NON-INTRUSIVE SYSTEM TO DETECT THE GAZE DIRECTION USING ARTIFICIAL NEURAL NETWORKS

HELTON M. PEIXOTO, ANA M. G. GUERREIRO, ADRIÃO D. D. NETO

Federal University of Rio Grande do Norte - UFRN
Postal Code 1655, CEP: 59072-970, Natal, RN, Brazil
Department of Computer Engineering and Automation - DCA
Phone: +55-84-3215-3881
E-mails: helton.maia@gmail.com, anamaria@dca.ufrn.br, adriao@dca.ufrn.br

***Abstract*—** The vision has many sensors responsible for capturing information that is sent to the brain. The gaze reflects the attention, intention and interest of the brain towards the outside world. Therefore, the detection of the gaze direction is a promising alternative for the simulation programs, virtual reality applications and human–machine special communication. Cheaper devices to capture images and increase the power processing of personal computers motivate studies that allow human-machine interactivity. The application of techniques to detect the gaze direction has the possibility of improving significantly the interaction between people with motor deficiency and personal computers. The objective of this work is to develop a system that uses non-intrusive techniques of digital image processing and neural networks for recognizing patterns in the gaze direction**.** The results show that the standards used: up, down, left and right can be classified as satisfactory, above 83% for all cases**.**

Keywords— Gaze direction, human–machine interaction, recognizing patterns, artificial neural networks.

***Resumo*—** A visão tem muitos sensores responsáveis pela captação das informações que são enviadas para o cérebro. O olhar reflete a atenção, intenção e interesse do cérebro em relação ao mundo exterior. Portanto, a detecção da direção do olhar é uma alternativa promissora para programas de simulação, aplicações de realidade virtual e da comunicação homem-máquina em especial. Dispositivos mais baratos para captar imagens e aumentar o poder de processamento dos computadores pessoais motivam os estudos que permitem a interatividade homem-máquina. A aplicação de técnicas para detectar a direção do olhar tem possibilitado melhoras significantes para interação entre as pessoas com deficiência motora e computadores pessoais. O objetivo deste trabalho é desenvolver um sistema que utiliza técnicas não-intrusivas de processamento digital de imagens e redes neurais para o reconhecimento de padrões da direção do olhar. Os resultados mostram que os padrões utilizados: em cima, embaixo, esquerda e direita podem ser classificados de forma satisfatória, acima de 83% para todos os casos.

Palavras-chave— Direção do olhar, interação homem-máquina, reconhecimento de padrões, redes neurais artificiais.

1 Introduction

With the technological growth of the past years, computers have been part of the daily life of millions of people. Developing resources for interaction between people with special needs and the machine would be a practical way to reduce the obstacles caused by the deficiency and to make possible the social inclusion of such people (Corno, 2002; Spivey, 2005).

Among the five senses of perception, the vision is the one which sends the largest volume of information to the brain. Thus, the detection of the gaze direction is a promising alternative for communicating with the machine, besides being a natural and faster form of communication in comparison to conventional forms of communication that humans have.

The eye tracking is the technique used to detect the position the person is looking at. This concept is based on focusing the user's eyes for estimating their gaze direction. This estimate can be made both in 3D space, by determining a vector corresponding to the user's line of vision, as well as in 2D space, by determining the point observed on a particular area of interest (a panel or the screen of the monitor, for example) (Morimoto, 2000).

Several systems that allow human-machine interaction through human vision have been developed. Among the various investigations on this subject, two approaches evolve in parallel: the intrusive and non-intrusive techniques, each of them presenting very specific characteristics.

For the eye tracking, most techniques require the use of special equipment, such as electrodes positioned near the eyes, contact lenses and helmets (Moretto, 2004; Pistori, 2003).

Techniques based on computer vision circumvent this limitation by estimating the gaze from the processing of images captured from the face or the individuals' eyes (Tavares, 2000). Typically, we search for the images' characteristic points, as the pupil, the iris, the sclera, or even the reflections generated by light sources, and use these as references in determining the eye direction (Morimoto, 2000).

Generally, cameras with infrared light-emitting diodes are used in these systems, allowing the detection of human pupils, in addition to cameras camcorders with auto focus. Despite the good results which can be achieved, such devices have a high cost, making it difficult for widespread use. Intrusive devices come into direct contact with the user eye or skin, which may cause some discomfort and restrict

their time of usage (Moretto, 2004). The quality of modern webcams, associated to its low cost, is an interesting alternative to capture images focused on systems to detect the gaze direction. This work presents a form of human-machine interaction, through human vision and make use of techniques such as digital image processing in conjunction with artificial neural networks to classify the gaze direction. The images were captured by a simple camera - Model Logitech QuickCam Communicate STX, 1.3 Mpixels – based on the FCC (Federal Communications Commission) standard. The results were obtained quickly through a PC Intel Celeron M, 1.6 GHz, 2 GB RAM, Linux Ubuntu 9.04 OS.

