UNIVERSIDAD POLITÉCNICA DE PUEBLA

PROGRAMA ACADÉMICO DE POSGRADO

Automatic detection of drivers with fatigue using deep learning

TESIS QUE PARA OBTENER EL GRADO DE

MAESTRÍA EN INGENERÍA EN SISTEMAS Y CÓMPUTO INTELIGENTE

PRESENTA:

JORDI JAROMIL CRUZ MEDRANO

Director: Dr. Jorge de la Calleja
Co-Director: Dr. Hugo Jair Escalante

Juan C. Bonilla, Puebla, Mexico, Enero 2016.

El presente trabajo fue realizado en el Laboratorio de Investigación y Posgrado del Departamento de computo de la Universidad Politécnica de Puebla, ubicada en Tercer carril del Ejido "Serrano" S/N, San Mateo Cuanalá, Municipio Juan C. Bonilla, Puebla // CP 72640. Apoyo del CONACYT, Beca No. 704756, Programa de Maestría perteneciente al Programa Nacional de Posgrados de Calidad (PNPC-CONACYT)

UNIVERSIDAD POLITÉCNICA DE PUEBLA
MAESTRÍA EN INGENIERÍA EN SISTEMAS Y CÓMPUTO INTELIGENTE

Automatic detection of drivers with fatigue using deep learning

TESIS REALIZADA POR:

JORDI JAROMIL CRUZ MEDRANO

Aprobada por ... Abril 22, 2017.

Profesor	**(Firma)**
Dr. Jorge de la Calleja	....
Dr. Hugo Jair Escalante	....
Dra.Ma.Auxilio Medina	....
Dr. Antonio Benitez	....

Juan C. Bonilla, Puebla, Mexico, Enero 2016.

Universidad Politécnica de Puebla
Maestría en Ingeniería en Sistemas y Cómputo Inteligente

(Firma)

....
Jordi Jaromil Cruz
Medrano

Abstract

This document presents ....

Keywords

Machine learning, Deep learning, Convolutional networks, fatigue.

Agradecimientos

To my family ....

Jordi Jaromil Cruz Medrano

Contenido

Lista de Figuras

Lista de Tablas

I

Planteamiento del problema de investigación

Capitulo 1: Planteamiento del problema de investigación

1.1 Introducción

Esta es mi introducción ....

1.2 Objetivos

1.2.1 Objetivo general

To develop a prototype for detecting drivers with fatigue using deep learning to prevent automobile accidents.

1.2.2 Objetivos especificos

- To characterize images that show signs of people with fatigue using deep learning.
- To develop models to recognize at least three manifestations of tired drivers.
- To build an interface to manage the recognition system of fatigued drivers.

1.3 Justificación

Esto se justifica ...

II

Marco teórico

2.1 Fatigue

The Highway Traffic Safety Administration, estimates that 100,000 accidents occurred because of tired or distracted people. Out of these accidents, 1,500 people died, 71,000 people were wounded, and 12.5 millions dollars were lost in damages [2].

Fatigue could be described as the loss of energy such as physical or mental, and is a common complaint. We need to remember that it is a symptom and not a disease and is different than drowsiness, which is the necessity to sleep. Mental fatigue is a feeling of decreased cognitive performance. It can manifest as somnolence or lethargy. In any case, this could be dangerous when performing tasks that require constant concentration, such as driving [2].

While it can vary in each person, people with fatigue have some of the following complaints:

- Distraction and loss of concentration
- Loss of sensory functions and perception
- Slowness in making decisions
- Automatic movements

Fatigue symptoms are:

- Yawing
- Inclined head
- Nodding off
- Closed eyes
- Cramps
- Headache

In this thesis, we are going to work with the following symptoms: yawning, inclined head and closed eyes(figure 2.1).

2.2 Machine Learning

Machine learning is a subfield of Artificial Intelligence (AI) which gives computers the ability to improve their experience and make predictions or decisions on data[1, **?**]. Machine learning has three different learning styles:

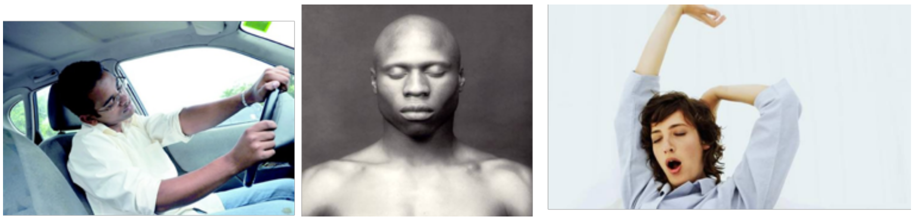

Figure 2.1: *Example of the symptoms inclined head, closed eyes and yawning.*

- **Supervised Learning**
  Input data has a known label and results [**?**].

