Real-time Non-Intrusive Eyetracking and Gaze-point Determination for Human-Computer Interaction and Biomedicine

ASHIT TALUKDER, JOHN-MICHAEL MOROOKIAN, S. MONACOS, R. LAM, C. LABAW
Intelligent Instruments and Systems Technology Group, In-Situ Technology and Experiments Systems Section
Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Drive, MS 300-123, Pasadena, CA 91109
USA

Abstract: - Eyetracking is one of the latest technologies that has shown potential in several areas including human-computer interaction for people with and without disabilities, and for non-intrusive monitoring, detection, and even diagnosis of physiological and neurological problems in individuals. Current non-intrusive eyetracking methods achieve a 30 Hz rate with possibly low accuracy in gaze estimation, that is insufficient for many applications. We propose a new non-intrusive visual eyetracking system that is capable of operating at speeds as high as 6-12 KHz. A novel fast image processing algorithm leverages specific features of the input CCD camera and system hardware to yield a real-time eyetracking system. Initial results show the excellent performance of our system under severe head motion and low contrast conditions.

Key-Words: - Detection, Eyetracking, Gaze mapping, Human-computer Interaction (HCI), Image Processing, Tracking,

1. Introduction

Non-intrusive human-computer interaction is a field that promises to ease the communications between machines and humans, by providing on the machine an easy-to-use interface that humans can understand, while simultaneously providing the tools that machines can use to interpret the actions of humans.

Recent technological advancements in human-computer interaction include audio-based interactions such as speech recognition [1], natural language processing, and speech synthesis. Visual recognition and interface tools include face detection [2] and recognition [3], facial expression estimation [4], and gesture recognition of sign language with hand motion [5]. While such interface mechanisms are useful for providing information about the identity of a user, and for explicit communication between a human and a computer, they are not very useful for assisting handicapped people who have trouble speaking or moving. Eyetracking and automatic gaze point determination is a technology that can be used as an alternative interface tool to assist normal and handicapped individuals, and also provide a plethora of other physiological information for various applications, such as visual acuity determination for ophthalmology applications, reading disorders, evaluation of user-interface designs, etc.

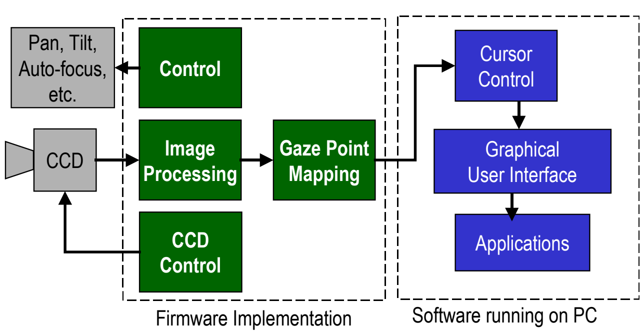


Figure 1: Eyetracking system software architecture

Eyetracking has traditionally been achieved using several approaches, many of which require the user to wear a device, thereby hindering their normal activities. A thorough review of prior techniques for eyetracking has been done in [6]. These prior eyetracking solutions are mostly intrusive, and work at a maximum rate of 60 Hz. Non-intrusive and remote gaze point determination at significantly higher rates would open the gates to a host of other applications that current eyetracking solutions cannot address.

Our solution is a non-intrusive, real-time eyetracker using a fast, remote CCD video camera and associated image processing for locating eye features from a distance and estimating the gaze point. The system operates at 6-12 KHz and is ultimately expected to have head-tracking incorporated to allow eyetracking during free motion of the head. The software architecture of the system is shown in Figure 1.

We first discuss non-intrusive eyetracking techniques in Section 1.1, followed by a discussion of potential applications (Section 1.2) that could arise from the high-speed, non-intrusive eyetracking solution that we propose. The system architecture for our eyetracker is presented in Section 2. Our image processing algorithms for eye feature location is discussed next (Section 3), followed by a description of our gaze mapping technique (Section 4). Initial results are presented in Section 5.

1.1. Operational Principle

Current non-intrusive eyetracking solutions are principally divided between three approaches. All three solutions use the fact that the eye reflects incident light from various surfaces, that can be then used to track an eye and estimate the gaze direction.

