutral result, a threshold map for each image needed to be manually preset, which made the method impractical in the real situation. In their work, each frame was independently adjusted and coherence between video frames was not taken into account.

Similar to the works above, translating the visual attention to a specified region using video modulation is our primary task. Namely, the goal of this work is to draw the gaze to a given target in a video by modulating the frames of the video. During the process, a target region is manually specified in the first frame, and the corresponding regions in subsequent frames of a video are automatically tracked by the TLD tracker [8], which is known to work robustly and accurately. Saliency in the region is estimated by a saliency map [7], with which we create a mask. Then we generate several centre-surround difference maps from the image and multiply them with the mask. After this, the maps are added back to the image as the modulation. The key idea of this operation is to enlarge the local contrast on the target. Considering the visual satisfaction for the viewer, we modulate both intensity and color simultaneously in the HSI space. Since we aim to draw the viewer's focus effectively and to make the modulation as less noticeable as possible, the strength of modulation is determined to raise of target's saliency as well as to maintain the similarity to the original image. Moreover, we introduce a *pre-enhancement* step to accelerate computation required for the modulation. Due to inconsistency of modulation across the video, the appearance of objects may fluctuate between frames, which is commonly known as flicker. To reduce this effect, all frames are rendered by a post-processing operation, which results in smoothing the modulations over frames. Experiments

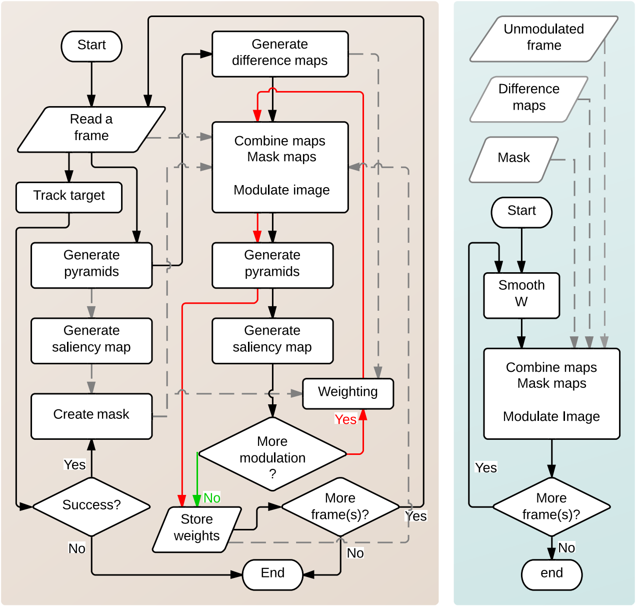


Fig. 1. Flowchart of process. Our method consists of two main parts: the processing loop part (on the left) and the output loop part (on the right).

using subjects demonstrate that our method can increase the time of fixation on the target, and that the flicker is more suppressed for intensity modulation than color modulation.

2 System Overview

A video is decomposed into frames and processed sequentially in our method. Firstly, pyramids are generated in a frame, which participate in constructing a saliency map. Meanwhile, the TLD tracker gives a bounding box of the tracked object, and then we create an binary image highlighting the bounding box area. By multiplying this image with the saliency map, we create the grayscale mask for modulation. To simplify the system, we only focus on pixel-wise modulation, in which each pixel has no movement in space. Therefore, orientation and motion are not modulated in our method. To prevent unnatural flickers, enhancement of flicker is also excluded from our method. Therefore difference maps only for intensity, red-green opponency and blue-yellow opponency, are generated from pyramids.

A flowchart of the entire algorithm is illustrated in Fig. 1. The procedures on the left constitute the processing loop, which generates the values of modulation to be applied on each frame. The procedures on the right constitute the output loop, which smooths the values over the time and applies them on frames.