- **Unsupervised Learning**
  Input data has no known label and does not have a known result [**?**].

- **Semi-supervised Learning**
  Input data is a mixture of labeled and unlabeled data [**?**]

Depending on the style selected, there are many kinds of machine learning algorithms grouped by similarity [**?**].

The learning process (figure 2.2), is as follows [1]:

- **Training set** is a set of images, texts, sounds, documents, data or information that could be classified.

- **Feature extraction** involves reducing the information necessary for describing a large set of information to obtain a new vectors set.

- **Labels** depend on the style selected (whether the vectors are labeled or not).

- **Machine learning algorithm** is the set of operations ordered to perform calculations in order to obtain a predictive model.

- **Predictive model** uses statistics measures to predict results based on a previous training. If we compare new data with the predicted model, this will be capable of determining what it is.

2.2.1 Deep Learning

Deep Learning is a new focus in Machine Learning. It promises general, powerful, and fast machine learning, moving us one step closer to AI. Deep learning is scalable performance that just keeps getting better as you feed it more data compared with older algorithms. Deep learning consists of several levels, which vary depending on the amount of data used for the training (figure 2.3) [**?**, **?**].

Feature extractions is an important factor for any machine learning process, but is even more critical in deep learning because the process of feature extraction uses high level

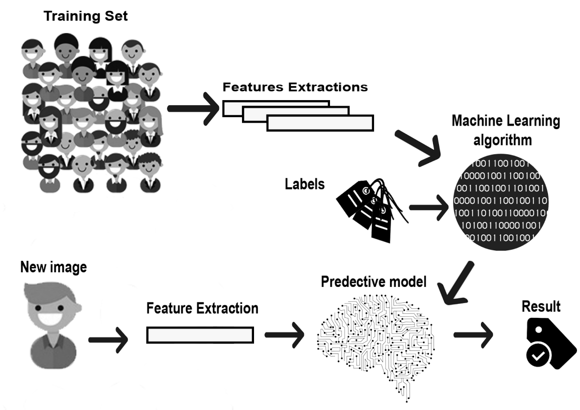

Figure 2.2: *Machine learning process [1]*

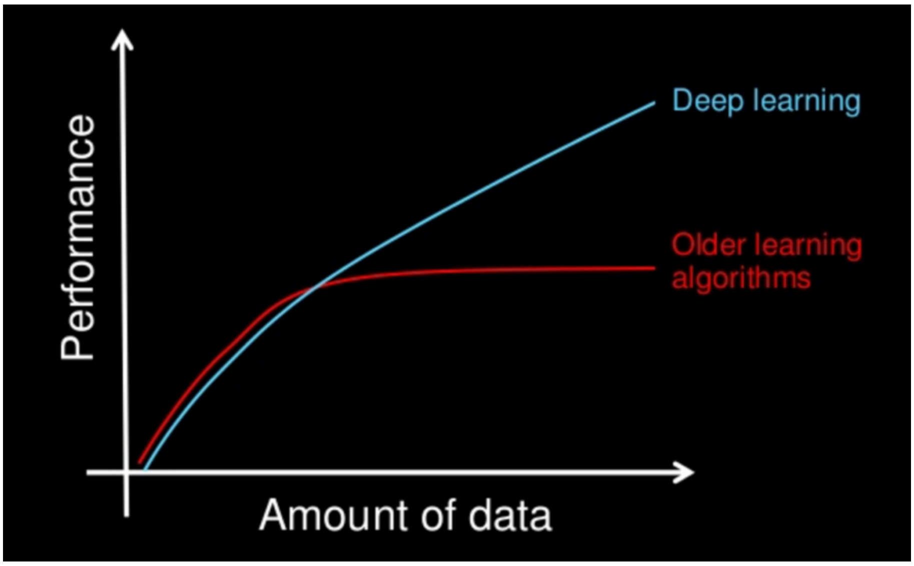

Figure 2.3: *How do data science techniques scale with amount of data? [**?**]*

features (figure 2.4) [**?**].
Giving way to new methods of training computers, deep learning needs a lot of information from training data. There are deep pure networks where the feature extraction goes hand in hand with the system training [**?**, **?**]. Models that have been incorporated from a low-level feature to a high-level are also considered deep networks [**?**, **?**] .