When light is shined into the user's eye, several reflections occur on the boundaries of the lens and cornea, the so-called Purkinje images (Figure 2). The first Purkinje image off the corneal surface is also called the glint.

The first solution to eye gaze mapping uses the location of the glint with standard incident visible light and the dark pupil, and uses an artificial neural network to estimate gaze coordinates based on glint and pupil locations [7]. However, this approach requires extensive training examples for calibration, and its gaze mapping is not highly accurate.

The first and fourth Purkinje images can be used for tracking the direction of gaze by the Dual-Purkinje Image technique [8], which uses the relative positions of these reflections to calculate the direction. The Dual-Purkinje-Image technique is generally more accurate than the prior technique, but the main disadvantage with this technique is that the fourth Purkinje image is rather weak, so the surrounding lighting must be heavily controlled.

Our eyetracking system uses the Pupil-Center/Corneal-Reflection (PCCR) method (Figure 2) to non-intrusively determine the eye's gaze direction. The main concept of the PCCR technique is to locate the pupil center and the center of the corneal reflection off the eye surface, and use these two centroids to determine the gaze direction of a user. To this end, a video camera is oriented close to the user's nominal gaze direction such that it is focused on the user's eye.

A small, low power, infrared emitting diode (IRED) located at the center of the camera lens illuminates the eye. The IRED generates the corneal reflection and causes the bright pupil effect, which enhances the camera's image of the pupil, as seen in Figure 3. This bright pupil effect (Figure 3), caused by light that enters the eye and is reflected by the fundus at the rear of the eye back through the pupil, is also commonly known as the red-eye effect that is noticed predominantly during flash photography.

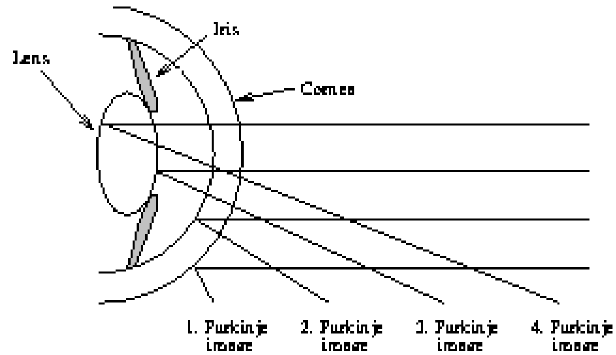


Figure 2: Reflection-based method(s) for eye tracking.

The iris acts as a sink for the IR radiation, while the aperture of the pupil acts as a reflector. An IR sensitive video camera can therefore be used to obtain a high contrast black and white image of the iris (black) and the pupil (bright).

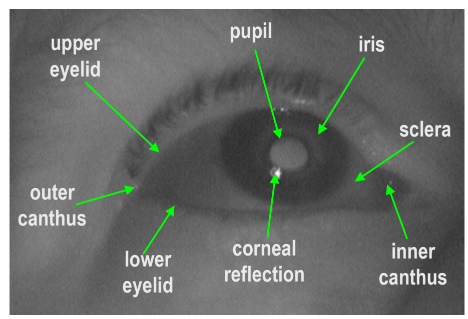


Figure 3: IR illuminated image

For many human/machine interface applications, we are interested in the point at which the gaze direction intersects a computer display rather than the gaze direction itself. Thus, we employ a gaze mapping algorithm which maps the relationship between the pupil and corneal reflection centroids to the user's gaze point on the computer display. The relative locations of the pupil and the corneal reflection vary systematically as a function of the direction of gaze. The vector difference between the pupil and the CR can be correlated to objects that are fixated.

1.2. Eyetracking Applications

Development of a high-frame rate eyetracking system enables a broader class of applications than current commercially available systems.

Refractive surgery requires accurate positioning of the laser beam onto the center of the pupil at speeds of 4KHz. Certain eye defects and physiological disorders such as amblyopia and those involving visual attention (or lack of it) can be diagnosed if knowledge of the pupil motion can be made available at speeds up in the kilo-hertz range.

Several severe clinical conditions, such as Parkinson's disease, Multiple Sclerosis, and Myasthenia Gravis, among others, can be associated with specific types of saccadic eye movement abnormalities, as detailed in [9].

