Appearance-based Eye Control System by Manifold Learning

Ke Liang, Youssef Chahir, Michele Molina, Charles Tijus, François Jouen

Appearance-based Eye Control System by Manifold Learning

Ke Liang[1], Youssef Chahir[1], Michèle Molina[3], Charles Tijus[1] and François Jouen[1]
[1]*CHArt Laboratory, EPHE Paris, 4-14 rue Ferrus 75014 Paris, France*
[2]*Computer Science Department, GREYC-UMR CNRS 6072, University of Caen, France*
[3]*PALM Laboratory EA 4649, University of Caen, France*
ke.liang@etu.ephe.fr, {youssef.chahir, michele.molina}@unicaen.fr, tijus@univ-paris8.fr, Francois.Jouen@ephe.sorbonne.fr

Keywords: Appearance eye descriptor, Appearance-based method, Manifold learning, Spectral clustering, Human-computer interaction.

Abstract: Eye-movements are increasingly employed to study usability issues in HCI (Human-Computer Interacetion) contexts. In this paper we introduce our appearance-based eye control system which utilizes 5 specific eye movements, such as closed-eye movement and eye movements with gaze fixation at the positions (up, down, right, left) for HCI applications. In order to measure these eye movements, we employ a fast appeance-based gaze tracking method with manifold learning technique. First we propose to concatenate local eye appearance Center-Symmetric Local Binary Pattern(CS-LBP) descriptor for each subregion of eye image to form an eye appearance feature vector. The calibration phase is then introduced to construct a trainning samples by spectral clustering. After that, Laplacian Eigenmaps will be applied to the trainning set and unseen input together to get the structure of eye manifolds. Finally we can infer the eye movement of the new input by its distances with the clusters in the trainning set. Experimental results demonstrate that our system with quick 4-points calibration not only can reduce the run-time cost, but also provide another way to mesure eye movements without mesuring gaze coordinates to a HCI application such as our eye control system.

1 INTRODUCTION

As gaze tracking technology improves in the last 30 years, gaze tracker offers a powerful tool for diverse study fields, in particular eye movement analysis and human-computer interaction (HCI), such as eye control system or eye-gaze communication system. Eye control helps users with significant physical disabilities to communicate, interact and to control computer functions using their eyes (Jacob & Karn, 2003). Eye movements can be measured and used to enable an individual actually to interact with an interface. For example, users could position a cursor by simply looking at where they want it to go, or click an icon by gazing at it for a certain amount of time or by blinking.

Nowadays most commercial gaze trackers use *feature-based* method to estimate gaze coordinates, which relies on video-based pupil detection and the reflection of infra-red LEDs. In general, there are two principal methods: 1) Pupil-Corneal Reflection(P-CR) method (Morimoto et al., 2000)(Zhu and Ji, 2007), and 2) 3D model based method(Shih and Liu, 2004)(Wang et al., 2005). IR light and extraction of pupil and iris are important for these *feature-based* methods, and the calibration of cameras and geometry data of system is also required.

Appearance-based methods do not explicitly extract features like the *feature-based* method, but rather use the cropped eye images as input with the intention of mapping these directly to gaze coordinates(Hansen and Ji, 2010). The advantage is that they do not require calibration of cameras and geometry data like *feature-based* method. Moreover, they can be less expensive in materials than *feature-based* method since they don't have to work on the same quality images like *feature-based* method does. But they still need a relatively high number of calibration points to get accurate precision. Different works can be seen in multilayer networks(Baluja and Pomerleau, 1994)(Stiefelhagen et al., 1997)(XU et al., 1998), or Gaussian process(Nguyen et al., 2009)(Williams et al., 2006), or manifold learning(Martinez et al., 2012)(Tan et al., 2002). Williams *et al.* (Williams et al., 2006) introduces the sparse, semi-supervised Gaussian Process (S^3GP) to learn mappings from semi-supervised training sets. Fukuda *et al.* (Fukuda et al., 2011) propose a gaze-estimation method that

uses both image processing and geometrical processing to reduce various kinds of noise in low-resolution eye-images and thereby achieve relatively high accuracy of gaze estimation.