2 Eye Detection

The techniques of digital image processing used in this study are used to detect the region's eyes in a digital image. The flow chart shown in Fig. 1 illustrates the architecture of the implemented system.

Initially, the image captured from a webcam, in order to be processed, adjusted in size, filtered and changed into binary elements. The system developed in this work was implemented with MATLAB 2008b software for acquisition and image processing. After all these steps, a database is created, based on the most relevant information, especially found on the face and eyes. Using this database, it is possible to make a selection of attributes, which are important for the training and classification of the gaze direction. After being trained, the neural network is able to classify the gaze direction in real time.

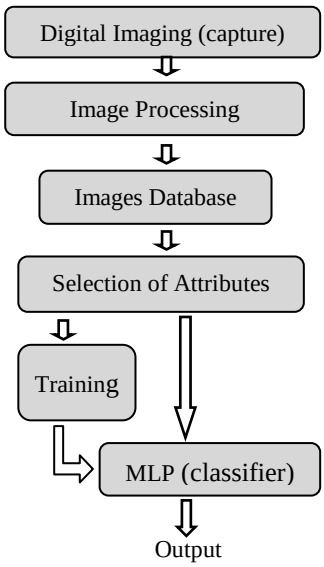

Fig. 1. Architecture of the system implemented.

2.1 Acquisition

The analog (real) image is represented by a two-dimensional light intensity function *f* (x, y), which needs to be sampled and quantized for scanning. Sampling means to discretize the space coordinates (x, y) in a matrix of elementary points, where each item is known as a pixel. The quantization is the representation of each pixel by a value that indicates the intensity of brightness, called of *grayscale*. The amount of gray levels depends on the amount of bits used in the representation of each pixel.

Some non-intrusive systems used to detect the gaze direction utilize a preliminary process of adjusting the values for each user, which is called *calibration*. These values are characteristic for different users, for example, the distance between them and the camera (Moretto, 2004). In this work, we used the scenario shown in Fig. 2. It is observed that the user is in acorrect ergonomic position; the distance between the eyes and the PC screen is about 60cm. Small variations in the distance between user and webcam, as well as some changes in the brightness of the environment have been considered for the acquisition of images.

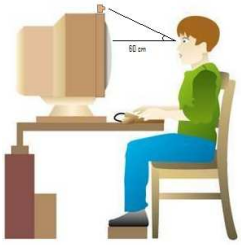

Fig 2. Scenario used for the capture of images.

2.2 Conversion to grayscale

Typically, the image captured by a camera is colorful and saved in RGB format, which represents the spectral components of the primary colors: red, green and blue, respectively. However, the image in RGB format turns the process more complicated, since some irrelevant highlights appear in certain parts of the image boundaries. This is due to the different levels of intensity for the components R, G and B, resulting in a change of their relative intensities (Gonzalez, 2002). Therefore, it is necessary to convert the image to the grayscale. When converting to grayscale with 256 levels, a translation of the original colors to intermediate color levels between black (plan origin) and white (the most distant point in relation to the plan origin) occurs. The conversion is done using (1), which takes into account each pixel of the image.

$$P'(x,y) = P_{red}(x,y)\cdot 0.299 + P_{green}(x,y)\cdot 0.587 + P_{blue}(x,y)\cdot 0.114 \quad (1)$$

where P' is the point of the new image in grayscale and P_{red}, P_{green} and P_{blue} are the points of the original RGB image (Gonzalez, 2002).

2.3 Face detection

The delimitation of the face area has as objective to eliminate the unnecessary information added by the

environment. In this work, the face detection is based mainly on the Successors Mean Quantization Transform (SMQT) used in image processing (Mikael, 2007; Mikael, 2005).

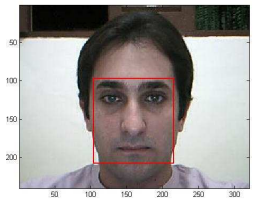

Fig. 3. Face detection.

