Eye Typing using Markov and Active Appearance Models

Dan Witzner Hansen, John Paulin Hansen, Mads Nielsen, Anders Sewerin Johansen
IT University of Copenhagen
Glentevej 67, 2400 Copenhagen NV, Denmark
{witzner,paulin,malte,dduck}@itu.dk

Mikkel B. Stegmann
IMM, Technical University of Denmark
2800 Kgs. Lyngby, Denmark
mbs@imm.dtu.dk

Abstract

We propose a non-intrusive eye tracking system intended for the use of everyday gaze typing using web cameras. We argue that high precision in gaze tracking is not needed for on-screen typing due to natural language redundancy. This facilitates the use of low-cost video components for advanced multi-modal interactions based on video tracking systems. Robust methods are needed to track the eyes using web cameras due to the poor image quality. A real-time tracking scheme using a mean-shift color tracker and an Active Appearance Model of the eye is proposed. It is possible from this model to infer the state of the eye such as eye corners and the pupil location under scale and rotational changes.

1 Introduction

Humans acquire a vast amount of information through the eyes, and in turn the eyes reveal information about our attention and intention. Detection of the eye gaze facilitates collection of valuable information for uses in disciplines such as psychophysics and human computer interaction (HCI). For severely disabled people, the need for means of communication is acute. Producing text through eye positioning ("eye typing") is an appropriate modality for this purpose, as conscious control of eye movements is retained in most types of handicaps. In our project we aim for enabling disabled people, and in particular people with Amyotrophic Lateral Sclerosis (ALS), to type using their eyes via non-intrusive methods and off-the-shelf hardware components. Low cost cameras produce low quality images, hence eye tracking based on these images requires robust methods.

Several types of advanced eye tracking systems have been developed. The most popular types use infrared light (IR) and cameras to detect and track the eye. The use of the pupil and purkinje image facilitates highly accurate and robust gaze tracking methods and is applied in many commercial products [5, 8]. Flexible placement of camera and light sources, avoiding headmounts and IR light emitters, allows for usage in a wide range of scenarios such as in wheel chairs or mobile units. The use of off-the-shelf hardware components makes it possible for a larger group of people to have easy access to the system.

Methods for extraction of eye features, such as eye corners and iris contours, can roughly be divided into two classes: a) methods based on inference of local information and b) methods based on deformable templates.

The current work is a derivative of the latter category, but inherent ideas from the first class are also incorporated. Many current eye tracking systems base their tracking (quite successfully) on edge detection, morphological operators and template matching. These methods are typically not made invariant to rotation or scale changes and only exploit the information about the shape of the eye indirectly. By using shape statistics, the shape of the eye, as it appears in the image, is captured by the model.

2 System Setup

As web cameras have wide angle lenses, the camera must be placed fairly close to the subject to capture sufficient resolution around the eyes. The actual geometry of the person, camera and monitor configuration is restricted but not predefined. We cannot take advantage of IR light in this setting as we are limited to off-the-shelf components. An illustration of the system setup is given in Figure 1.

Proceedings of the Sixth IEEE Workshop on Applications of Computer Vision (WACV'02)
0-7695-1858-3/02 $17.00 © 2002 **IEEE**

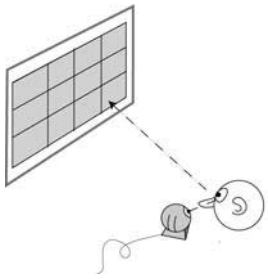

Figure 1. The screen, camera and subject system configuration.

3 Tracking the Eye

Both color and shape are powerful cues for robust detection and tracking of the eyes. Large areas of skin surround the eyes. The color information in the eye and skin regions are distinct, making color information very useful for eye tracking. The shape and texture of the eye provide information about the direction in which people are looking and the state of the eye (pupil position, eye corners and blinks). Combining these pieces of information provides evidence of the presence of the eyes. To our knowledge, these cues have not been employed jointly in the eye tracking community until now.

The shape and texture model is implemented using an Active Appearance Model (AAM) [2, 4]. Given a reasonable initialization, these models can find shapes of a particular class,but can only be used for tracking in cases where the actual movement is fairly small compared to the size of the object. In this approach, greater shifts in eye position (caused by head movements) are tracked based on the color distributions of the eye region using the mean-shift paradigm [1]. The center of the mean-shift kernel is used for initializing the shape model.

3.1 Eye Model

Active Appearance Models establish a compact parameterization of object variability as learned from a training set by estimating a set of latent variables. The modeled object properties are usually shape and *texture* (i.e. pixel intensity). From these quantities, new images similar to the training set can be generated.