It has also been shown in previous work [10, 11] that vital neurological signs such as heart-rate and inter-cranial pressure can be extracted by evaluating the pupil's size, shape, equality, and response to light, and charting the changes in these pupil characteristics over time.

Therefore, extremely high-speed eyetracking is necessary in order to achieve desired improvements in biomedicine such as refractive surgery and medical diagnosis.

2. System Architecture

Many commercial eyetracking systems that operate using the PCCR technique rely on standard video cameras, which produce full-frame data at approximately 30 frames-per-second. The video stream is typically processed in a computer using either the Central Processing Unit (CPU) or a Digital Signal Processor (DSP) on a frame-grabber board. Such systems are limited to slow-speed, full-frame operation and place undue requirements on the host computer system.

In contrast, our eyetracking system is envisioned to ultimately do both control and image processing in reprogrammable "firmware" on a Field-Programmable Gate Array (FPGA). Because the time-intensive processing is implemented in the FPGA, the burden is removed from the host computer.

We also implement a novel control mechanism for a commercial charge coupled device (CCD) imager, which makes region-of-interest (ROI), or sub-window, readout possible at very high-rates [12] (several kilohertz frame rates), compared to other commercially available cameras that have full-frame, large form factor video streams. Since the features of interest in eyetracking (corneal reflection and pupil) typically occupy a small subsection of the entire camera frame, ROI capability allows the user's gaze direction to be determined at extremely high rates that exceed current eyetracking system speeds by orders of magnitude.

3. Image Processing Algorithms for Fast Detection and Tracking

The image processing algorithm for locating the pupil and cornea in the eye is a very important component of our eyetracking system. Several options exist for detecting the pupil and cornea. These include automatic thresholding followed by simple blob coloring and binary shape-based blob filtering to remove blobs that are not "circular" in shape. Such a solution that uses absolute gray-level measures, while very fast in practice, is not expected to work well when multiple reflections off the sclera are present, and when the person's skin tone is light, and therefore similar to the pupil reflection. Additionally, this solution requires that the CCD camera capture the entire image prior to processing, and does not take advantage of the fact that eyetracking time can be dramatically reduced by using the fact that subwindowing and scanning specific rows in the CCD camera is significantly faster than scanning the entire image frame.

The Hough transform has been actively used to robustly locate specific shapes in an image [13] such as lines and circles. Initial results of the Hough transform for locating circular shapes such as the pupil and cornea under different motion and contrast conditions were very promising. However, the Hough transform for detecting circles is not suited for running in real-time and it also requires the entire image frame to be acquired prior to processing.

We use a new slice-based image processing technique that leverages on the CCD's fast

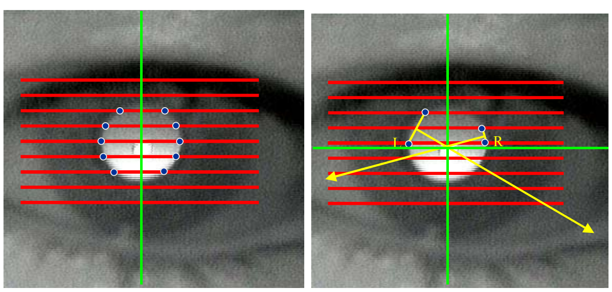
Figure 4: Typical images

acquisition rate of image row slices, thereby ensuring a highly optimized end-to-end eyetracking system. We take advantage of the symmetry of the circles and the gray-scale contrast of the pupil and cornea from other parts of the eye in designing our image processing algorithm. In this approach, we determine horizontal image slices (Figure 5) that contain the pupil and cornea, and note the start and end coordinates of the pupil/cornea on each valid slice. Information from multiple slices is then combined to robustly locate the pupil and corneal centers. The number of slices is fewer than the total number of rows in the image, i.e. the slices are a subset of image rows, as shown by the horizontal red lines in Figure 5.

We first discuss the slice-based algorithm to locate borders of a pupil in a scanned row (slice). For a given image row (slice), we apply the 1D Canny edge filter [14] to extract the high-frequency components in the scan-line. The Canny edge filter has been shown [14] to provide excellent signal-to-noise ratio in extracting edges in signals. The gray value at each pixel fused with its edge magnitude is used to mask out pixels in the slice that are not likely candidates for the pupil/corneal edges. The magnitude and sign of the edge value at each remaining pixel is then used to select likely candidates for the start and end of the pupil within that slice. Grouping of such candidate pairs based on edge magnitude, sign, and distance between pairs of pixels combined with prior knowledge of the expected size of a pupil is used to determine the pupil border pixel locations within the row (slice).