2.4 Delimitation of the eye's area

Face detection had been explored so far in the last decade, and different algorithms were developed and consolidated. Thus, several of these studies have evolved to detect the eye, which is the goal of many current researches. In (Huchuan, 2007), a method for the eye detection, based on rectangle features and pixel-pattern based texture feature (PPBTF) is proposed. In (Yepeng, 2007), a method to locate eyes from face images is presented, based on multi-cue facial information. Using color characteristics is a useful way to detect the eyes according to (Jalal, 2008). The detection of the eyes used in this work is an algorithm of low computational cost, which is divided into two parts. Fig. 4 shows the beginning of the process

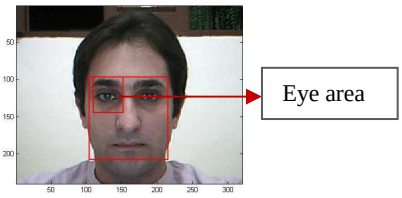

Fig. 4. Detection of the eye's area.

Based on parameters of facial geometry, an area is defined, which possibly represents the eye and eyebrow locations in addition to extra facial information. Following from that, an algorithm (Haralick, 1992) is employed to detect the objects. This algorithm allows a separation between the eye and other objects which occasionally belong to that geometrically defined area.

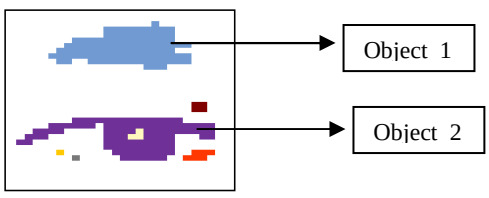

Fig 5. Identification of objects inside the eye area.

Considering the two largest objects found in Fig. 5, it is observed that object 1 is the eyebrow, whereas object 2 is the eye.

Fig. 6 shows a photograph that accurate represents the eye detection.

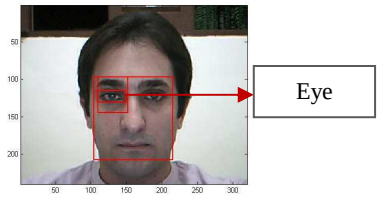

Fig. 6. Eye detection.

Finally, after finding the eye, it is found its center of mass, that will adjust the eye position in a box of previously defined size. The information contained in the box will be used later by the neural network, not only for the training database but also for the pattern classification.

Fig. 7. The eye center of mass.

3 Classification of the Gaze Direction

3.1 System classification

For the gaze classification, neural network techniques were used. The artificial neural networks are developments of computational applications capable of storing acquired knowledge to solve problems, therefore gaining new knowledge through experience (Grauman, 2001; Haykin, 2005). A similar and important study was performed by (Park, 2004), where the artificial neural networks are used in two stages. The first stage is the use of a Support Vector Machine (SVM) for the face detection, and the other is the use of a Multilayer Perceptron (MLP) in order to detect the eye gaze position. As well as this work, the investigation shown in (Park, 2004) enables the evaluation of the advantages of using neural networks to obtain a pattern recognition for image processing.

For the MLP-1 network, the Resilient Backpropagation (RPROP) was used as a training algorithm. RPROP is an efficient scheme of learning that runs the direct adaptation of the update of the synaptic weights, based on the information extracted from the local gradient. An important difference between this and the Backpropagation algorithm is the fact that the effort of adjusting weights is not affected by the behavior of the gradient, where a value (Δ_{ij}) of each synaptic weight is introduced and is responsible for determining only the amplitude of the updated weight. This suitable update evolves during the process of training, based on the local view of the cost function E, according to the rule of learning shown in (2) (Riedmiller, 1993). where $0 < \eta^- < 1 < \eta^+$ and t represents the number of training epochs. Following the rule of (2), whenever the partial derivative corresponding to the weight w_{ij}

changes its sign with respect to the previous iteration, it means that the last update was very high and the function passed by a local minimum.

$$\Delta_{ij}^{(t)} = \begin{cases} \eta^{+} * \Delta_{ij}^{(t-1)}, \; if \; \dfrac{\partial E^{(t-1)}}{\partial w_{ij}} * \dfrac{\partial E^{(t)}}{\partial w_{ij}} > 0 \\ \eta^{-} * \Delta_{ij}^{(t-1)}, \; if \; \dfrac{\partial E^{(t-1)}}{\partial w_{ij}} * \dfrac{\partial E^{(t)}}{\partial w_{ij}} < 0 \\ \Delta_{ij}^{(t-1)}, \; otherwise. \end{cases} \quad (2)$$

