Lane Detection and Estimation using Perspective Image

Marcos Paulo Batista, Patrick Y. Shinzato, Denis F. Wolf and Diego Gomes
Mobile Robotics Laboratory
Institute of Mathematics and Computer Science
University of Sao Paulo - ICMC-USP, Sao Carlos, SP Brazil
[marcos.batista, diegogomes]@usp.br, [shinzato, denis]@icmc.usp.br

***Abstract*—Lateral localization of an autonomous vehicle within its lane is major information for its adequate control and navigation. Computer vision and robotics communities have used primarily images to Bird's Eye View for easier data manipulation than perspective image. Nevertheless, this technique usually assumes that the terrain is flat and needs calibration for its transformation matrix. In this paper an efficient method of detection and estimation of lane-based perspective image of a monocular camera is presented. Our algorithm is based on robust image processing, using Probabilistic Hough Transform, road marker estimation, and vehicle lateral localization in the lane. Our system provides satisfactory results, demonstrating its ability to detect lanes in several situations, including in variable light conditions, and even during the night. Our system also does not rely on metric data, but provides useful control information using the pixel's proportion. Therefore, the proposed methodology contributes a robust and user-friendly system that depends exclusively on a perspective image.**

***Index Terms*—Lane detection, lane estimation, perspective image, computer vision.**

I. Introduction

Autonomous vehicles and assisted steering systems are important to reduce traffic accidents and to improve traffic flow in large cities. Towards this objective, lane detection is an important tool to locate the vehicle on a navigable route to either aid the driver or control a fully autonomous vehicle. Due to commercial interests, it is preferred that detection is capable of functioning with a monocular camera, which costs considerably less than more sophisticated sensors, like LIDARs.

A large deal of research is being done on the development of this kind of system. However, the scientific community is faced with the problem of lane detection and its estimation in various situations, such as: urban environments, sharp curves, variable lighting (night time, high contrast, bright light with shade) and demarcations problems with road markers. Thus, one decision has to be made in this research: use detection and estimation methods for environments where there are continuous markings as in the case of highways, or get a more robust approach to detection for a dynamic urban environment. For the latter, the estimation may not be necessary. However, it might require significant additional research.

Several groups have worked to improve the processing time, accuracy and lane estimation in challenging environments. One technique is used frequently by these groups [1][2][3]: Bird's Eye View [4](BEV). It transforms a perspective image in a top view one, assuming that the road is flat and that the camera stays in a known position on the car. Aly [1] implemented a robust and fast (50 Hz) lane detection for urban environment. His system was able to detect additional lanes besides the ego-lane. This method used RANSAC[5] after a BEV transformation and thresholding, obtaining possible lines that form the road markers. However, this group tested the system just in a particular sequence of images and scenario, so it is not possible to evaluate and prove its efficiency for general urban situations.

Other work in the field, presented by Liu [2], used common tools in the area, such as BEV, Particle Filter [6] and Hough transform [7], but his system did not use image processing nor a filter. Therefore, his system might have a bad range of situations, mainly in variable light conditions. Borkar [3] also used BEV and a Kalman Filter [8] probabilistic estimator. The authors worked in night shots and got good results in straight and curvy roads using RANSAC for road markings detection.

Fig. 1. Carina - Autonomous test vehicle from Mobile Robot Laboratory - USP

Zhao [9] developed a real-time system using Hough Trans-

978-1-4799-6711-7/14 $31.00 © 2014 IEEE
DOI 10.1109/SBR.LARS.Robocontrol.2014.43

form for detection and a Kalman Filter for prediction. However, they did not use BEV and he attached the use of parallelism of road markers, which is assumed for the most of groups that work with BEV. Besides that, Lin [10] developed an approach based on image processing using edge-linking, YUV color space, and Bayesian probability model which did not achieve good results comparing to other works in the field.

This paper presents an efficient system for detection and estimation of lanes with data sets on varied scenarios. Our system, unlike other approaches in the field, proposes the detection of the lane and the localization of the car on it without the requirement of a calibration that gives the transformation of Birds Eye View, which can depend on a sensor and assumes that the ground is flat.

In order to evaluate our system, the vehicle Carina [11] (Intelligent Robotic Car for Autonomous Navigation), Fig. 1, has been used as the test platform. One of the goals of the presented system is to contribute to the development of autonomous vehicles and assisted-steering systems.