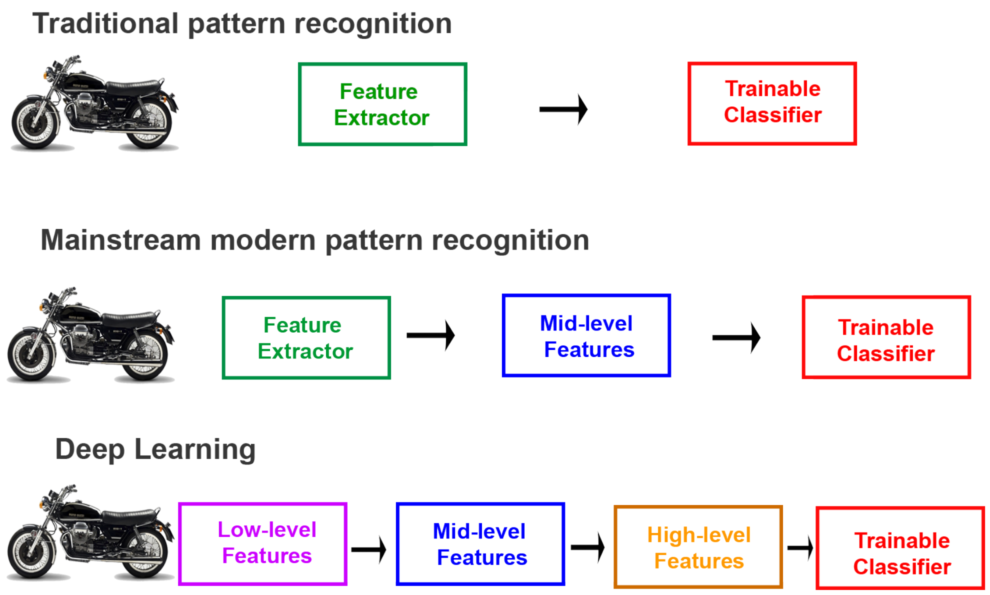

Figure 2.4: *Deep Learning Hierarchical Representations*

2.3 Artificial Neural Networks

Artificial neural networks are inspired by the structure and performance of biological neural networks. These contain layers of simple computing nodes that operate as non-linear summing devices(X). These nodes are interconnected by strong connections lines, and the weights are adjusted when data is presented to the network during the training process (figure 2.5). Successful training could result in a network that performs tasks such as predictions (Y). Many applications of artificial neural networks have been reported in computer engineering literature, and applications in medicine are growing [**?**, **?**].

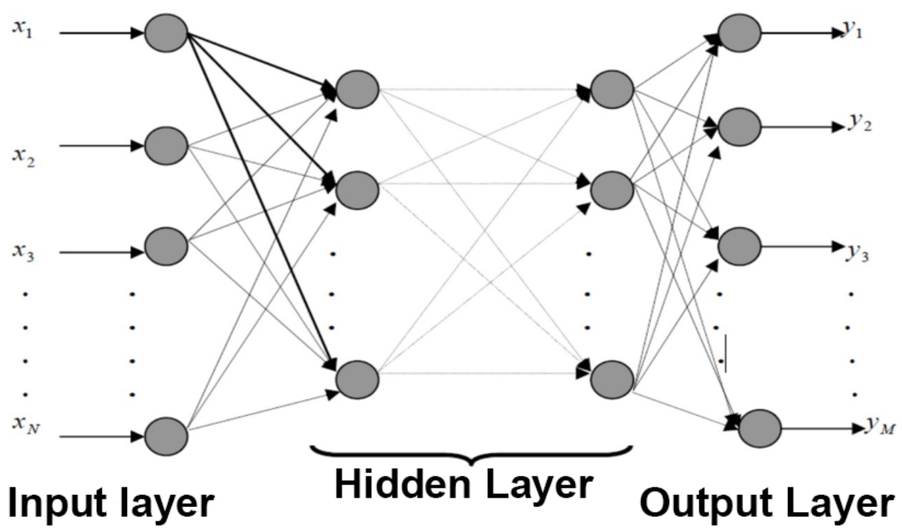

Figure 2.5: *Structure of an artificial neural networks*

2.3.1 Convolutional Networks

Convolutional Networks are more sophisticated structures used to solve difficult image pattern recognition, with their precise and simple architecture. They offer a simplified method for pattern recognition and future extractions. Convolutional neural networks use three basic ideas: local receptive fields, shared weights, and pooling [**?**, **?**].