In the processing loop part, the difference maps are weighted, combined, and masked, resulting in 3 enhancement maps. To make a better use of the coherence between frames, and for fast computation, the initial difference map weights of each frame are passed from the previous frame. This step is called *pre-enhancement* in our method. The enhancement maps are converted to enhancement maps of hue, saturation, intensity and the image is also decomposed into the H, S, I channels. The modulation is executed through vector combination of each channel and its corresponding enhancement map. Then, after the conversion of the image back to RGB space, the current status is evaluated through the saliency map to have a decision: if the target region achieves the maximum of saliency in the image, the process terminates (illustrated by green arrow in Fig. 1); otherwise weights increments are calculated to promote boosting of this frame (illustrated by red arrow). Finally weights are saved before the next frame processing starts.

The output loop part is to established to solve the flickers due to unpredictable change of weights. The key idea here is the smoothing of weights over the time. To accelerate the speed of modulation, we use the cached pyramids and masks obtained from the processing loop parts. The output loop part executes after the processing loop part completes for a frame, but a delay of a few frames exists for the smoothing of weights over the time. Since the processing is currently not real time, the output loop part can also run independently when the processing loop part terminates.

3 Saliency Map Creation

Visual attention is human's ability of selecting a region in the visual field to reduce scene analysis. In order to quantify the human visual attention, we need a saliency map for each frame. Our saliency map is generated based on the bottom-up model proposed by Itti et al. [6] [7]. An input RGB frame is firstly split into 3 channels: r, g, b. Then 5 images are generated from the channels: $I = (r+g+b)/3, R = r-(g+b)/2, G = g-(r+b)/2, B = b-(r+g)/2,$ and $Y = r+g-2(|r-g|+b)$. All of the mono-color images are normalized by I and small values (< 0.1 for example) in them are set to zero. Next, Gaussian pyramids of each image are created: $I(\sigma), R(\sigma), G(\sigma), B(\sigma), Y(\sigma)$, where $\sigma \in [0, 8]$ represents the scale. For orientation detection, 4 Gabor pyramids $O(\sigma, \theta)$ are also created from I [2]. For detecting time-varied features, 4 motion pyramids $R(\sigma, \theta)$ are created from $O(\sigma, \theta)$ and its shifts according to the Reichardt model [12].

Next, feature maps are created through absolute across-scale difference of scales in each pyramid. Among them, the color opponent feature maps come from difference between mono-color pyramids. In order to highlight the most discriminative feature within each map, we have a normalization step upon each feature map. Here we employ the simple max-local normalization [6] to compute fast and preserve more features. The reason for this is that our modulation mask generation needs the detail of small-salient features in the map, which the iterative normalization cannot provide. After normalization, feature maps are added in the across-scale manner, resulting in conspicuity maps. Finally the normalized conspicuity maps are combined into a master saliency map $\mathcal{S}$. In our implementation saliency maps are linearly normalized to the fixed range [0,1], which guarantees that the pixel with value 1 is the maxima of saliency.

Differently from the model by Itti et al. [7], the feature maps of flicker and conspicuity map of flicker are removed in our implementation. Our intensive experiments suggested that the flicker detection module always makes modulated videos flickering, which plays a negative role in keeping the modulated video smooth. To make the video visually comfortable, we decided to remove the flicker module.

4 Generating Enhancement Maps

In order to quantify the modulation, we create a grayscale image, called *enhancement map* in this paper, holding the adjustment on pixels in every particular channel. In creating the enhancement maps, we start from creating a mask, which discriminates the area to raise saliency from the area to decrease saliency.

4.1 Target Filtering

By employing the TLD tracker [8], we can fetch a bounding box of a target. In processing each frame, we firstly create a raw binary image B, in which pixels inside the bounding box are set to 1, while those outside are 0. In order to exclude the background from raising saliency, the area of the target object should be refined. Since the saliency map always highlights moving objects, we make use of it to accurately extract the target. The final mask is generated by:

$$M = \mathcal{S} \times (B * H),$$

where $\times$ denotes the pixel-wise product, $*$ denotes convolution, and H is a Gaussian kernel. Convolving B with H smooths the border of the bounding box, making the border less noticeable. Then, by means of the pixel-wise product, we highlight only highly salient pixels in the resulting mask. After these, the range of mask M is normalized to [0,1] to guarantee a constant maximum value for each frame across the video.