II. Lane estimation

This work presents a lane-detection system based on perspective and gray-scale images. The entire system is composed by two major steps. The first step is a pre-processing step where an enhancement filter emphasizes all road markings before running the Probabilistic Hough Transform, which generates a hypothesis for the next step. Using the lines generated by first step, the second step tries to detect the ego-lane: if it is unsuccessful, then an estimation method is performed using a previous ego-lane.

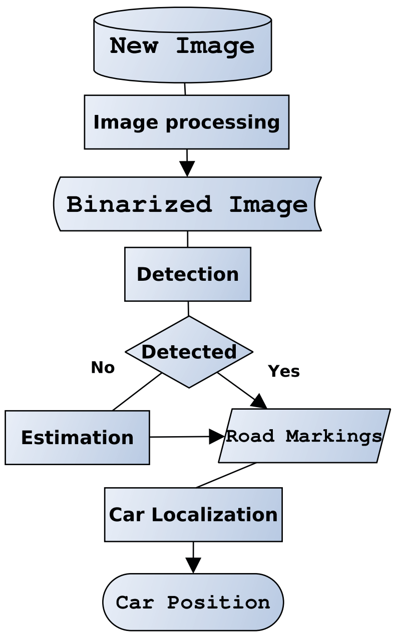

Fig. 2. Overview of the algorithm

The first step of our algorithm is pre-defining a ROI (Region of Interest) in front of the car to reduce computational cost in image processing , Fig. 3(a). Moreover, each angle value formed by all combinations of the lower and upper image's points is stored.

A. Image Pre-processing

The original image is defined as $\mathbf{I}(m,n)$, where m is number of rows and n is number of columns of image. The main idea of this filter is to accumulate differences of intensity between pixels with same distance and same line. For each line, a set of pixels from left side and a set of pixels from the right side from pixel $\mathbf{I}(u,v)$ are used to calculate a cost. For each pixel from the image, a new value $\mathbf{M}(u,v)$ is calculated based on the sum of these costs, as shows Equation 1:

$$\mathbf{M}(u,v) = \sum_{i=1}^{S} min \left(\begin{array}{l} max\left(\mathbf{I}(u,v) - l^{(3i)}, 0\right), \\ max\left(\mathbf{I}(u,v) - r^{(3i)}, 0\right) \end{array} \right), \qquad (1)$$

where S is a constant arbitrarily defined that represents the further pixel neighbor, $\mathbf{I}(u,v)$ is the gray-scale value from center pixel, $l^{(3i)}$, and $r^{(3i)}$ is, respectively, the gray-scale value from $3i^{th}$-pixel on left side and right side from $\mathbf{I}(u,v)$ in same line. Before the next step, $\mathbf{M}$, figure 3(b), is binarized using the Otsu's method [12]. Thus, the result of this process is an image (Fig. 3(c)) that has pixels classified as belonging or not belonging to the road markings.

Fig. 3. (a) ROI from the original image. (b) Image M which has accumation from pixel's contrast. (c) Processed ROI image using a Otsu thresholding.

B. Road Markings Detection

For road markings detection we use a Probabilistic Hough Transform [13] in the binarized image in order to detect all possible lines that have at least a minimum number of votes V and size L, which creates a set of all the possible straight lines that form the lane.

After the line detection, an efficient way of storing these lines is required. They are divided into two vectors of different lines, one for lines that represent right road markings and other for left road markings. This classification of right and left is given by the value of the tangent of the angle formed by the lane marking with a horizontal line, being positive for the right set and negative for the left set, as shown in Fig. 4.

At the same time, our system replaces these lines with the data points at the ends of the lines at the top and bottom of the image, Fig. 4(b). Now there are only two parameters to define the line, the upper and lower horizontal coordinates. Thus, this approach provides a reduction of computational costs in searching and any calculations. So, the system stores the following parameters:

$$S_a = \alpha + \beta \tag{2}$$

$$d = r - l \tag{3}$$

which S_a is the sum of the angles α and β, and d is the difference between r and l, which are the horizontal coordinate on the image's base indicated in Figure 4(b).