- **Local receptive fields** This image is a matrix of n x n In the example (figure 2.6), we have a 5 x 5 matrix. In the algorithm, the operations will be performed in steps within a small region of x size. In the example, we use a area in order to obtain a new, smaller matrix (figure 2.7).

0	7	2	8	8
0	0	5	16	11
0	0	0	31	12
0	0	1	20	6
0	57	5	5	17

Figure 2.6: *Example of the resulting matrix of an image in convolutional neural networks*

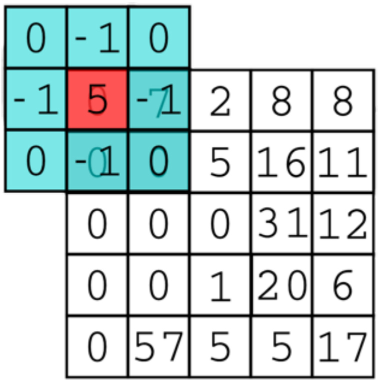

Figure 2.7: *It shows the obtaining of a value that represents a fragment in the new matrix*

- **Shared weights and biases** The network structure described so far can detect just a single kind of localized feature. For this process we need more than one feature map [**?**, **?**]. Each feature map is defined by a set of shared weights, and a single shared bias. The result is 3 different kinds of features (figure 2.8). We show just 3 feature maps in this example. However, in practice, convolutional networks may use more feature maps [**?**, **?**, **?**].

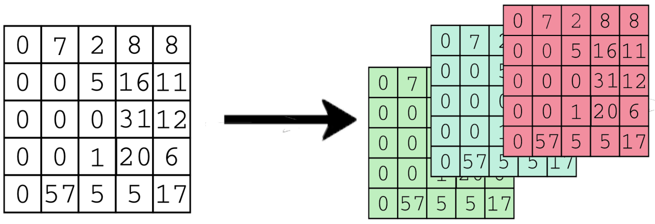

Figure 2.8: *8 We show how shared weights are used to obtain three new matrices with different abstraction characteristics.*

- **Pooling layers** What the pooling layers do is simplify information. This takes each feature map output from the convolutional layer and prepares a condensed feature map. The result is a matrix with half of its size(figure 2.9). If we put all the steps mentioned together we create a completed convolutional network (figure 2.10)[**?**, **?**, **?**].

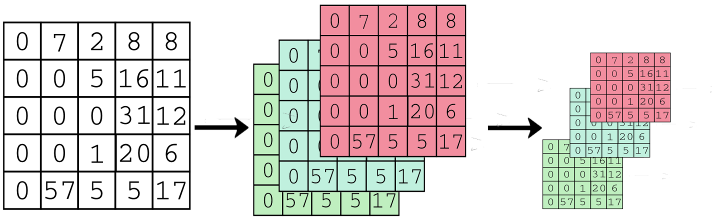

Figure 2.9: *It shows the obtaining of a value that represents a fragment in the new matrix.*

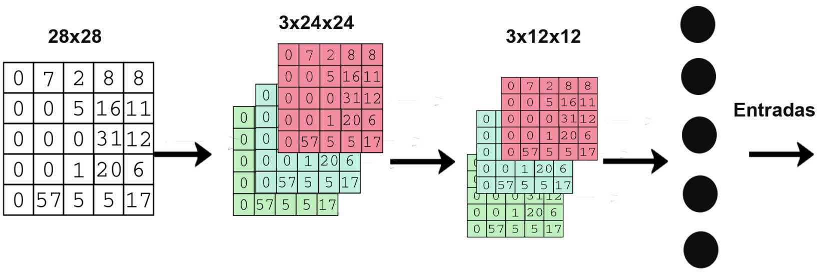

Figure 2.10: *Shows a complete convolutional network.*

2.4 Support Vector Machine

Support Vector Machine is a classifier formally using separating hyperplane.In other words, the algorithm outputs an optimal hyperplane which categorizes new examples (figure 2.11) [**?**].