Thus, the update value Δ_{ij} decreases by a η^{-} factor. If the derivative maintains the same sign, the update value is increased in order to accelerate the convergence. In summary, if the derivative is positive (increasing the error), the weight is reduced by the update value; on the other hand, if the derivative is negative, the update value becomes positive, as shown in (3) (Riedmiller, 1993) :

$$\Delta w_{ij}^{(t)} = \begin{cases} -\Delta_{ij}^{(t)}, \; if \; \dfrac{\partial E^{(t)}}{\partial w_{ij}} > 0 \\ +\Delta_{ij}^{(t)}, \; if \; \dfrac{\partial E^{(t)}}{\partial w_{ij}} < 0 \\ 0, \; otherwise. \end{cases} \quad (3)$$

However, there is one exception: if the partial derivative changes its sign, that is, if the previous step is too large and the minimum is exceeded, the update of the weight Δw_{ij} is reverted:

$$\Delta w_{ij}^{(t)} = -\Delta w_{ij}^{(t-1)} \; if \; \frac{\partial E_{ij}^{(t-1)}}{\partial w_{ij}} * \frac{\partial E_{ij}^{(t)}}{\partial w_{ij}} < 0 \quad (4)$$

Because of this, the derivative must change its sign again in the next step, to prevent a further reduction of the update value. There should be no adjustment of the update value in the next step. A practical way to avoid this is to enforce $\frac{\partial E_{ij}^{(t-1)}}{\partial w_{ij}} = 0$.

The update values and the weights are modified only after the entire set of training is presented to the network, which is characterized by a lot or batch learning (Riedmiller, 1993). Initially, all the setting values are equal to the constant Δ_0, which is one of the parameters of RPROP. If Δ_0 directly determines the extent of the first set of weights, it can be chosen according to the magnitude of the initial weights, for example, $\Delta_0 = 0.1$ (Riedmiller, 1993). The choice of this value is not critical, since it is adjusted as the training occurs.

For the network training using the RPROP algorithm to avoid an excessive variation of weights, it is defined a parameter for the maximum adjustment $\Delta_{máx}$, which assumes a value of 50, as suggested in (Riedmiller, 1993). The increasing and decreasing factors are set at $\eta^{+} = 1.2$ and $\eta^{-} = 0.5$. These values are based on theoretical and empirical considerations. Thus, the number of parameters is reduced to two: Δ_0 and $\Delta_{máx}$ (Riedmiller, 1993).

In this work, the MLP-1 and MLP-2 networks makes use of the tangential sigmoid activation function and presents two input nodes, 28 hidden neurons and 4 neurons on output layer. Also in impirica adjustments were made in networks MLP1 and MLP2, to obtain the best performance.

4 Results

In this investigation, we used 400 images in two tests, each test consists of the following: 152 were used for training and 48 for generalization. The database created for the first test contains images of ten people and the second test, only one person. The following configuration was used: each image has been normalized and assigned as an input to the network, where it is used for future comparisons. The output was previously set with only four positions, which are called *left*, *right*, *down* and *up*. The training executed with the RPROP algorithm had a small run-time, requiring only 100 epochs to obtain satisfactory results, that are shown in Tables I and II. It is important to note that the following parameters were used as stop criteria in the training process: error = 10^{-5} and number of epochs = 100. For the training, a fixed input matrix of size 697 x 152 was used, where each of the 152 columns represented an image.

4.1 Images obtained

The images are obtained from the webcam connected on the top of the notebook screen and configured to capture images of 320 X 240 pixels. We obtained images of the webcam connected at the top of the screen and laptop configured to capture images of 320 X 240 pixels. At the beginning, the conversion to grayscale was done to the image captured; after that, the image went through a process of face detection: the image was cropped using the same values obtained in the previous procedure, trying to eliminate the unnecessary information to submit the network to the training. The images utilized as input for the network training represented only the eyes of users, for the four positions chosen, as seen in Fig. 8. It is worth to say that all these images were submitted to the steps described in the previous section.

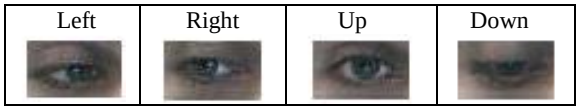

Fig. 8. Some of the images utilized in this work.

4.2 Generalization

In generalization, the recognition performace of the trained network is tested with data not see before.

To generalize the results of networks MLP-1 and MLP-2, we used two databases of entry consists of 48 images for each, not used in their training. After the training, the results of generalization were obtained using the MLP-1 and MLP-2 networks, which are shown in Tables I and II. The networks MLP-1 and MLP-2 together with an error of 10-4, the number of times equal to 100 and an execution time less than 4s. In Table II, where we have a set of more specialized training, we obtained the best results. The recognition of patterns (over 91%), with the exception of the "Down", which was 83.3%.