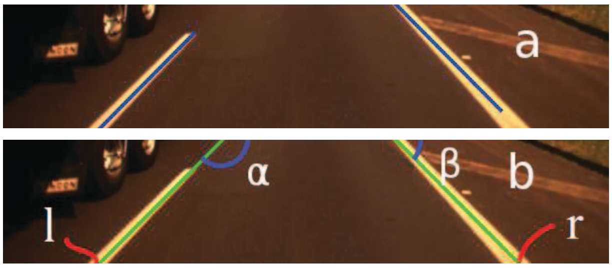

Fig. 4. (a) Normal lines in blue. (b) Replaced lines in green

When our system detects straight lines to the right and left vector, it assumes that the lane road markings are the pair of lines which has its d small as possible, provided that the value is larger than half the number of columns of the image. It is a reasonable assumption for the position of the camera installed in the experiment. Even if just right or left lines are detected, our system assumes that this detection is true and they are part of the lane. However, it still necessary to obtain the other one.

When the two road markers are detected, the sum of the tangent of the angles α and β (Fig. 4(b)) are stored with the coordinates of the points of each line, as well as the size in pixels d of the trapezoid's lower base formed by the lane.

C. Road Marker Estimation

Our system assumes the existence of lanes as it focuses highway environments. It generally has good demarcation throughout its length. Even so, there are many segmented road markers, not guaranteeing the detection of road markers in all frames of the ROI, i.e. , situations which have poor lighting or high contrast hindering the detection of road markers. In this way, it is necessary to estimate these not detected road markers using temporal data.(For this, time estimation was modeled that relies on data that is updated with a more recent detection.)

There are basically two cases to estimate the tracks: when one or both road markers are not detected. For this case, it is reasonable to assume that the lane position has not changed drastically on a highway, since there is not many sharp bends in this environment. Thus, when it is in the last context, the road markings are formed by the same lines detected in the last frame which there is detection.

Besides that, it is necessary to estimate road markings when just one road marker is detected. As seen in Section II-B, when the road markings are detected, the related parameters are stored. Based on the parameters α or β, l or r, S_a and d, it is possible to estimate a road marker using these parameters and from the detected one. If one road marker is detected, it needs that the S_a and d must be constants. Thus, it's possible to use the loop to search the road marker that makes those parameters as close as possible of the original values. However, if the estimation is happening for more than 15 frames (0.5 second), the system updates the values stored to improve the estimation. It might be important in segmented curvy road for which the estimation must be updated to avoid errors.

D. Vehicle localization

Since the road marking are detected and estimated, it is possible to obtain the lane connecting them to form a trapezoid (Fig. 5). Thus, the center of the lane, i.e., the ideal trajectory for that part is formed by the line through the centers of the bases of the trapezoid.

Based on the trapezoid, it is necessary to obtain the position of the car in lane. This is possible using, as the camera is stuck in a fixed place on the middle of the vehicle, the center of the image, since it is the corresponding half of the vehicle. Thus, we can get the difference in pixels of the ideal car position and the current position of the car and it will be an important information for its control. .

III. EXPERIMENTAL RESULTS AND DISCUSSION

The experiments were performed using ROS [14] platform and OpenCV [15] library. The former was used for collecting images from real situations. These images have been saved in logs in order to perform simulated tests by ROS. The images were collected in different types of road scenarios with a resolution of 640x480 pixels.

Fig. 5. (a) System result image. (b) Lane Ground Truth image.

We selected 127 images of the 12 minutes of video data set in different situations, including those where persistent system