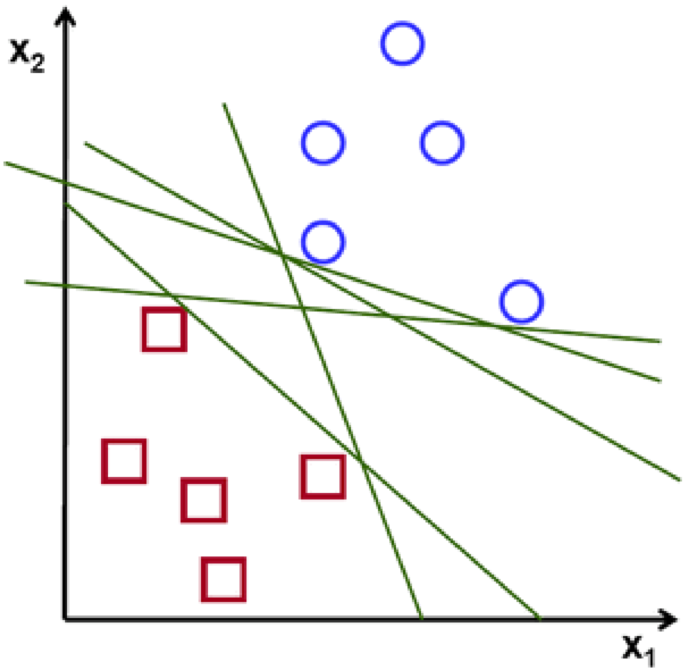

Figure 2.11: *Set of points separated by straight line.It's using support vector machine with different kernel*

SVM algorithm works looking for the hyperplane that gives the largest minimum distance to the training examples named Margin. Therefore, the optimal hyperplane maximizes the margin of the training data (figure 2.12) [**?**]. Other kinds of SVM is done by

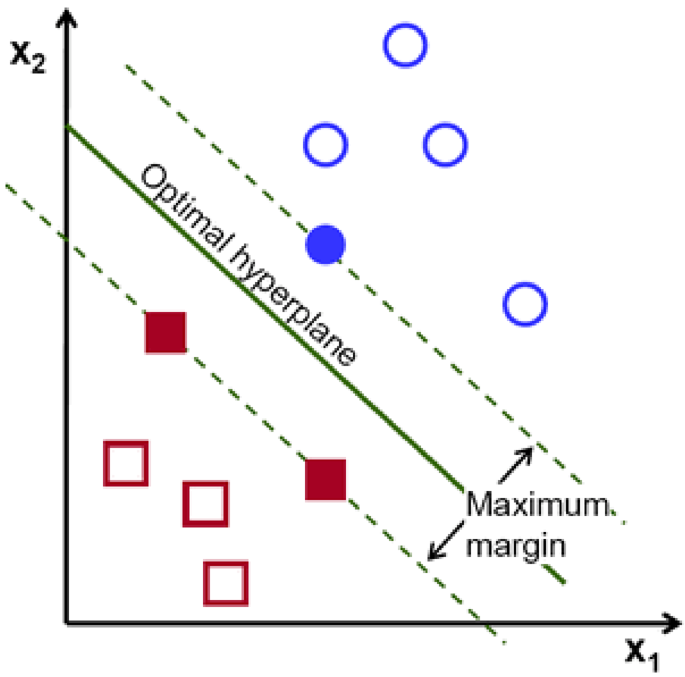

Figure 2.12: *The optimal hyperplane determinate by maximum margin*

transforming the problem using some linear algebra changing the kernel in the algorithm. The other kernels are:Linear Kernel, Polynomial Kernel, Radial Kernel. This gives to SVM the opportunity to classify with hyperplane more nonlinear even in a circle form (figure 2.13) [?].