Manifold learning is widely applied to solve many problems in computer vision, in pattern recognition etc (Lee and Kriegman, 2005)(Rahimi et al., 2005) (Weinberger and Saul, 2006)(Zhang et al., 2004). Manifold learning, often also referred to as nonlinear dimensionality reduction, is also one of the approaches applied in appearance-based gaze tracking(Tan et al., 2002), and one of the reason to apply manifold learning techiques is to reduce computational costs. Manifold learnig means the process of estimating a low-dimensional structure which underlies a collection of high-dimensional data, also preserves characteristic properties of the set of high-dimensional data. Here we are interested in the case where the mainifold lies in a high dimensional space $\mathbb{R}^D$, but will be homeomorphic with a low dimensional space $\mathbb{R}^d$ ($d < D$). Laplacian eigenmaps(Belkin and Niyogi, 2001)(Belkin and Niyogi, 2003) most faithfully preserves proximity relations of a high-dimensional non-linear data set in the low dimensional space, by using spectral graph technique.

Here our emphasis is on creating a pratical, real-time eye control system with our appearance-based method. Our contributions are:

- A subregion CS-LBP concatenated histogram is used as eye appearance feature which not only reduce the dimension of raw images, but also can be robust against the changes in illumination.
- During the calibration phase, we use spectral clustering to build on-line the subject's eye movements model by selecting and trainning a limited number of eye feature samples.
- To infer its movement for an unseen eye input, Laplacian Eigenmaps is introduced to analyse the similarities and the manifold structure for the training samples and the unseen input.
- The system requires only a remote webcam and without IR light.

The rest of the paper is organized as follows. Section2 describes the eye manifold learning to the proposed eye feature. Section 3 presents a global view of our eye control system. Section 4 shows the experimental setup and results. Finally section 5 concludes the paper.

2 EYE APPEARANCE MANIFOLD LEARNING

2.1 Eye appearance descriptor

Let an eye image I be a two-dimensional M by N array of intensity values, and it may also be considered as a vector of dimension $M \times N$. The proposed gaze tracker captures left and right eyes together and combines them into one image. Our eye image of size 160 by 40 becomes a vector of dimension 6400. Appearance-based gaze tracking methods mostly rely on the eye images as input. Extracting eye appearance descriptor not only helps to reduce the dimension of eye images, but also preserves the feature and variation of eye movemnts.

There exist a number of eye appearance feautre extraction methods for gaze tracking system, like multi-level HOG (Martinez et al., 2012), eigeneyes by PCA (Noris et al., 2008), and subregions feature vector (Lu et al., 2011). Lu et al. have proven the efficiency of using 15D subregions feature vector in (Lu et al., 2011). To compute this feature vector, the eye image I_i is divided into $N^{'}$ subregions of size $w \times h$. Let S_j denote the sum of pixel intensities in $j-$th subregion, then feature vector X_i of the image I_i is represented by

$$X_i = \frac{[S_1, S_2, ..., S_j]}{\sum S_j} \qquad j \in N^{'} \qquad (1)$$

Here we introduce our subregion methods with Center-Symmetric Local Binary Pattern (CS-LBP) to calculate low dimensional feature vector for raw eye image content. Local Binary Pattern (LBP) operator has been highly successful for various computer vision problems such as face recognition, texture classification etc. The histogram of the binary patterns computed over a region is used for feature vector. The operator describes each pixel by the relative graylevels of its neighboring pixels. If the graylevel of the neighboring pixel is higher or equal, the value is set to one, otherwise to zero.

We calculate the CS-LBP(Heikkilä et al., 2009) histogram, which is a new texture feature based on the LBP operator, for each subregion in Fig.1(a) and concatenate them to form the eye appearance feature vector. Instead of describing a center pixel by comparing its neighboring pixels with it in LBP, CS-LBP compares the center-symmetric pairs of pixels in Fig.1(b).

The CS-LBP value of a center pixel in position (x,y) is calculated as follows:

$$CS-LBP_{R,N,T}(x,y) = \sum_{i=0}^{N/2-1} s(n_i - n_{i+(N/2)})$$

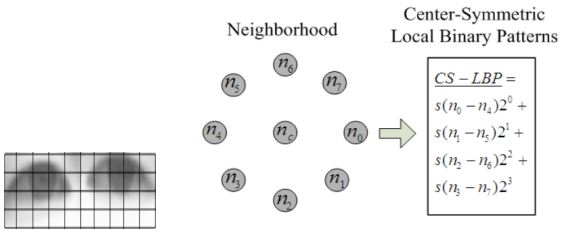


Figure 1: a) 40 subregions of an eye image sample b) CS-LBP for a neighborhood of eight pixels.

where $s(t) = \begin{cases} 1 & t > T \\ 0 & else \end{cases}$, n_i and $n_{i+(N/2)}$ are the gray values of center-symmetric pairs of pixels of N equally spaced pixels on a circle with radius R , and the threshold T is a small value. From this equation, the value of CS-LBP may be any integer from 0 to $2^{N/2}-1$, and the histogram dimension will be $2^{N/2}$. CS-LBP is fast to compute and its histogram has been proven to be robust against the changes in illumination as a texture descriptor(Heikkilä et al., 2009).