A similar method is used to detect corneal borders in a given image row slice. The gray-scale and edge threshold values used to mask candidate corneal border pixels are different from those used to locate pupil borders.

The prior steps yield a list of slices that contain pupil and corneal regions. This list might contain false alarms: slices that are incorrectly categorized as containing pupil or cornea. Additionally, some slices containing pupil/cornea may not have been



(a) (b)

Figure 5: Horizontal slicing algorithm for locating pupil and cornea

detected. Therefore, we postprocess the data where we fill gaps in the slice list by grouping adjacent slices that have similar pupil/corneal midpoints along the x-axis (horizontal direction). Slices that have outlier midpoint values along the x-axis, or are isolated and do not have adjacent slices (along the y-axis or vertical direction) are rejected as false alarms. The average midpoint between the pupil edges in the remaining slices gives the horizontal (x-axis) center of the pupil, as shown by the vertical green line in Figure 5a.

In order to locate the y-center of the pupil, we choose two pairs of points on the edge of the pupil from the list of slices, compute the normal vectors to these line pairs, as shown in Figure 5b, and note the intersection of these normals with the horizontal coordinate of the pupil center (vertical line in Figures 6a and b). The average y-coordinate of the intersection computed from several line pairs (4-tuple of points along the pupil borders) yields the y-coordinate of the pupil center.

A similar mechanism is used to locate the center of the cornea. The post processing is effectively able to detect the corneal slices in the presence of reflections on the sclera that look similar to the corneal reflection.

This approach of processing slices (image rows) makes effective use of the camera hardware. One parameter to choose is the number of rows to skip between two adjacent slices. The image processing algorithm is faster when the rows skipped are greater, at the cost of reduced resolution and risk of missing much (or all) of the pupil, especially when the eyelids cover the pupil. Conversely, pupil detection accuracy will improve greatly if fewer rows are skipped between adjacent slices, but at the cost of slower performance.

4. Gaze-point Mapping Algorithm

An estimate of the gaze point (or direction) can be obtained by locating the centers of the pupil and cornea. Traditionally, gaze mapping in the PCCR uses the assumptions that: (1) the pupil is approximately centered around the optical axis of the eye, (2) the corneal surface is spherical, and (3) light reflects off the corneal surface, i.e. the first Purkinje reflection (Figure 2) is visible. Therefore, the gaze angles (horizontal and vertical), relative to the camera pointing direction Z, can be measured by computing the vector between the pupil and corneal reflections (Figure 6). When the person looks directly at the camera, the two centers approximately overlap, when the person looks to his/her right, the pupil center moves to the left of the corneal center.

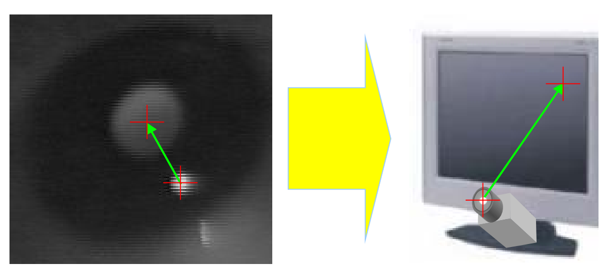


Figure 6: Gaze point estimation concept

If the linear difference vector between the pupil center (p = [px py]) and corneal center (c = [cx cy]) are d = [dx dy], a linear approximation to gaze point mapping can assume that:

$$\begin{aligned} Xscreen &= a11 + a12\ dx \qquad (1) \\ Yscreen &= a21 + a22\ dy \end{aligned}$$

where Xscreen is the x-coordinate of the gaze point on the screen. This concept is illustrated in Figure 6.