Objects are defined by marking up each example with points of correspondence over the set either by hand or by semi- to completely automated methods. The key to the compactness of these models lies in proper compensation of shape variability prior to modeling texture variability. By exploiting approximate prior knowledge about the local nature of the optimization space, these models can be fitted to unseen images in a fraction of a second, given a reasonable initialization. An overlap of roughly 60% between the model and the object in the image is generally sufficient for accurate location.

Variability is modeled by means of an eigen analysis of the dispersions of shape and texture. Shapes are brought into alignment using a Generalized Procrustes Analysis, and textures are warped into correspondence using piece-wise affine warp, hence compensating for any variation in shape. Let $\overline{\mathbf{x}}$ and $\overline{\mathbf{t}}$ denote the shape and texture mean, respectively. The model parameters, $\mathbf{c}$, can then be used for linear generation of new instances by:

$$\mathbf{x} = \overline{\mathbf{x}} + \mathbf{\Phi}_s \mathbf{c} \quad , \quad \mathbf{t} = \overline{\mathbf{t}} + \mathbf{\Phi}_t \mathbf{c} \tag{1}$$

where $\mathbf{\Phi}_s$ and $\mathbf{\Phi}_t$ are eigen vectors obtained from the training set covariances. The object instance, $(\mathbf{x}, \mathbf{t})$, is synthesized into an image by warping the pixel intensities of $\mathbf{t}$ into the geometry of the shape $\mathbf{x}$.

Residual vectors between the model and image, $\delta\mathbf{t} = \mathbf{t}_{model} - \mathbf{t}_{image}$, are regressed against known displacement vectors, $\delta\mathbf{c}$, using principal component regression:

$$\delta\mathbf{c} = \mathbf{R}\delta\mathbf{t} \tag{2}$$

Embedded into an iterative updating scheme, this has proven to be a very efficient and robust way of matching these models to unseen images [2].

AAM provides a method that can model changes in scale, translation and rotation. Additionally, the model facilitates analysis and deduction of information regarding the state of the eye directly from the model parameters, $\mathbf{c}$.

3.2 Color Tracking

The mean-shift tracker algorithm, proposed by Comaniciu, Ramesh and Meer [1], is a fast appearance-based tracker which has proven useful especially in tracking of non-rigid objects [11]. The task of the mean-shift paradigm is to estimate the exact location of the mean given data points x_i and an approximate location of the mean of the data. This is done by determining the shift vector from the initial estimate. In terms of single hypothesis tracking, mean-shift is a real-time algorithm that endeavors to maximize the similarity between two statistical distributions. These distributions are a) the target model and b) the current estimate of the target. The similarity measure between the two distributions is expressed as measurements derived from the Bhattacharyya coefficient [1] and can be built using any characteristic discriminating to a particular object of interest. The model may include information of color, texture and feature strengths, but only color information is used here.

For incorporation of spatial information and for the optimization procedure, the contributions in the image are weighted by the convex and monotonic decreasing Epanechnikov kernel prior to building the distribution. This has the effect of assigning lower weight to locations that are more distant to the center of the target. The search for the object in the current frame starts with the location estimated in the previous frame and continues in the gradient direction of the similarity landscape using the mean-shift vector.

3.3 Gaze Determination

We have used a Gaussian Process interpolation method for inferring the mapping from image coordinates to screen coordinates [6]. A Gaussian process is a stochastic process where every joint density function is Gaussian. The idea of Gaussian process modeling is to place a prior directly on the space of functions. Just as a Gaussian distribution is fully specified by the mean and covariance matrix, a Gaussian Process is specified by a mean and covariance function. The actual function to be inferred is assumed to be a single function from this Gaussian distribution. In terms of eye tracking, the mean value of the interpolated point corresponds to the estimated position, and the confidence in the estimate is expressed by the variance. Hence it is possible to evaluate the actual performance with the estimated confidence for qualitative evaluation of the tracker. The calibration procedure is done by having the subject look at 12 on-screen points, corresponding to the button centers in the user interface (Figure 2) for a predetermined time interval.

By using the learning method, camera calibration is avoided. This is particularly important when addressing people with limited technical skills.

4 User Interface

Many interactive systems have been proposed, which integrate with a gaze tracking systems. An overview of different systems and applications with emphasis on eye typing is given in [7]. Most current eye typing systems implement an on-screen keyboard. A noticeable exception to this is "Dasher" proposed by Ward et al. [10]. The eye tracking device tracks the user's gaze, and the system determines which letter has been selected. Usually, a full keyboard is used, and it is therefore crucial that the accuracy of the gaze tracker is high. Since we use low-end cameras, the poor quality of the cameras and low resolution of the images result in a reduced accuracy in gaze determination. We have consequently proposed a method that uses a reduced set of three by four on-screen buttons accompanied with a letter and word prediction scheme for fast typing in both Danish, English and Japanese.