2.2 Spectral clustering

Graph Laplacians are the main tools in spectral graph theory. Here we focus on two kinds of graph Laplacian:

- Unnormalized graph Laplacian.

$$L_{un} = D - W$$

where W is the symmetric weight matrix with positive entries for edge weights between vertices. If $w_{ij} = 0$, then vertices i and j are not connected. D is the degree matrix: $d_{ii} = \sum_{j=1}^{n} w_{ij}$ and $d_{ij} = 0$ $\forall i \neq j$.

- Normalized graph Laplacian.

$$L_{sym} = D^{-1/2} L_{un} D^{1/2} = I - D^{-1/2} W D^{1/2}$$

$$L_{normalized} = D^{-1} L_{un} = I - D^{-1} W = I - L_{rw}$$

Where L_{sym} is a symmetric matrix, and L_{rw} is closely related to a random walk. There are 3 properties:

1) λ is an eigenvalue of L_{rw} with eigenvector v if and only if λ and v solve the generalized eigenproblem $Lv = \lambda Dv$.
2) L_{rw} is positive semi-definite with the first engenvalue $\lambda_1 = 1$ and the constant one vector $\mathbf{1}$ the corresponding eigenvector.
3) All eigenvectors are real and it holds that: $1 = |\lambda_1| \geq |\lambda_2| \geq ... \geq |\lambda_n|$.

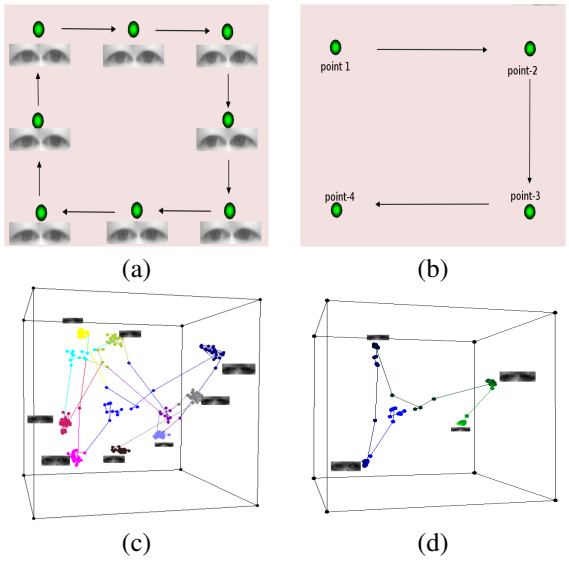


Figure 2: (a) Eye capture for 8 visual patterns in the screen (b) 4-points calibration (up-right, up-left, down-left, down-right) (c) Eye manifolds in the phase of 8-points calibration (d) Eye manifolds in the phase of 4-points calibration (up-right, up-left, down-left, down-right)

Laplacian eigenmaps use spectral graph technique to compute the low-dimensional representation of a high-dimensional non-linear data set, and they most faithfully preserves proximity relations, mapping nearby input patterns to nearby outputs. The algorithm of Laplacian eigenmaps has a similar structure as LLE. First, one constructs the symmetric undirected graph $G = (V, E)$, whose vertices represent input patterns and whose edges indicate neighborhood relations (in either direction). Second, one assigns positive weights W_{ij} to the edges of this graph; typically, the values of the weights are either chosen to be constant, say $W_{ij} = 1/k$, or a heat kernel, as $W_{ij} = exp(-\frac{||x_i - x_j||^2}{2l^2})$ where l is a scale parameter. In the third step of the algorithm, one obtains the embeddings $\psi_i \in \mathbb{R}^m$ by minimizing the cost function:

$$\mathcal{E}_L = \sum_{ij} \frac{W_{ij} ||\psi_i - \psi_j||^2}{\sqrt{D_{ii} D_{jj}}}$$

This cost function encourages nearby input patterns to be mapped to nearby outputs, with "nearness" measured by the weight matrix W. To compute the embeddings, we find the eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq ... \leq \lambda_n$ and eigenvectors $v_1, ..., v_n$ of the generalized eigenproblem: $Lv = \lambda Dv$. The embeddings $\Psi : \psi_i \rightarrow (v_1(i), ..., v_m(i))$.

Spectral clustering refers to a class of techniques which rely on the eigen-structure of a similarity matrix to partition points into disjoint clusters with points in the same cluster having high similarity and

points in different clusters having low similarity. We follow the works of (Shi and Malik, 2000). Their algorithm of spectral clustering computes the normalized graph Laplacian L_{rw}, and its first k generalized eigenvectors $v_1, ..., v_k$ as embeddings, and then utilise k-means to cluster the points.