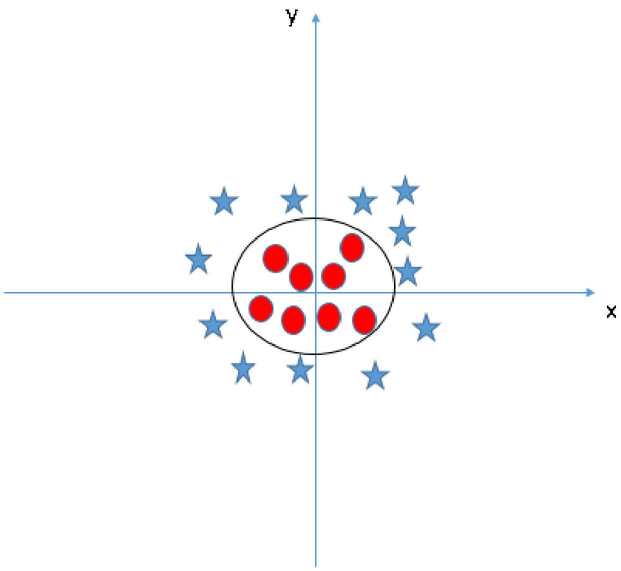

Figure 2.13: *Example of a Polynomial kernel*

2.5 Computer Vision

Computer vision is a subfield of Artificial Intelligence that gives computers the ability to see. It has to take into account the hardware and software limits. This uses techniques, algorithms and theories to accomplish a visual understanding [?, ?]. Computer vision is made up of different elements such as memory, recuperation of information, reasoning, estimation, recognition, coordination and other senses and consists of diverse tasks (figure 2.14) [?, ?].

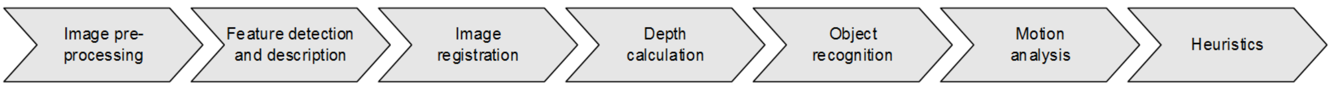

Figure 2.14: *Feature extraction of a image.*

- **Image pre-processing** includes tasks such as noise reduction, normalizing color and gamma values among others.
- **Feature detection and description** identifies important regions in the image that can be shrunk instead of the full-sized image, reducing computational complexity.
- **Image registration** aligns multiple images for simplifying pixels comparisons

- **Depth calculation** improves the performance of algorithms that require an understanding of three-dimensional space such as 3D model reconstruction.
- **Object recognition** identifies groups of pixels or features that could represent objects.
- **Motion analysis** extracts from multiple frames of video in order to predict the trajectory of objects.
- **Heuristics** facilitate decision-making with previous information in a split-second.

2.5.1 Feature Extraction

Feature extraction applies functions that look for attributes that represent and describe the object of interest.It is the first step for pattern recognition and could be used to assign the object a determined class.[**?**, **?**].
Feature detection transforms an image from a large set of pixels into a reduced representation as a feature vector (figure 2.15) [**?**, **?**, **?**].

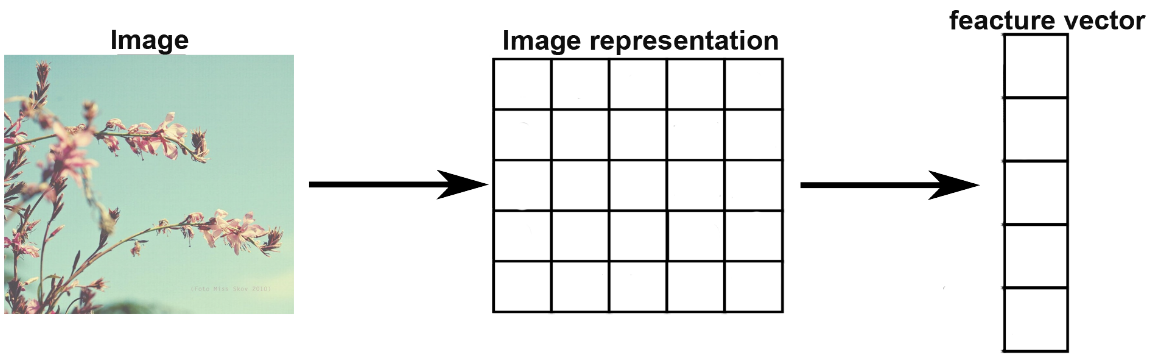

Figure 2.15: *Typical computer vision processing.*

Face detection is growing deeper and deeper into the feature extraction area.Many algorithms exist to detect faces but haar-like features has been highlighted for its velocity and efficiency in face image recognition[**?**, **?**].