TABLE I
MESS MATRIX OF THE MLP-1 NETWORK

MLP-1	Up	Down	Left	Right	Hits
Up	10	1	0	1	83,3%
Down	0	10	1	1	83,3%
Left	0	2	10	0	83,3%
Right	1	0	0	11	91,7%

TABLE II
MESS MATRIX OF THE MLP-2 NETWORK

MLP-1	Up	Down	Left	Right	Hits
Up	11	0	0	1	91,7%
Down	0	10	2	0	83,3%
Left	0	1	11	0	91,7%
Right	1	0	0	11	91,7%

5 Conclusion

Detecting the gaze direction appears as a promising tool for the human-machine interaction, since it allows people with motor deficiencies to have access to the computer. This work proposed a series of systems to detect the gaze direction. A simple webcam for the capture, techniques for digital image processing for the eye detection and neural networks for image classification were used. The advantage of these systems is due to its low cost and simplicity. The results show the effectiveness of the techniques in this type of application. For the continuity of the work, we intended to determine the eye's area more efficiently, by improving the level of classification, even utilizing a larger set of training and accomplishing comparative tests using other kind of classifiers. Characteristics of automatic brightness control for image acquisition must be included in the system. In addition, some optimizations have to be performed in order to make the proposal more and more interesting and promising, by the usage of the eyes as a tool for the human-machine interaction.

Acknowledgments

We thank the Postgraduate Program of Electrical and Computer Engineering, Federal University of Rio Grande do Norte (PPgEEC, UFRN).

References

Corno F., Farinetti L., Signorile I., "A cost-effective solution for eye-gaze assistive technology". *ICME2002: IEEE International Conference on Multimedia and Expo*, Lausanne, Switzerland, 2002.

Gonzalez, R.C. Woods, R.E. "Digital Image Processing". Second Edition. Prentice-Hall, Inc. 2002.

Grauman K., Betke M., Gips J., Bradski G. R., "Communication via Eye Blinks – Detection and duration analysis in real time". *Proc. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Lihue, HI, Vol. 1 pp.:I-1010 - I-1017 2001.

Haralick M. R., Linda S. G., "Computer and Robot Vision", Volume I, Addison-Wesley, pp. 28-48, 1992.

Haykin S., N*eural Networks A Comprehensive Foundation.* Second Edition, 2005.

Huchuan L., Wei Y. Z., Del Y., "Eye detection based on rectangle features and pixel-pattern-based Texture features". Department of Electronic Engineering, Dalian University of Technology, 2007.

Jalal N. A., Sara K., Hamid P. R. "Eye Detection Algorithm on Facial Color Images". Department of Computer Engineering, Ferdowsi University of Mashhad and Department of Computer Engineering, University, 2008.

Mikael N., Jorgenm N., Ingvar C., "Face Detection Using Local SMQT Features And Split Up Snow Classifier". Blekinge Institute of Technology. 2007.

Mikael N., Mattias D., Ingvar C., "The Successive Mean Quantization Transform". Blekinge Institute of Technology. School of Engineering, 2005.

Moretto E. G., "Rastreamento da posição dos olhos para detecção da direção do olhar". Universidade Católica Dom Bosco, 2004.

Morimoto C. H., Koons D., Amit A., Flickner M., "Pupil detection and tracking using multiple light sources". *Image and Vision Computing*. The Netherlands, v. 18, n. 4, p. 331-336, 2000.

Park K. R., "Real-Time Gaze Detection via Neural Network", *ICONIP* 2004, LNCS 3316, pp. 673–678, 2004.

Pistori H., Neto J. J., "Utilização de tecnologia adaptativa na detecção da direção do olhar". *Spc Magazine*, Lima, Peru, v. 2, n. 2, 2003.

Riedmiller, M. and Braun, H. (1993). A direct adaptative method for faster backpropagation learning: The RPROP algorithm., in *Proceedings of the IEEE International Conference on Neural Networks*, San Francisco, USA, 586–591.

Spivey M., Pederson B.. "Offline tracking of eyes and more with a simple webcam". Departament of psychology, Cornell university. Uris Hall, Ithaca, NY USA, 2005.

Tavares J. M. R. S., "Análise de movimento de corpos deformáveis usando visão computacional", Faculdade de Engenharia. Julho, 2000. Universidade do Porto.

Woods A.W., "Important issues in knowledge representation" *Proceedings of the IEEE*, vol 74, pp. 1322-1336, 1986.

Yepeng G.. "Robust Eye Detection from Facial Image based on Multicue Facial Information". School of Communication and Information Engineering Shanghai University, 2007.