From the section 2.1 we've introduced our subregion CS-LBP methods to extract the eye appearance feature descriptor. Here we'd like to at first obtain eye manifold distribution by using Laplacian Eigenmaps, and then we apply the normalized spectral clustering. The Fig.2 (a,b) shows eye samples of the subject's eye movements when the subject follows the visual pattern(green points) shown in the screen. The Fig.2 (c,d) demonstrate the distribution of embeddings in the subspace. (c) gives the distribution of a dataset of 240 points which contains only the eye samples on the 8 points in the screen , while (d) contains only 120 eye samples from the 4 points in the corner(up right, up left, down left, down right). For a given number C of visual patterns, generally we can get l clusters $U = \{U_1, U_2, ..., U_l\}$ associated with weights $W = \{w_1, w2, ..., w_l\}$ by the size of cluster, where $C \leq l < n$.

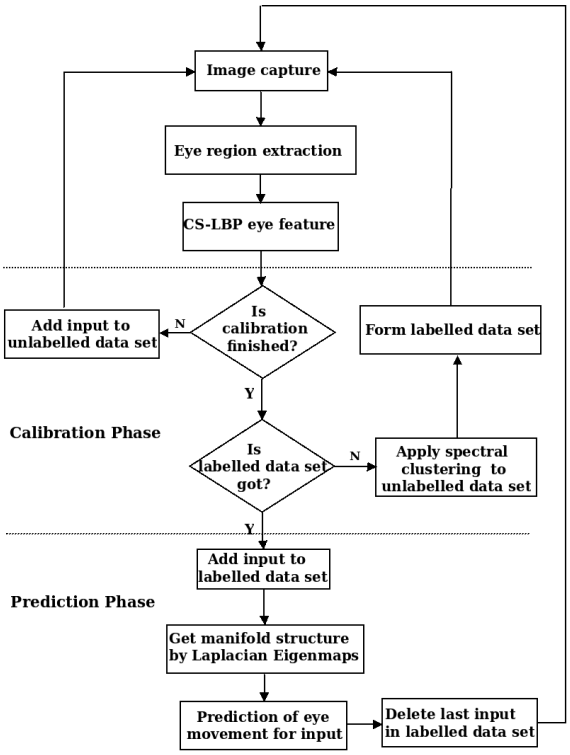


Figure 3: Flowchart of system

3 EYE CONTROL SYSTEM

In this section we introduce our eye control system which aims to recognize 5 specific eye movements, such as closed-eye movement and eye movements with gaze fixation at the up-right, up-left, down-right and down-left of the screen. The four eye gaze movements are used to select events and the closed-eye movements can be used as control signal like a 'click' of the mouse. The system has two components: calibration phase and prediction phase which are shown in Fig. 3.

3.1 Calibration Phase

As an appearance-based approach, the system needs an eye-gaze mapping calibration which can be considered as the collection and analysis phase of labelled and unlabelled eye data. This on-line calibration procedure aims to provide a model of the subject's eye movement in a given region (for example, the screen of laptop etc) with limited number of eye samples.

The system utilizes 4 points (up-right, up-left, down-right and down-left of the screen) as calibration points (Fig. 2b). In order to achieve an efficient calibration, we apply spectral clustering (Sec. 2.2) to all the unlabelled eye samples and get the clusters of their manifolds during the calibration procedure. A labelled data set then can be built by selecting the center samples of each clusters, and can be used as training model for a new inputs. The manifold structure of labelled data set might be shown in Fig. 2d.

3.2 Prediction Phase

The prediction phase executes a classification task for new eye feature inputs. In order to map feature vector $X \in \mathbb{R}^{M'}$ to gaze region output $Y \in \mathbb{R}$, which represents a set of categories, we have a trainning set $\mathcal{D} = \{(\mathcal{X}_i, \mathcal{Y}_i) | i = 1, 2, ..., n\}$, where $\mathcal{X}_i$ denotes an input vector of dimension M' and $\mathcal{Y}_i$ denotes the number of category, n is the number of observations. Given this trainning set $\mathcal{D}$, we wish to make predictions for the new inputs $\mathcal{X}'$ which is not in the training set.We apply Laplacian Eigenmaps to the trainning set and the new input together.