We use dwell time activation of the buttons for reasons of simplicity and comfort to the users. For coping with a reduced set of on-screen buttons, a probabilistic character layout is proposed in which the six most likely words (displayed on 1 button) and letters (displayed on 6 buttons) are suggested. This is done by representing the letter and word configurations in the language by a Markov model. The Markov model was trained on 10MB of text taken from various newsgroups. The remaining buttons are used for selecting sub-menus, which enables the user to select letters that are not suggested by the language model, deleting letters and various other features such as sending emails. A screenshot of the Japanese user interface is shown in Figure 2. The top left two buttons are used for displaying the text, and the button in white is the currently selected button.

Figure 2. The Japanese version of the user interface (Courtesy of Kenji Itoh).

5 Results

In our approach, each color channel for the mean-shift tracker is quantized to 32 levels. The mean-shift tracker is not very sensitive to projective distortions and consequently, the color eye tracker performs very well in tracking the eye under moderate head rotations. Using the AAM-API [9], the shape and appearance model is trained on 28 images selected from image sequences of different people with different gaze directions and head poses. The possibility of making personalized models for each individual user is obvious, but manually annotating the images is time consuming. All the sequences are recorded placing the camera below the line of sight of the user because: a) The web camera needs to be placed fairly close to the subject b) the camera must not obscure the line of sight of the user c) images taken from below usually obtain a much better view of the eye because

of smaller movements of the lower eyelid.

The Active Appearance Model is a 3-layer pyramidal model where each layer has been trained to the corresponding scale. The reason for using the hierarchical model is mainly speed but also robustness towards local minimas. We obtain a low dimensional subspace where the 16 largest modes of variation in shape and texture correspond to 95 % of the variation in the training set. Compared to correlation-based methods, which have dimensions equal to the size of the matching kernel, this is a significant improvement as it is possible to get a direct interpretation of the state of the eye in this low dimensional space.

In Figure 3, the first synthesized mode of variation of the training set is shown. The first mode corresponds to eyelid and pupil position, the others are refinements of shape variations.

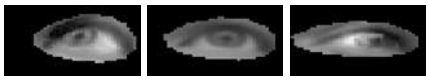

Figure 3. The first combined mode of variation for $-3\sigma_1$, mean shape and $3\sigma_1$.

The use of the layered scale implementation reduces the possibility of the AAM being stuck in local minimas. However, it may still be slightly inaccurate. The inaccuracy occurs most frequently around the pupil since a large pupil movement in space from one image to another corresponds to a large change in shape space as well. A local optimization scheme based on the ideas from Active Shape Models [3] is used around the pupil segment of the model. This is done by selecting points of high intensity gradient along the normals of the shape segment. This found shape does not necessarily live in shape space, and it is consequently projected back into shape space using the inverse of the generation procedure defined in Equation 1.

The eye model and tracker are robust towards some changes in lighting conditions. However, the eye model is particularly robust with respect to i.i.d. image noise and hence, it is well-suited for tracking in images of low quality. Figure 5 shows examples of the recovered shape for three different people with varying pose, scale, resolution and ethnicity using the same model. Having manually initialized an approximate position and scale of the eye and using a 1 GHz Pentium PC, real-time tracking performance of the eyes is obtained in images with the dimensions of 320×240 pixels. We have tested the tracker off-line on 48 sequences of approximately one minute duration. In 6 of the sequences, the tracker failed due to fast head movements (color tracker failure) or the AAM failed in accurately determining the state of the eye. The latter failure was based on a subjective evaluation, but the errors generally occurred because the fitted model had been rotated such that the eye corners were inaccurate. A crude solution to this is to neglect all rotations. This obviously results in slight inaccuracies in eye corner detection. One way of correcting this is by having a more elaborate model which includes more features, e.g. using a two-eye model or multi channel models.

In Figure 4, the results of using a web camera and a high resolution camera is shown. In the case of the web camera, the user is sitting roughly 60 cm away from a 17" screen and looking at the buttons on the screen. In this result, the user is only moving the head slightly. The squares correspond to the buttons on the user interface. The centers of the ellipses are the estimated position (the mean) and the circles are the corresponding variances. Hence it is desirable to have either points that are far from the center with large variance or points that are accurate with low variance. The straight lines show the estimation error under the assumption, that the person is looking directly at the center of the button. The mean on-screen deviation for the web camera is 1.4 cm and 0.79 cm for the high resolution camera.