4.2 Extracting Center-Surround Difference

Raw features are extracted through center-surround differences between different scales in a pyramid. This procedure is similar to generating feature maps in the saliency model [7]. Here we name an image holding raw features as a *difference map*. To modulate intensity and color modules, their corresponding difference maps are generated through subtracting surrounding scales from centre scales:

$$D_I(c, s) = I(c) \ominus I(s),$$

$$D_{RG}(c,s) = (R(c) - G(c)) \ominus (G(s) - R(s)),$$

$$D_{BY}(c,s) = (B(c) - Y(c)) \ominus (Y(s) - B(s)),$$

where c, s denote scales, $c \in \{0,1,2\}, s = c + \delta, \delta \in \{3,4\}$.They are not same as feature maps: firstly, in order to keep the peak/valley property of features, difference maps are signed; secondly, scales enrolled (starting from 0) here are lower than those in feature maps (starting from 2). This is beacuse we found that lower scales work in more detail, producing more visually comfortable results.

4.3 Weighting the Difference Maps

To effectively navigate the gaze in any circumstances, saliency on the target should be enhanced and, at the same time, saliency outside the target (in the background) needs decreasing. To decrease saliency, we can subtract difference maps from the background to suppress the discriminative features. For each difference map D (including $D_I(c,s), D_{RG}(c,s), D_{BY}(c,s)$), we denote the maps in charge of increasing and decreasing saliency respectively as D_1 and D_2:

$$D_1 = D \times M,\ D_2 = D \times (1 - M).$$

In a channel, each difference map contains features on its own center-surround scales. In some scales the features are homogenous between the target and the background, while in the other scales they are discriminative. To modulate saliency as effectively as possible, we let the discriminative scales contribute more to our work. This is achieved through assigning weights to maps. We denote the mean and the standard deviation for peak values in the absolute map $|D_1|$ by $\hat{\mu}_1$ and $\hat{\sigma}_1$, and those in $|D_2|$ by $\hat{\mu}_2, \hat{\sigma}_2$. Then the weights for D_1 and D_2 are designed respectively as:

$$w_1 = \hat{\sigma}_1 e^{(\hat{\mu}_1 - \hat{\mu}_2)},\ w_2 = \hat{\sigma}_2 e^{(\hat{\mu}_2 - \hat{\mu}_1)}.$$

We take only the absolute peaks in calculation because centre-surround features only exist in a local peak or valley. A weight is large only when the mean value of features in its corresponding region (target/background) is larger than that in another region, and when there is a large discrimination in its region.

4.4 Enhancement Maps

An enhancement map is the across-scale combination of weighted difference maps. The difference maps are enlarged to the size of original image before combination. In a channel, let E_1 be an enhancement map in charge of increasing saliency, and E_2 be that in charge of decreasing saliency:

$$E_1 = \bigoplus_{c=0}^{2} \bigoplus_{s=c+3}^{c+4} w_1(c,s) n(D_1(c,s)),$$

$$E_2 = \bigoplus_{c=0}^{2} \bigoplus_{s=c+3}^{c+4} w_2(c,s)n(D_2(c,s)),$$

where $n(\cdot)$ is a rescaling operation to fit the map to [-1, 1]: $n(A) = \frac{A}{\max|A|}$.

Then a master enhancement map can be created from E_1 and E_2:

$$E = \alpha[n(E_1) - \beta_d n(E_2)],$$

where α is a rate controlling the strength of each time of modulation, and β_d is a ratio constraining the modulation on the background. To maintain the same visual perception even after the modulation, pixels in the background (which have a large population) are modulated as slightly as possible. Since the size of the tracked target is almost fixed throughout the video, the value of β_d is initialized only by the portion of the target area to the background area. In most cases β_d becomes a small value between [0,1] due to the area of the background being much larger than that of the target. We remark that when areas of the target and the background are equal with each other, β_d becomes 1 to guarantee the same speed in modulating two parts.