In practice, a quadratic mapping is traditionally used to compensate for roll angle misalignments between the camera's and screen's horizontal axes, as shown in Eq (2).

$$\begin{aligned} Xscreen &= a11 + a12\,dx + a13\,dy + a14\,dx\,dy \\ Yscreen &= a21 + a22\,dy + a23\,dx + a24\,dy\,dy \end{aligned} \qquad (2)$$

However, we note that the pupil may not lie in the center of the optical axis, nonlinearities can occur due to corneal flattening, and there might be misalignments and nonlinearities associated with the camera pointing vector, and the change in eye (pupil and corneal) shape as the gaze direction changes,. Therefore, we use the actual x,y image coordinates of the pupil and cornea to estimate the gaze point. We use the quadratic form in Eq (2), but with the **true** pupil and corneal centers (not the difference vectors). This is expected to yield more general solutions than those that use only the difference vector.

5. Results

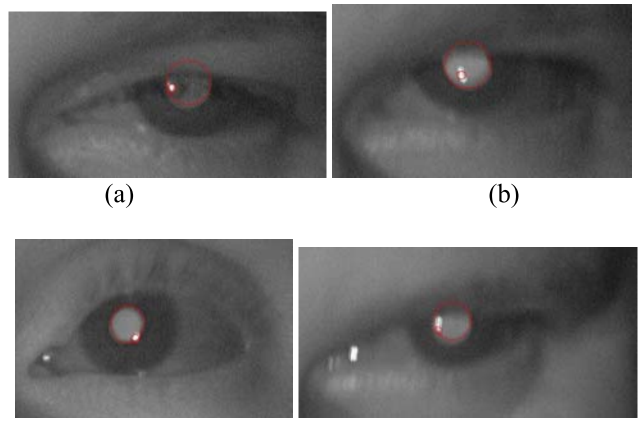
Our eyetracking system is comprised of two main components: the hardware and system software. The results from our software component are available. The novel software features are the image processing and the gaze-point mapping algorithms. We discuss our initial results from our system software components next.

5.1. Image Processing Results

The image processing algorithm is one of the critical components of our eyetracking system. Initial tests of our image processing algorithms were done on a PC platform with prototyping in MATLAB. Tests on three subjects were carried out. In two of these tests, the subject was "compliant" and no sudden head movements were carried out. The third case involved intentional rapid head motion by the subject to test the limits and robustness of our image processing algorithms.

The first two tests were comprised of the subject looking at 20 gaze points on the screen. The last one involved random eye and head movement. All tests were done at a 30 Hz frame-rate with a total time of 20 seconds per subject.

The average accuracy in detecting the pupil and centroid centers was determined by counting the number of frames where the error in locating the pupil and centroid centers is less than 15 pixels. Since no truth data is available, the true pupil and corneal centers were marked manually.



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

Figure 7: Results of pupil and corneal detection

We first evaluated the image processing performance on the two subjects without artificial head motion. The average accuracy in pupil detection was 96.4%. The average accuracy in corneal reflection detection was 96.8%. Most of the errors occurred for frames where the eyelid covered most of the pupil during the process of blinking.

For the third case where the subject intentionally moved the head rapidly, the accuracy in correctly locating the pupil and cornea was 91.4% and 92.6% respectively. The decrease in tracking performance occurred due to motion blur caused by rapid motion in the eye features over the camera integration time. Figure 7 shows results of our image processing algorithm that accurately locate the pupil and corneal reflection under various conditions. The red circles are the locations of the detected pupil and corneal reflection. It locates the corneal reflection and pupil even in the presence of strong motion blur (Figure 7b) that results in blurring and defocusing of the cornea and pupil areas. Figure 7a shows an example where only about 55-60% of the pupil is visible and 40% is covered by the eyelid. Additionally, its contrast relative to the iris is very low. The pupil center is located with high accuracy even in this case. Figure 7d shows our image processing results where 25-30% of the pupil is covered by the eyelid, and head motion causes the eye features to be blurred. The presence of reflection off the sclera further complicates the problem, but performance is robust even in this case.

The average execution time of the current prototype MATLAB image processing code is 0.7 seconds per frame. The run-time is expected to speed up tremendously after the algorithm is implemented in C/HDL in the near future.

5.2. Gaze-point Mapping Results

The centers of the pupil and cornea are used from the image processing outputs to estimate the gaze point of the eye at each frame. Considerable improvement in gaze mapping accuracy was obtained when our algorithm (that employs the actual location of the pupil and corneal centers to evaluate the gaze mapping) was used, compared to the standard approach (that employs the difference vector between the pupil and corneal centers).