We propose a 24-points visual pattern scenario Fig. 4(a) in a given region which size is 754 $\times$ 519 pixels. With the screen in which pixel is 0.225mm/pixel, the real size is about 17cm $\times$ 11.6cm. The subject is asked to gaze each point one by one. Fig. 4(b,c,d) shows the distribution of 3D eye manifolds projected by laplacien eigenmaps. Notice that there are some degrees of similarity between the manifold surface and position plane of the 24 points. From

their manifold structure we can infer the eye movement of the new input by the distance of similarity with the training datas given a threshold T_{corner}.

For an input like eye blinkings or closed-eye movements, which are very different with the trainning samples, their manifold structure is completely changed as shown in Fig. 5f. The scale of trainning set becomes more smaller than the scales in the other case such as Fig. 5 (a,b,c,d), while the eye blinking image is added into the trainning set. Laplacian Eigenmaps helps to give prominence to the difference. We take advantage of this difference in scale to recognize eye blinkings by the distance of similarity with the training datas given a threshold T_{blinks}.

4 EXPERIMENTATION

This section evaluates our proposed methods presented in the previous sections. Our experimentation is tested on MacBook Pro 8,1 with Intel Core i5-2415M CPU. Microsoft LifeCam HD-5000 is used for image acquisition of gaze tracking system in the experiments. The USB color webcam captures 30 frames per second with a resolution of 640 $\times$ 480.

4.1 Eye detection and tracking

The distance between the subject and the camera is about 40 - 70 cm. To the entire RGB image captured from camera, we firstly use a face components detection model, which is based on Active Shape Model(T.F.Cootes et al., 1995), to localize the eye regions and the corners. We introduce then Lukas-Kanade method to track the corner points in the following frames. Finally we combine the left and right eye regions to the eye appearance pattern, which is converted to grayscale and used as the input data for gaze estimation process. The process of eye detection and tracking is shown in Fig. 6. The eye appearance pattern is an image of 160$\times$40. Fig. 7 shows eye samples of five subjects. As introduced in section 2.1, the pattern is divided into 40 subregions and we calculate CS-LBP histogram for each subregion. The size of the feature vector is 640.

4.2 Eye manifolds

In this section we evaluate eye manifold structure for different persons, and in different conditions.

Taking eye samples of five subjects such as Fig. 7 who follow the 16 points outside-round the screen, we can see that the distributions of their eye manifolds are relatively similar (Fig. 8), despite the difference

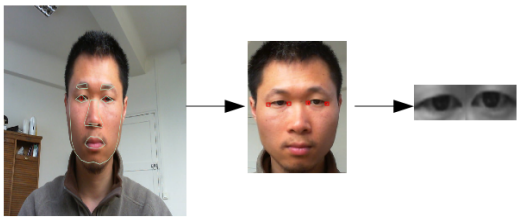


Figure 6: Eye localization and tracking

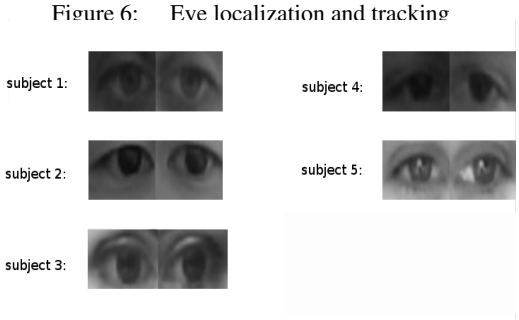
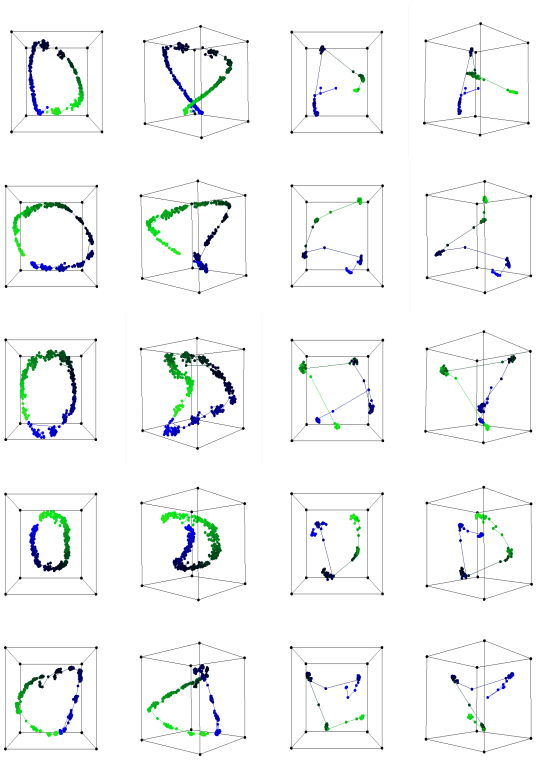
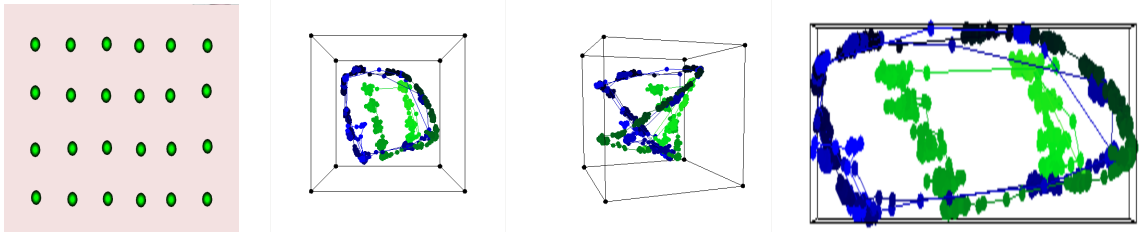