To make the target getting close to the saliency maxima, α is proportional to the difference between the peak saliency in the target area and 1:

$$\alpha = \begin{cases} k[1 - \max(\mathcal{S} \times B)] & \text{for intensity,} \\ k\beta_c[1 - \max(\mathcal{S} \times B)] & \text{for RG and BY,} \end{cases}$$

where the coefficient k constrains the modulation speed, and β_c is the ratio constraining the color modulation. β_c is also initialized during processing at the first frame: it becomes a large value only when the target area is both brighter and more saturated than the background. The reason for this is that we only execute strong color modulation on vivid objects to prevent unnatural appearances of other objects.

4.5 Weight Management

To save the computational cost, we take over the modulation at a frame t to that at its next frame $t+1$. Since features in even nearby frames have differences, directly applying enhancement maps of frame t to $t+1$ will cause blurring. Therefore, we take only the weights at t and apply them to difference maps at $t+1$. The step of constructing the enhancement map in this way is named *pre-enhancement*.

We provide each difference map D^t with a pre-enhancement weight W^t (where t denotes a frame index). For the first frame, we set all $W_1^1(c,s) \leftarrow 0$ and $W_2^1(c,s) \leftarrow 0$. We define increments ΔW by the products of all multipliers on difference maps in the construction of enhancement maps:

$$\Delta W_1^t(c,s) = \alpha^t \frac{1}{\max|D_1^t(c,s)|} \frac{1}{\max|E_1^t|} w_1^t(c,s),$$

$$\varDelta W_2^t(c,s) = \alpha^t \beta_d \frac{1}{\max|D_1^t(c,s)|} \frac{1}{\max|E_1^t|} w_1^t(c,s).$$

Then all W^t are adjusted by adding the increments (subscripts 1, 2 and parameters c, s are eliminated):

$$W^t \leftarrow W^t + \varDelta W^t.$$

Except for the first frame, we firstly construct pre-enhancement maps before calculating new weights:

$$E_0^t = \bigoplus_{c=c_1}^{c_3} \bigoplus_{s=c+\delta_1}^{c+\delta_2} [W_1^t(c,s) D_1^t(c,s) - W_2^t D_2^t(c,s)].$$

Then we apply the pre-enhancement maps to the image and evaluate the saliency. New weights are calculated only when the peak saliency of target is less than 1. It saves the computation when the predicted weights meet the required modulation of this frame. In most cases, the weights $W_1^1(c,s)$ and $W_2^1(c,s)$ increase frame after frame to raise the saliency of the target to the maximum. However, when the stimuli in the background is weakened, the required modulation for target decreases. In order to make W^t convergent, we multiply the weights with a coefficient γ^t between $[0,1]$ before applying the pre-enhancement of the next frame (subscripts 1, 2 and parameters c, s are omitted):

$$W^{t+1} = \gamma^t W^t.$$

γ^t has a small value when the image differs significantly from the original image, while it returns to 1 when no modulation is applied. Therefore γ^t and α^t always promote the modulation in two directions: γ^t aims to make the modulated image similar to the original image, while α^t encourages the image to have more modulation. They cooperate to get W^t fixed when the target region in the original image has already the saliency maxima. In all other cases, they cooperate to make the modulation as effective and un-noticeable as possible.

In the output loop part of our method, all frames can be rendered according to the weights generated in the processing loop part. To reduce the flicker resulted from the inconsistency of frame modulation, before the rendering, each weight is averaged with its neighboring values in the temporal domain:

$$\overline{W}^t = \frac{1}{2T+1} \sum_{\tau=t-T}^{t+T} W^\tau,$$

where $(2T+1)$ is the number of participating frames in an operation. After this operation, modulations on contiguous frames become familiar, thus flickering from the modulation are greatly reduced. More frames participating in an averaging operation can produce more visually comfortable result, but also increase the delay of output. In our experiments the choice of $(2T+1) = 9$ produced good results.

5 Saliency Modulation in the HSI Space

To achieve a natural appearance of the modulated image, enhancement maps are applied to the image in the HSI color space. In this space, a color value consists of hue θ, saturation ρ, and intensity i. Every RGB pixel of the original image needs to be converted to the HSI space for modulation [10].