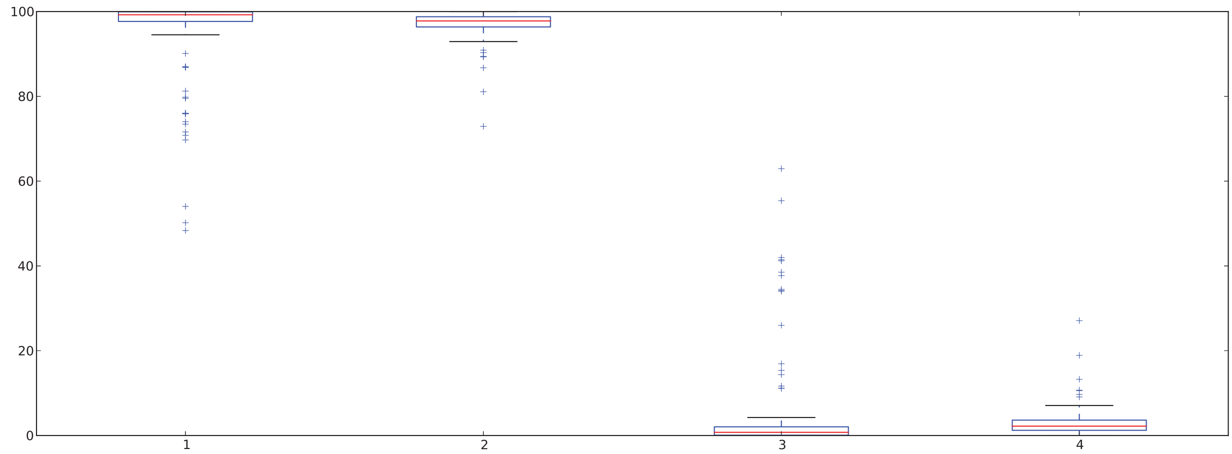

Fig. 6. Distribution of (1)Precision, (2)Recall, (3)False positive and (4)False negative, respectively.

errors were observed and 32 night shots. The system evaluated each image and they were compared with the groundtruth, which were made manually, for evaluation (see Fig. 5). The lane points are shown in white pixels and the black ones represent those outside the lane.

Furthermore, the precision and recall [16] of the dataset have been calculated for the ROI in each image. As a quantitative evaluation, the precision shown in Equation 4 is the relative probability that do not allow false positives. The recall in Equation 5 is the percentage chance of there is not allowing false negatives in the lane. The result of these analyses is shown in Table 1 and Figure 8.

$$PRE = \frac{T_p + F_p}{T_p} \tag{4}$$

$$REC = \frac{T_n + F_n}{T_n}, \tag{5}$$

, where PRE is the precision, REC is the recall, T_p true positive, F_p false positive, T_n true negative and F_n false negative.

TABLE I
RESULTS [%] OF LANE AREA EVALUATION

	PRE	REC	FPR	FNR
Lane detection	94.79	96.65	5.01	3.35

The results have considered satisfactory as showed in Fig. 6. This shows a distribution about the correleted evaluations using Precision and Recall. The red line represent the value of the paremeter calculated and the rectangle represents the range of the values inside the standard deviation bound. Furthermore, the blue plus signes show the poins that are considered as noise. The numerical result is showed in Table 1.

In Figures 7 and 8, images displayed with red lanes indicate that the center of the vehicle was 50 pixels or more distant from the center of a lane, indicating incorrect positioning of the vehicle. The yellow marked lanes indicate the vehicle is positioned slightly to the left and blue marked lanes similarly show a slightly right offset from the center of the lane.

In the following images, Fig. 7(a, b, c), the error is due to the fact the system has pre-processed based on high image contrast and that leads to true result. Thus, as the road marking was not detected on the left for the first case and on the right for the second, the system returned straight line on the car and on the sideband, respectively. Picture 8(f) shows an error in a night shot which has same problem than 7(b). In either case, errors presented are easily solved by implementing a probabilistic model or, in the case of an autonomous vehicle, integrating an obstacle detector.

The next three images show the robustness of the algorithm in different scenarios. Figure 7(d) shows a traffic situation, in which the test vehicle is next to a truck. Figure 7(e) shows segmented road markers encountered during system development. In the dataset worked, some road markers were spaced distantly, which hampered their detection numerous times and thus the estimation became important for these type