Figure 7: Five eye samples in different light condition. The subjects have head-free movement and the distance between the subject and the camera is about 40 - 70 cm.



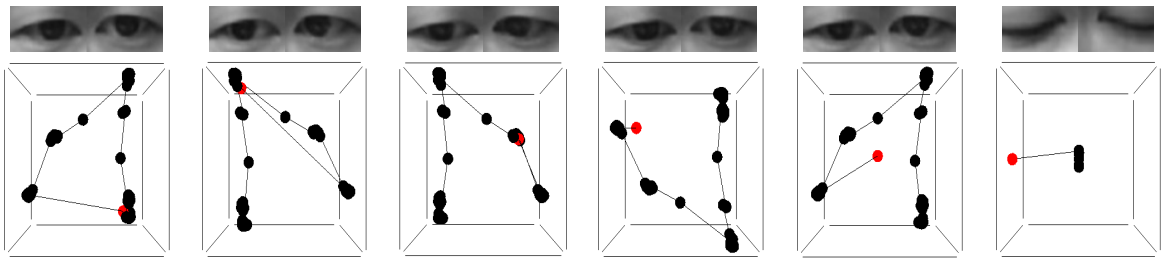
(a) (b) (c) (d)

Figure 8: Each line of figures represents the eye manifold distribution for each subject mentioned above. $\alpha = 0.8$ $l = 100$ a)b) 750 eye samples c)d) 120 eye samples



(a) (b) (c) (d)

Figure 4: Projection of 990 eye gaze samples on 24 points in the screen (a) by Laplacian Eigenmaps. b) c) 3D eye manifolds $e_i = \{v_i^1, v_i^2, v_i^3\}$ d) 3D eye manifolds $e_i = \{\lambda_i^1 v_i^1, \lambda_i^2 v_i^2, \lambda_i^3 v_i^3\}$.



(a) (b) (c) (d) (e) (f)

Figure 5: Projection of the trainning samples plus new eye input of different movements by by Laplacian Eigenmaps (l=700). (Red point represents the new input data, black points represents the trainning samples.)

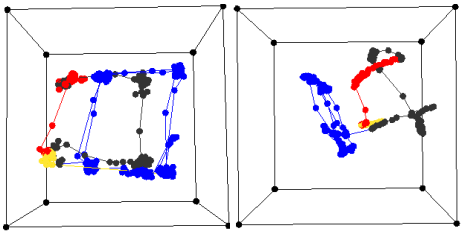
between their eyes' form. If taking eye samples at four points on the corners, the distibution shows the different clusters distinctly such as Fig. 8(c)(d). The curve in Fig. 13 shows the exponential growth of time cost for spectral clustering with the growth number of eye samples.

Here we analyse the structure of eye manifolds projected by laplacien eigenmaps in 2 different conditions as the illumination changes and the head movement. We also compare our subregion CS-LBP descriptor with the original subregion method, which calculates the feature vector as the equation(1) shown in Sec. 2.1.

- Illumination changes
  Here the subject follows the 4 points in the screen (Fig. 2b) 2 times within 20 seconds, while the indoor-illumination changes as Fig. 9. We extract the eye appearance descriptor from the 500 eye images by our proposed CS-LBP descriptor, alos by the original subregion descriptor as a comparative method. From the observations (Fig. 10) of projection by Laplacian Eigenmaps, CS-LBP descriptor gives the translation of eye movement stucture for the changes of illumination, while subregion descriptor is totally disturbed by the changes illumination.