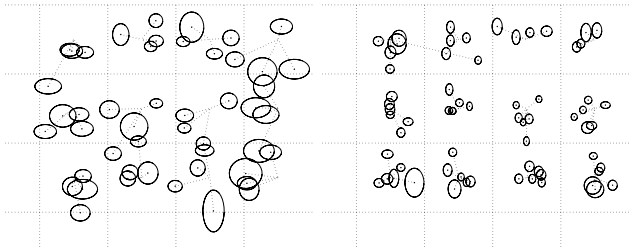

Figure 4. Results from using the Gaussian Process interpolation method for gaze determination using a) web camera and b) high resolution camera.

The word and letter prediction scheme facilitates typing $10-20$ words per minute for common phrases. Subjects can produce more than 20 Japanese characters per minute (Hiragana and Kanji) after four hours of training on a similar typing interface, using a standard IR eye tracker.

6 Conclusion and Future Work

In this paper, we propose a novel method for eye tracking without making explicit assumptions on the shape of the eye.

We have combined a fast mean-shift color tracker for tracking of the eye with a specialized model for finding the location of eye features. We have shown that by using the Active Appearance Model, it is possible to directly deduce information regarding eye corners and pupil position in an easy and intuitive manner. The method has been applied

Figure 5. Examples of the recovered eye shape (left to right) for different people (rows 1-3) under different poses, scales and resolution using the same model.

to images taken by low cost web cameras, but the method can equally well be used with IR and high quality cameras. We have shown that high resolution of the gaze determination is not needed for eye typing due to natural language redundancy. Using a training method for gaze determination facilitates the usage of the system for a larger audience as knowledge of the camera is avoided.

In this work, we have not applied the method for blink detection (using shape information), and it is doubtful that this can be employed in the current setting due to the shape variability and occlusions (eyelids and pupil). The shape model gives one mode for both pupil and eyelid positions. A natural extension of the current work would be to replace the shape model with a corresponding model, which separates these components, hence obtaining a direct interpretation of the current state of the eye from the model. The mean-shift tracker may in some cases fail to track due to lack of handling multiple hypothesis. We are currently trying to solve this using particle filtering. The particle filter could be a way of initializing the tracker automatically. To track the eyes for a long time, we are currently working on a shape model which includes both eyes. This will naturally imply less resolution for each eye and hence we are considering better cameras.

Acknowledgments. The Danish Ministry of Science, Technology and Innovation funds this project. We would like to thank David MacKay, Cambridge University for helpful comments and guidelines regarding the Gaussian Process Model. We would furthermore like to thank Kenji Itoh and co-workers at Tokyo Institute of Technology regarding the Japanese version of the user interface.

References

[1] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, pages II:142–149, 2000.

[2] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. In *Proc. European Conf. on Computer Vision*, volume 2, pages 484–498. Springer, 1998.

[3] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models - their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, 1995.

[4] G. J. Edwards, T. F. Cootes, and C. J. Taylor. Face recognition using active appearance models. In *ECCV'98. 5th European Conf. on Computer Vision. Proc.*, volume 2, pages 581–95. Springer-Verlag, 1998.

[5] K. Grauman, M. Betke, J. Gips, and G.R. Bradski. Communication via eye blinks:detection and duration analysis in real time. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, pages I:1010–1017, 2001.

[6] D. J. C. MacKay. Introduction to Gaussian processes. In C. M. Bishop, editor, *Neural Networks and Machine Learning*, NATO ASI Series, pages 133–166. Kluwer Academic Press, 1998.

[7] Päivi Majaranta and Kari-Jouko Räihä. Twenty years of eye typing: Systems and design issues. In *Symposium on ETRA 2002: Eye Tracking Research Applications Symposium, New Orleans, Louisiana*, 2002.

[8] S. Sirohey, A. Rosenfeld, and Z. Duric. A method of detecting and tracking irises and eyelids in video. *Pattern Recognition*, 35(6):1389–1401, June 2002.

[9] Mikkel B. Stegmann and Rasmus Larsen. Multi-band modelling of appearance. In *First International Workshop on Generative-Model-Based Vision*, 2002.

[10] David J. Ward, Alan F. Blackwell, and David J. C. MacKay. Dasher - a data entry interface using continuous gestures and language models. In *UIST 2000: The 13th Annual ACM Symposium on User Interface Software amd Technology*, 2001.

[11] A. Yilmaz, K. Shafique, N. da Victoria Lobo, X. Li, T. Olson, and M.A. Shah. Target-tracking in flir imagery using mean-shift and global motion compensation. In *Computer Vision Beyond the Visible Spectrum: Methods and Applications*, 2001.