5.1 Intensity Modulation

Since the intensity enhancement map E_I is created from the intensity channel of each pixel, we are able to modulate the intensity individually. We simply combine every value in the map E_I (denoted by e_I) to its corresponding intensity value of the pixel:

$$i \leftarrow i + e_I.$$

5.2 Color Opponency Modulation

After intensity modulation, the intensity of each pixel is fixed. The color opponency modulation of each pixel works on the pixel's chromatic plane (which is a rounded horizontal profile with a constant intensity in the HSI space). We denote an individual value in E_{RG} by e_{RG}, and that in E_{BY} by e_{BY}. For each pixel, we project values of both e_{RG} and e_{BY} as the magnitude of vectors in the unit circle of a chromatic plane (Fig. 2a). The directions of vectors follow the directions of colors that win in the opponency. We obtain a vector $\boldsymbol{e}_C$ for each pixel by combining e_{RG} and e_{BY}:

$$\boldsymbol{e}_C = \frac{\sqrt{3}}{2} e_{RG}\ \boldsymbol{u}_0 + (\frac{1}{2}|e_{RG}| - e_{BY})\ \boldsymbol{v}_0,$$

where $\boldsymbol{u}_0$ is the unit vector with hue $\theta_u = -\frac{\pi}{6}$ (along the R-G opponent axis), and $\boldsymbol{v}_0$ is the unit vector with hue $\theta_v = \frac{\pi}{3}$ (along the B-Y opponent axis).

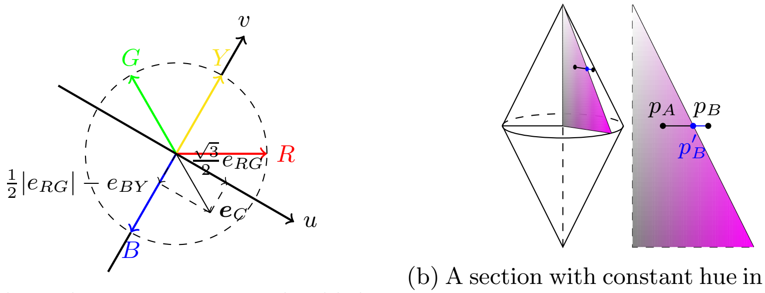
$\boldsymbol{e}_C$ can then be converted to the polar coordinate system: $\boldsymbol{e}_C = (e_\theta, e_\rho)$, where the magnitude of e_θ is the modulation for saturation and angle e_ρ is that for hue. Now for each pixel, the color modulation is carried out through the vector addition:

$$\theta \leftarrow \theta + e_\theta,\ \ \rho \leftarrow \rho + e_\rho.$$

The modulated image is then converted back to RGB space.

5.3 Data Correction

Because the HSI space is an irregular double-cone space, raising the saturation or intensity of a pixel may result in an out-of-range value (as it shown in Fig. 2b). Simply constraining θ, ρ, i value within [0,1] may shift the hue value after the conversion to the RGB space. Due to the irregular surface of the space, formulating the surface with H, S, I values is hard and thus using a convex linear combination is not appropriate. Our solution to this problem is to fix



(a) The enhancement maps should be converted to vector $\boldsymbol{e}_C$.

(b) A section with constant hue in the HSI space. In the section, p_A is enhanced to p_B, but should be corrected to p'_B.

Fig. 2. Conversion of enhancement maps and illustration of the 'out of range' problem

the maximum out-of-range channel (either r, g, or b) to 1, while keeping h and i unchanged, and then to find the maximum admissible ρ inversely. In this subsection only, we denote the individual pixel values of r, g, b by non-italic r, g, b to differentiate the notation of the entire image. When r is out-of-range and also the largest in $\mathsf{r}, \mathsf{g}, \mathsf{b}$, we set r to 1. The corrected g and b values are:

$$\mathsf{g} = \frac{\sqrt{3}}{2}(1-i)\tan\theta + \frac{3}{2}i - \frac{1}{2}, \ \mathsf{b} = -\frac{\sqrt{3}}{2}(1-i)\tan\theta + \frac{3}{2}i - \frac{1}{2}.$$