- Head movements
  Different humans vary widely in the tendency to

Figure 9: Demonstration of the changes of illumination by 2 sample frames



(a) (b)

Figure 10: Comparison of using CS-LBP and subregion methods as eye descriptor in the condition of changes of illumination (500 eye samples). The different colors show the changes of illumination. a) $CS-LBP_{1,8,0.01}$ feature vector projected in 3D by Laplacian Eigenmaps(l = 10000). b) original subregion feature vector by Laplacian Eigenmaps($l = 700$).

move the head for a given amplitude of gaze shift. We are interested in the difference of eye manifold structure between a limited natural head movement as Fig. 11 and the movement keeping the head still. Here the subject is asked to follow a point which moves along the edge line of screen. The size of screen is 33cm $\times$ 22cm. The distance between the subject and the screen is about 60 cm. From the result as shown in Fig. 12, we can see that both the descriptors can keep the structure of eye movement while moving the head slightly, but the scale of structure changes.

4.3 Calibration phase and prediction phase

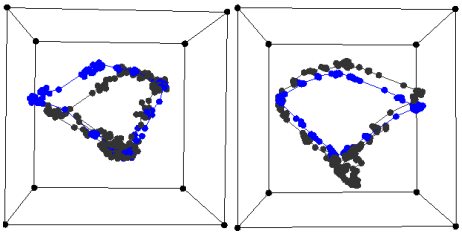
The calibration phase will vary depending on application. Generally the conventional calibration procedure makes each calibration point appear in the screen for one second, and the subject is expected to follow it. The 30Hz camera can capture 120 images for 4 senconds. So in this case we can get 120 eye samples for unlabelled data set.

But for our eye-control application, the calibration phase can be done more quickly et efficiently, because the position of the four calibration points are just at the corner of the screen. The more efficient way to calibrate is just that let the subject look at the corner by himself and the camera captures the images during the time. It might take one or two seconds and we can get 30 or 60 unlabelled eye samples.

In our experimentation, the prediction phase utilize a trainning set of 12 samples (3 samples for each corner). In order to predict an unseen input, we compute the 3D manifold for the set of 13 eye features (12 trainning samples + 1 new input) by Laplacian Eigenmaps and compare the scale of structure and their distances as shown in Fig 5. The thresholds T_{blinks} and $T_{corners}$ take empirical values as 2 and 900. The experimentation movie in the attachment shows that the system can predict the eye movements for four directions, at the same time, it is also able to recognize the closed-eye movements as well as eye blinkings(Fig. 14).

Fig. 13 demonstrates the consuming time of Laplacian Eigenmaps for different numbers of trainning samples. The consuming time for 60 eye samples is about 41 ms, and 137ms for 120 eye samples. So with 12 trainning samples provides a possiblity for the real-time application.

Figure 11: Demonstration of the free-head movement while the suject follows the points in the screen. The distance between the subject and the screen is about 60 cm.



(a) (b)

Figure 12: Comparison of using CS-LBP and subregion methods as eye descriptor for the head movement (520 eye samples). Blue points represent the stucture with fixed head and black points represent the structure with slight head movements. a) $CS-LBP_{1,8,0.01}$ feature vector projected in 3D by Laplacian Eigenmaps($l = 9000$). b) original subregion feature vector by Laplacian Eigenmaps($l = 900$).

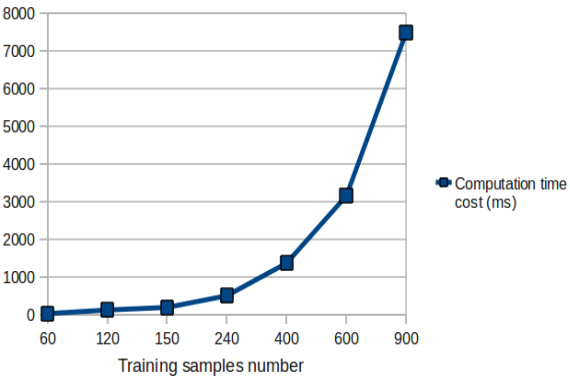


Figure 13: Consuming time of spectral clustering to different numbers of eye samples

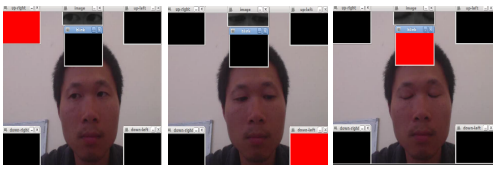


Figure 14: Demonstration of eye control system interface. Red labels present the result of prediction: up-left, down-right and closed eye movement

5 CONCLUSIONS

We presented our appearance-based eye movements tracker and the application for an eye control system. We used subregion CS-LBP concatenated histogram as eye appearance feature, which not only can reduce the dimensionality of eye images, but also can be robust againt the changes in illumination. Additionally, we introduced Laplacian Eigenmaps and spectral clustering which help to learn about the "manifold structure" of eye movement and give an efficient calibration phase. With limited number of trainning samples, the system can provides a quick prediction even when the number of calibration samples is limited. The efficiency and reasonable accuracy can help to provide a real-time application.