2.5.1.1 Haar-like features

Haar-like features are rectangular regions that are placed in a specific location. Next, these features sums up the pixel intensities in each region and calculate the difference between them. The difference is then used to categorize subsections of an image by looking for something specific. It is important to remember that this algorithm works by appearance [**?**, **?**]. In human faces, commonly, the eye area is darker than the areas on the cheeks. Haar-like features for face detection work in rectangular areas above and between the eye and cheek regions [**?**].

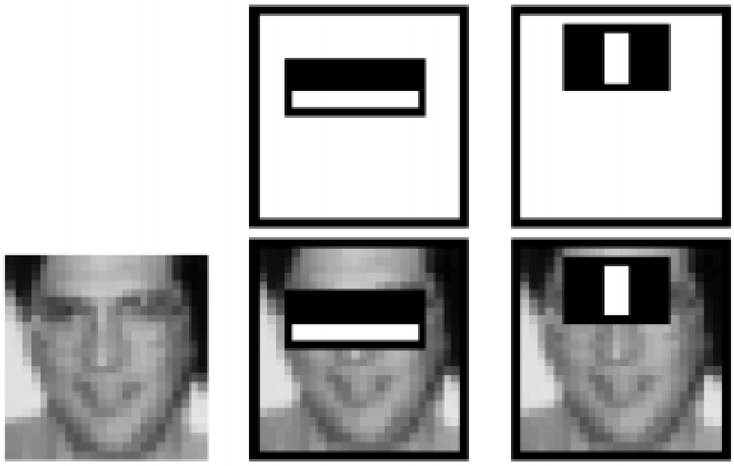

Figure 2.16: *Haar-like Features areas.*

2.6 Literure Review

In 2002,[?] Singh Sarbjit and Papanikolopoulos Nikolaosporpose detected people with fatigue by finding and tracking the location of the pupil. They used a gray scale model to determine whether the eye was open or closed. The result was a model that apparently works well. However, there were some problems. Under these circumstances, the system was able to detect prolonged eye blinks 95% of the time, and it produced occasional false alarms.

In 2007, [?] Xiao Fan, Bao-Cai Yin and Yan-Feng Sun proposed to locate and track a driver's mouth movement using a camera to detect people yawing.They used garbor for feature extraction and Latent Dirichlet Allocation for training the model. They obtained an average recognition rate of 95%.

In 2008, [?] Saradadevi Mandalapu and Bajaj Preeti proposed a method to locate and track drivers' mouths using Viola-Jones and the Support Vector Machine. They had as a result 81% accuracy.

In 2009, [?] Hu Shu Han and Zheng Gangtie tried to perform a model to detect drowsiness. It employed the Support Vector Machine. They got 86.67% accuracy on average during the trials to detect when the subject was 'sleepy'.

In 2011, [?] Zhao Chunlin, Zheng Chongxun, Zhao Min, Tu Yaling and Liu Jianping explored the possibility to detect fatigue using electroencephalographic signals (EEG) as data and the Support Vector Machine for training. They obtained 81.64% of accuracy in their investigation.

In 2011, [?] Coetzer Reinier C and Hancke Gerhard P developed a monitoring system, centered around the tracking of drivers' eyes. It was using the following techniques: Artificial Neural Networks (ANN), Support Vector Machines (SVM) and Adaptive Boosting (AdaBoost). They obtained 95.5% accuracy with SVM, 97.7% accuracy with AdaBoost and

96.1% accuracy in their results.

In 2014, [**?**] Jiao Yingying, Peng Yong, Lu Bao-Liangc, Chen Xiaoping, Chen Shanguang and Wang Chunhui explored a way to detect fatigue using Slow Eye Movement (SEM) data.It used the Support Vector Machine and the Graph Regularized Extreme Learning Machine for training the models. They had as a result of 93% accuracy on average.

III

Metodología

Capitulo 3: Metodología

3.1 Create Database

Mi metodología ...

IV

Resultados

4.1 Resultados

Mis resultados fueron ...

V

Conclusiones

Capitulo 5: Conclusiones

5.1 Conclusiones

Aquí van mis conclusiones ...

Bibliografía

[1] T. M. Mitchell, *Machine Learning*, 1997.

[2] F. CEA, *El sueño y la fatiga en la conducción: ¿Cuáles son los hábitos de los conductores españoles?*, Jul. 2015, no. 1.