Similarly, we can apply this strategy to other cases where g or b is out-of-range. When g gets out of range, g is set to 1, and

$$\mathsf{r} = i - 2(1-i)\frac{\cos\theta}{\cos\theta - \sqrt{3}\sin\theta}, \ \mathsf{b} = 2i + 2(1-i)\frac{\cos\theta}{\cos\theta - \sqrt{3}\sin\theta} - 1;$$

when b is out-of-range, we set b to 1, and

$$\mathsf{r} = i - 2(1-i)\frac{\cos\theta}{\cos\theta + \sqrt{3}\sin\theta}, \ \mathsf{g} = 2i + 2(1-i)\frac{\cos\theta}{\cos\theta + \sqrt{3}\sin\theta} - 1.$$

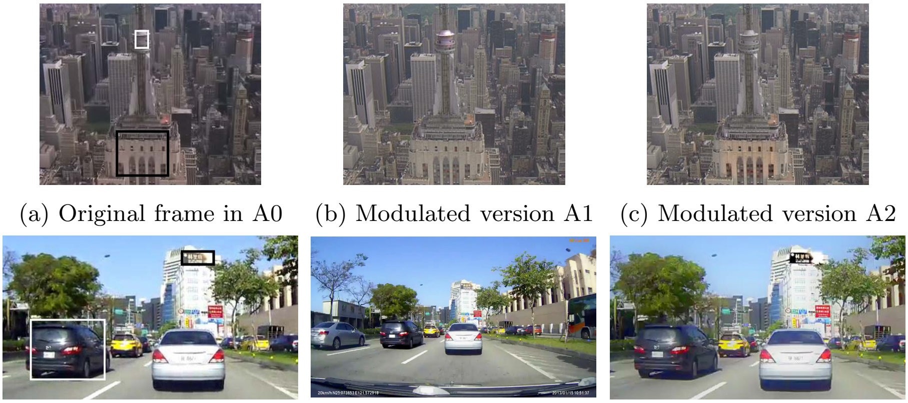
6 Experiments

6.1 Experiment Design and Preparation

We implemented our method in MATLAB, and prepared 2 original video clips (with names given as A0[1], B0[2]) for processing. Both of them were single shots with 30fps. For each clip we defined 2 completely different bounding boxes for the TLD tracker, to obtain 2 versions of modulated video (one version is named A1, B1, and the other version is A2, B2). Some snapshots of each video clip are shown in Fig. 3 The specifications of each video and bounding boxes are illustrated in Table 1. The relative area of the target is computed by the proportion of the target area to the image area.

[1] Source: <ftp://ftp.tnt.uni-hannover.de/pub/svc/testsequences/>

[2] Source: <http://www.youtube.com/watch?v=aU5Hq_Kz7n0>



(a) Original frame in A0 (b) Modulated version A1 (c) Modulated version A2

(d) Original frame in B0 (e) Modulated version B1 (f) Modulated version B2

Fig. 3. Snapshots of original and modulated videos. The white bounding boxes indicate the target regions to be modulated in A1/B1, while the black boxes corresponds to those in A2/B2.

Table 1. Specification of clips and targets

Resource	Resolution	Length	Proportion of area	
		(frames)	1	2
A	352×288	300	0.0056	0.0576
B	852×480	300	0.0263	0.0021

In total, 6 videos (including original and modulated versions) were used for each subject. Each video was equally treated and played once for each subject. 15 subjects participated in our experiments. We divided 15 subjects into 3 groups. For different groups, videos were shown in a different order, and videos of the same content were displayed alternatively. These were designed to reduce the impact of human high-level vision (gaze movements directed by thoughts for example). We also forced the subjects to reset their fixation points between videos to eliminate the effect of the previous content to the next one. After watching videos, each subject reported whether or not each video he/she watched was modulated, and whether he/she observed flicker in each video.

Table 2. Rates of successfully directed samples before and after modulation

Video	A1	A2	B1	B2
Original	0.0188	0.0350	0.0989	0.0076
Modulated	0.0813	0.1020	0.1497	0.0594

During the experiments, a Tobii TX300[3] eye tracker was employed to track the fixation point of the subjects. We used only the first sample in the duration of each frame as the fixation position in this frame.