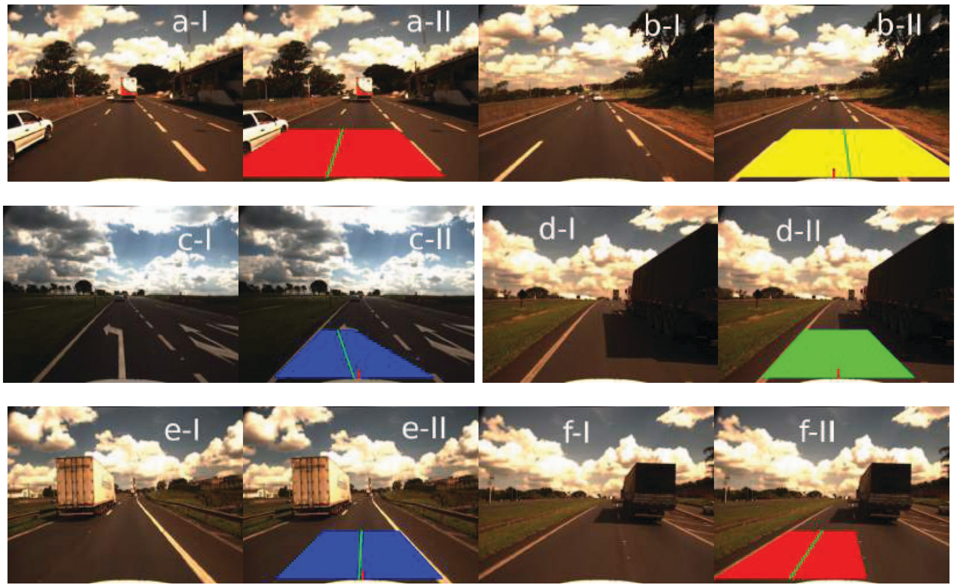

Fig. 7. Images result from system in normal conditions

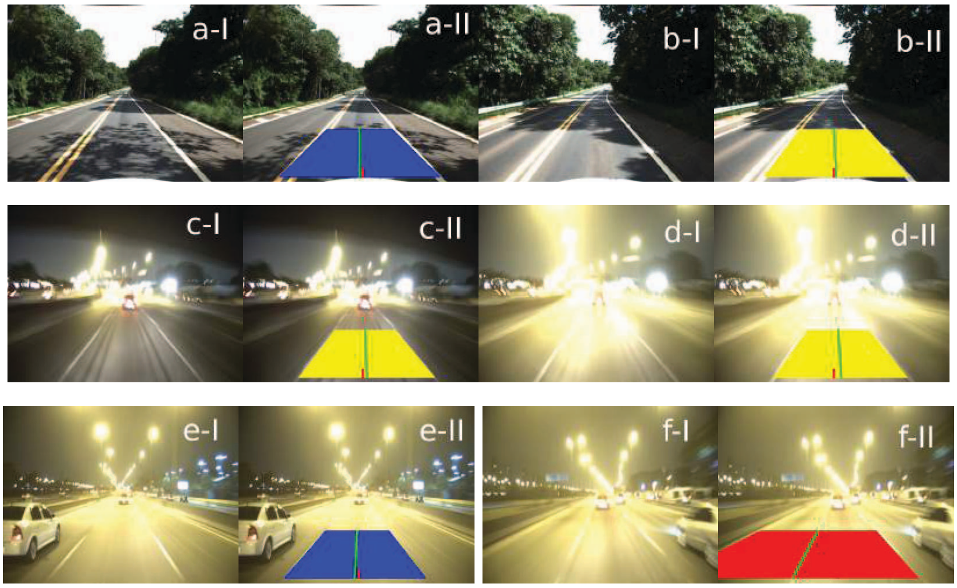

Fig. 8. Images result from system in night situations and strong shadows occurences

of roadways. As the system looks for ego-lanes, it would have problems with lane keeping control whenever the system fails to track them. Figure 7(f) shows an aggravating factor when the lane estimation is performed in situations with segmented road markers and the vehicle is moving laterally. In this situation, the system can successfully estimate the other line based on the detected one.

The algorithm achieves good results even in situations with plenty of shade and brightness, i.e. high contrast (see Figures 8(a,b)). We made this possible through a good preprocessing thresholding. In addition, the algorithm was robust enough to get a good estimated lane when preprocessing failed. Furthermore, obtaining lane in images at night is a significant result. In our data set, the camera was unable to get good images in this situation, as shown in Figures 8(c,d,e). However, the system was capable of detecting and estimating the ego-lane in this challenging environment. Even if a better camera is employed some unpredictable scenarios are likely to arise. Thus, our system could detect and estimate most ego-lanes in night environment.

Another key issue addressed here is the vehicle localization within the lane without the use of metric data. After the lane is detected, our system estimates a trapezoid and uses its base to locate the vehicle. As this line is horizontal and the number of pixels relate to the lane size, we conclude the center of this line will be the center of the lane. From this, it was possible to obtain the localization of the vehicle in the lane. This is useful information obtained from a perspective image to be used in the control and navigation of the vehicle.