During training and validation, a calibration target generator program was used to plot points (attractors) on the screen at one-second intervals, and the user was required to gaze at each point. The average of the absolute difference between the true and estimated gaze points was used as a measure of the efficacy of the gaze-mapping algorithm.

The average absolute gaze estimation error in the horizontal (x) direction was 12.3 pixels, and 30.7 pixels in the vertical direction when our gaze-mapping algorithm was used. In contrast, the standard (commonly used) gaze-mapping algorithm yielded an average absolute error of 16 pixels in X, and 40.4 pixels in the vertical direction. Further tests need to be done to evaluate and improve the performance of the gaze-mapping algorithms.

6. Conclusions and Future Work

In this paper, we have discussed a new real-time remote eyetracking and gaze mapping algorithm that is ultimately expected to operate at 6-12 KHz, thereby making it applicable to a variety of applications that current eyetrackers cannot be used for. New image processing algorithms that take advantage of the CCD subwindowing capabilities is detailed. Initial results are promising and illustrate the efficacy of our solution for pupil/corneal detection and tracking, and it's robustness to rapid head and eye motion. The new gaze mapping algorithm is shown to yield good estimates. Further improvements to the gaze mapping algorithm are expected, using advanced nonlinear mapping such as support vector machines or neural networks. We are currently evaluating conversion of our image processing routines from the MATLAB environment to implementation on the FPGA using either SIMULINK and Real-time Workshop tools in MATLAB, or direct implementation in HDL for the FPGA.

Acknowledgements

The research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, and was sponsored by the NASA Advanced Eye Tracker System Development Program (Task Order # 10562) by the NASA Office of Aero-Space Technology. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.

References:

1. Rabiner, L. and B.-H. Juang., *Fundamentals of Speech Recognition*. 1993: Prentice Hall.
2. Yang, G. and T.S. Huang, *Human face detection in a complex background.* Pattern Recognition, 1994. **27**(1): p. 53 - 63.
3. Talukder, A. and D. Casasent. *Adaptive Activation Function Neural Net for Face Recognition*. in *IEEE Intl Joint Conf. on Neural Networks*. 2001: IEEE.
4. Tian, Y., T. Kanade, and J. Cohn, *Recognizing action units for facial expression analysis.* IEEE Transactions PAMI, 2001. **23**(2): p. 97-115.
5. Starner, T., J. Weaver, and A. Pentland, *Real-Time American Sign Language Recognition Using Desk and Wearable Computer Based Video.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 1998. **20**(12): p. 1371-1375.
6. Collewijn, H., *Eye Movement Recording*, in *Vision research: A Practical Guide to Laboratory Methods*, C. R.H.S. and R. J.G., Editors. 1999, Oxford University Press. p. 245-285.
7. Baluja, S. and D. Pomerleau, *Non-intrusive gaze tracking using artificial neural networks*. 1994, Research Paper CMU-CS-94-102, School of Computer Science, Carnegie Mellon University: Pittsburgh PA.
8. Müller, P.U., et al., A comparison of a new limbus tracker, corneal reflection technique, purkinje eye tracking and electro-oculography, in Perception and Cognition, G. dYdewalle and J.V. Rensbergen, Editors. 1993, Elsevier Science Publishers. p. 393-401.
9. Ciuffreda, K.J. and B. Tannen, *Eye movement basics for the clinician*. 1995, Missouri: Mosby.
10. Yoshida, M., et al. *Statistical properties of simultaneously recorded fluctuations in pupil diameter and heart rate*. in *Proc. IEEE EMBC and CMBEC*. 1995: IEEE.
11. Calcagnini, G., et al., *Cardiovascular rhythms in spontaneous pupil diameter fluctuation.* IEEE Computers in Cardiology, 1997: p. 211-214.
12. Monacos, S.P., et al. *A High Frame Rate CCD Camera with Region-of-Interest Capability*. in *Proc. IEEE Aerospace Conference*. 2001. Big Sky Montana: IEEE.
13. Ballard, D.H., *Generalizing the Hough Transform to detect arbitray shapes.* Pattern Recognition, 1981. **13**: p. 111-122.
14. Canny, J., *A computational approach to edge detection.* IEEE Transactions PAMI, 1986. **8**: p. 679-698.