6.2 Experimental Results

Over an entire video, by taking the rate of the number of fixation points falling into the bounding box of each frame to the number of valid frames, we computed the rate of successfully directed fixations. The rates of fixations before and after modulation for each modulated version are shown in Table 2.

Here we can see for all the modulation on different contents, the rates of fixations on the targets in modulated videos all greatly increased compared to those in the original videos. This indicates that our method successfully drew the gaze to given targets. However, the rate for modulated videos itself did not achieve a large value. This is because human gaze is always wandering over the image and it is hard to force someone always to look at the target over the entire video. Moreover, the area of the target affects the success of modulation. We can see in the same content, a larger area has a high probability of fixation falling into it, while a small area is hard to attract gaze.

To analyze the tracked fixations in more detail, we calculated the relative distances from fixations to the bounding box in each frame. Then we created a grid of 2-D containers and binned all the distances into the grid, resulting in a target-centered 3-D histogram. Among all the containers, the one at (0,0) bins all the data of in-bounding-box fixation, while the other containers bin the fixation in their particular positions to the target. We then plotted 2-D heat-maps where the color of each grid represents the height of each container. Fig. 4 shows the heat-map for modulation version B1, which had the lowest growth (51.41%) of the falling-in rate. We can observe that the target (the black van) was originally salient in a large bunch of gaze attractions. After modulation, the accumulation on container (1,0) was diluted and some energy was transferred to (0,0).

Statistics of questionnaire are shown in Fig. 5. For each video, the blue bar illustrates the ratio of subjects who felt the video was modulated, and the red bar shows the ratio of subjects who observed flicker. For all contents, the feeling of modulation on modulated versions was higher than that on the original version. This means that our modulation was easy to be noticed. For content B, we can see that the observation of flicker was almost proportional to the growth of falling-in rates in Table 2. Although the feeling of flicker should be eliminated, we have to admit that flicker does attract gaze. However, for the feeling of flicker, A0 received a higher mark than its modulated versions. No subject observed flicker in A1 although in this video both color opponency modules were greatly modulated. This may come from the quality of video and the moving characteristic of the target.

To investigate the relationship between the perceived flicker and the strength of modulation, we computed the mean and the standard deviation of difference

[3] Product information: `http://www.tobii.com/en/eye-tracking-research/global/products/hardware/tobii-tx300-eye-tracker/`

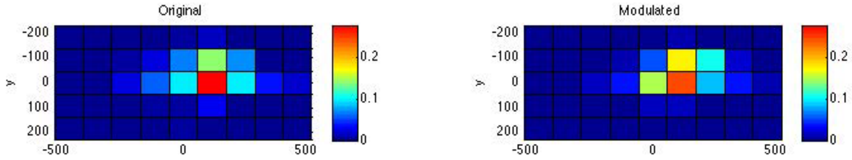


Fig. 4. Heat-map of containers of video B0 vs. B1

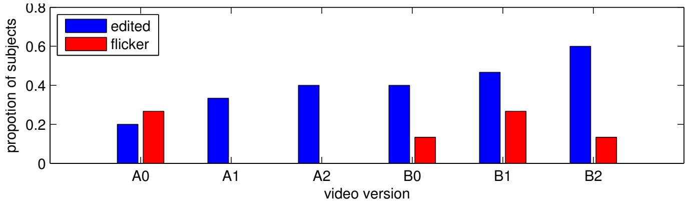


Fig. 5. Subjective judgements of modulation and flicker

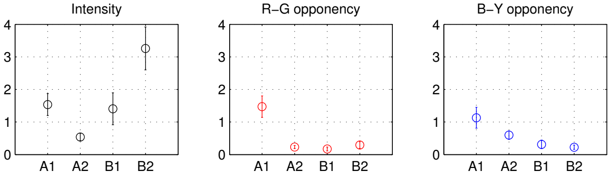


Fig. 6. Mean and standard deviation of weights of modules

map weights for each module (shown in Figure 6).In this figure, a larger mean indicates that the clip is more strongly modulated, while a larger standard deviation indicates greater fluctuation of modulation is caused across frames. We observe that a strong modulation in one of the color opponency modules might be the cause of perceived flicker. Especially for video B2, although the modulation on intensity was great, the colors were adjusted very slightly, which made it less likely to be marked as 'flickering'. Reason for this may be human vision is more sensitive to color opponent contrast than intensity contrast.