ACKNOWLEDGEMENTS

This work is supported by company UBIQUIET, and the French National Technology Research Agency (ANRT).

REFERENCES

Baluja, S. and Pomerleau, D. (1994). Non-intrusive gaze tracking using artificial neural networks. *Advances in Neural Information Processing Systems*.

Belkin, M. and Niyogi, P. (2001). Laplacian eigenmaps and spectral techniques for embedding and clustering. *NIPS*, 15(6):1373–1396.

Belkin, M. and Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput.*, 15(6):1373–1396.

Fukuda, T., Morimoto, K., and Yamana, H. (2011). Model-based eye-tracking method for low-resolution eye-images. *2nd Workshop on Eye Gaze in Intelligent Human Machine Interaction*.

Hansen, D. W. and Ji, Q. (2010). In the eye of the beholder: A survey of models for eyes and gaze. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 32(3):478–500.

Heikkilä, M., Pietikäinen, M., and Schmid, C. (2009). Description of interest regions with local binary patterns. *Pattern Recogn.*, 42(3):425–436.

Lee, K.-C. and Kriegman, D. (2005). Online learning of probabilistic appearance manifolds for video-based recognition and tracking. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 852–859, Washington, DC, USA. IEEE Computer Society.

Lu, F., Sugano, Y., Okabe, T., and Sato, Y. (2011). Inferring human gaze from appearance via adaptive linear regression. In Metaxas, D. N., Quan, L., Sanfeliu, A., and Gool, L. J. V., editors, *ICCV*, pages 153–160. IEEE.

Martinez, F., Carbonne, A., and Pissaloux, E. (2012). Gaze estimation using local features and non-linear regression. *ICIP(International Conference on Image Processing)*.

Morimoto, C. H., Koons, D., Amir, A., and Flickner, M. (2000). Pupil detection and tracking using multiple light sources. *Image and Vision Computing*, pages 331–335.

Nguyen, B. L., Chahir, Y., and Jouen, F. (2009). Eye gaze tracking. *RIVF '09*.

Noris, B., Benmachiche, K., and Billard, A. (2008). Calibration-free eye gaze direction detection with gaussian processes. *Proceedings of the International Conference on Computer Vision Theory and Application*.

Rahimi, A., Recht, B., and Darrell, T. (2005). Learning appearance manifolds from video. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 868–875, Washington, DC, USA. IEEE Computer Society.

Shi, J. and Malik, J. (2000). Normalized cuts and image segmentation. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*.

Shih, S.-W. and Liu, J. (2004). A novel approach to 3d gaze tracking using stereo cameras. *IEEE Trans. Systems, Man, and Cybernetics*.

Stiefelhagen, R., Yang, J., and Waibel, A. (1997). Tracking eyes and monitoring eye gaze. *Proc. Workshop Perceptual User Interfaces*.

Tan, K. H., Kriegman, D. J., and Ahuja, N. (2002). Appearance-based eye gaze estimation. *Proc. Sixth IEEE Workshop Application of Computer Vision '02*.

T.F.Cootes, C.J.Taylor, D.H.Cooper, and J.Graham (1995). Active shape models– their training and application. *Computer vision and image understanding*, 61(1):38–59.

Wang, J.-G., Sung, E., et al. (2005). Estimating the eye gaze from one eye. *Computer Vision and Image Understanding*.

Weinberger, K. Q. and Saul, L. K. (2006). Unsupervised learning of image manifolds by semidefinite programming. *Int. J. Comput. Vision*, 70(1):77–90.

Williams, O., Blake, A., and Cipolla, R. (2006). Sparse and semi-supervised visual mapping with the s3p. *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition*.

XU, L.-Q., Machin, D., and Sheppard, P. (1998). A novel approach to real-time non-intrusive gaze finding. *Proc. British Machine Vision Conference*.

Zhang, J., Li, S. Z., and Wang, J. (2004). Manifold learning and applications in recognition. In *in Intelligent Multimedia Processing with Soft Computing*, pages 281–300. Springer-Verlag.

Zhu, Z. and Ji, Q. (2007). Novel eye gaze tracking techniques under natural head movement. *IEEE TRANSACTIONS on biomedical engineering*.