IV. CONCLUSION AND FUTURE WORK

In this paper, it is proposed a lane detector and estimator based on perspective image, efficient image processing, Probabilistic Hough Transform and road markers estimation. The results indicate the system is robust to the variation of illumination (including night time) and lane types. We provide an alternate solution to the Bird's Eye View approach and avoid any prior calibration dependency. Our system needs just to set up the ROI(the position only, the size is constant) in front of the car, which is generally in the bottom of the image. Therefore, the cameras can be quickly mounted on autonomous vehicles or assisted steering vehicles. Although we do not take a metric approach in our algorithm, it can still contribute to vehicle control as pixels and distance are correlated.

Future work includes a probabilistic model to make this system more robust to highway side road markings and other types of road information. We are also looking to validate the lane detection algorithm for autonomous vehicles without dependency on Bird's Eye View or a sensor system in varied scenarios, including challenging ones. In consequence, the system will have low cost and easy setup and use.

ACKNOWLEDGMENT

The authors acknowledge the support granted by FAPESP, process 2014/01465-0.

REFERENCES

[1] Aly, Mohamed. *Real time detection of lane markers in urban streets.* Intelligent Vehicles Symposium, 2008 IEEE. IEEE, 2008.

[2] Liu, Guoliang, F. Worgotter, and Irene Markelic. *Combining statistical hough transform and particle filter for robust lane detection and tracking.* Intelligent Vehicles Symposium (IV), 2010 IEEE. IEEE, 2010.

[3] Borkar, Amol and Hayes, Monson and Smith, Mark T.. *A novel lane detection system with efficient ground truth generation.* Intelligent Transportation Systems, IEEE Transactions on 13.1 (2012): 365-374.

[4] Mallot, Hanspeter A. and Blthoff, Heinrich H. and Little, J.J. and Bohrer, Stefan. *Inverse perspective mapping simplifies optical flow computation and obstacle detection.* Biological cybernetics 64.3 (1991): 177-185.

[5] Fischler, Martin A., and Robert C. Bolles. *Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography.* Communications of the ACM 24.6 (1981): 381-395.

[6] Van Der Merwe, Rudolph and Doucet, Arnaud and De Freitas, Nando and Wan, Eric. *The unscented particle filter.* NIPS. 2000.

[7] Dahyot, Rozenn. *Statistical hough transform.* Pattern Analysis and Machine Intelligence, IEEE Transactions on 31.8 (2009): 1502-1509.

[8] Welch, Greg, and Gary Bishop. *An introduction to the Kalman filter.* (1995).

[9] Zhao, Kun, et al. *Zhao, Kun and Meuter, Mirko and Nunn, Christian and Muller, D. and Muller-Schneiders, S. and Pauli, Josef.* Intelligent Vehicles Symposium (IV), 2012 IEEE. IEEE, 2012.

[10] Lin, Qing and Han, Youngjoon and Hahn, Hernsoo. *Real-time lane departure detection based on extended edge-linking algorithm.* Computer Research and Development, 2010 Second International Conference on. IEEE, 2010.

[11] Fernandes, Leandro C. and Souza, Jefferson R. and Pessin, Gustavo and Shinzato, Patrick Y. and Sales, Daniel and Mendes, Caio and Prado, Marcos and Klaser, Rafael and Magalhaes, Andres Chaves and Hata, Alberto and Pigatto, Daniel, and Branco, Karlinka Castelo, and Junior, Valdir Grassi, and Osorio, Fernando S., and Wolf, Denis Fernando. *CaRINA Intelligent Robotic Car: Architectural Design and Applications.* Journal of Systems Architecture (2014).

[12] OTSU, Nobuyuki. *A threshold selection method from gray-level histograms*. Automatica, v. 11, n. 285-296, p. 23-27, 1975.

[13] Kiryati, Nahum and Eldar, Yuval and Bruckstein, Alfred M. *A probabilistic Hough transform.* Pattern recognition 24.4 (1991): 303-316.

[14] Jason M. O'Kane, *A Gentle Introduction to ROS* . Available at http://www.cse.sc.edu/ jokane/agitr/, 2nd ed. vol. 3, J. Peters, Ed. New York: McGraw-Hill, 2013, pp. 1564.

[15] Bradski, Gary, and Adrian Kaehler. *Learning OpenCV: Computer vision with the OpenCV library.* O'Reilly Media, Inc., 2008.

[16] Buckland, Michael K., and Fredric C. Gey. *The relationship between recall and precision.* JASIS 45.1 (1994): 12-19.