7 Conclusion

We have successfully navigated human gaze to our given targets in videos, by modulating saliency of videos under the HSI space. For every frame the modulation was simultaneously carried out for intensity, red-green opponency, and blue-yellow opponency. Given the target tracked by the TLD tracker, saliency in the target region becomes boosted, while saliency in the background becomes reduced. This is evaluated with the help of a saliency map. Moreover, our proposed method employs the *pre-enhancement* step for computation efficiency, as well as a post-processing module for the prevention of flickers. Experimental results showed that this method can effectively draw attention of subjects to the predefined targets. We discovered that subjects were more likely to notice the cue of modulation when exposed to greatly modulated videos. The observation of flicker was likely to be stronger when color opponency channels are greatly modulated. To reduce the flicker effect is left for future work on this topic. In our experiments, the size of bounding box was always maintained, and the tracker might sometimes lose the target. Therefore, to find a more robust way of target tracking, with an ability of zooming the bounding box with the target, is another piece of future work. Additionally, a real-time modulation is also promising development for the need of on-line processing.

References

1. Bailey, R., McNamara, A., Sudarsanam, N., Grimm, C.: Subtle gaze direction. ACM Trans. on Graphics 28(4), 1–14 (2009)
2. Greenspan, H., Belongie, S., Goodman, R., Perona, P., Rakshit, S., Anderson, C.: Overcomplete steerable pyramid filters and rotation invariance. In: Proc. of IEEE Conf. on CVPR, pp. 222–228 (1994)
3. Hagiwara, A., Sugimoto, A., Kawamoto, K.: Saliency-based image editing for guiding visual attention. In: Proc. of the 1st Int. Workshop on Pervasive Eye Tracking & Mobile Eye-Based Interaction, pp. 43–48 (2011)
4. Harel, J., Koch, C., Perona, P.: Graph-based visual saliency. Advances in Neural Information Processing Systems 19, 545–552 (2007)
5. Huang, C., Liu, Q., Yu, S.: Regions of interest extraction from color image based on visual saliency. The Journal of Supercomputing 58(1), 20–33 (2010)
6. Itti, L., Koch, C., Niebur, E.: A Model of saliency-based visual attention for rapid scene analysis. IEEE Trans. on PAMI 20(11), 1254–1259 (1998)
7. Itti, L., Dhavale, N., Pighin, F.: Realistic avatar eye and head animation using a neurobiological model of visual attention. In: Proc. of SPIE. Applications and Science of Neural Networks, Fuzzy Systems, and Evolutionary Computation VI, vol. 5200, pp. 64–78 (2004)
8. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-Learning-Detection. IEEE Trans. on PAMI 6(1), 1–14 (2011)
9. Koch, C., Ullman, S.: Shifts in selective visual attention: towards the underlying neural circuitry. Matters of Intelligence 188, 115–141 (1987)
10. Ledley, R., Buas, M., Golab, T.: Fundamentals of true-color image processing. In: Proc. of the 10th ICPR, pp. 791–795 (1990)
11. Mendez, E., Feiner, S., Schmalstieg, D.: Focus and context in mixed reality by modulating first order salient features. In: Proc. of the 10th Int. Symposium on Smart Graphics, pp. 232–243 (2010)
12. Reichardt, W.: Evaluation of optical motion information by movement detectors. Journal of comparative physiology. A, Sensory, Neural, and Behavioral Physiology 161(4), 533–547 (1987)
13. Veas, E., Mendez, E., Feiner, S., Schmalstieg, D.: Directing attention and influencing memory with visual saliency modulation. In: Proc. of the SIGCHI Conf. on Human Factors in Computing Systems, pp. 1471–1480 (2011)