



An Adaptive Image Pre-processing System for Quality Control in Production Lines

Esteban Arroyo Esquivel

Relatório Final do Trabalho de Projecto apresentado à
Escola Superior de Tecnologia e de Gestão
Instituto Politécnico de Bragança

para obtenção do grau de Mestre em
Engenharia Industrial

Julho de 2012

An Adaptive Image Pre-processing System for Quality Control in Production Lines

Esteban Arroyo Esquivel

Relatório Final do Trabalho de Projecto apresentado à
Escola Superior de Tecnologia e de Gestão
Instituto Politécnico de Bragança

para obtenção do grau de Mestre em
Engenharia Industrial

Orientador:

Dr. Paulo Leitão

Co-orientador:

Dr. Jose Luis Lima

Julho de 2012

“Make everything as simple as possible, but not simpler.”

Albert Einstein.

To God, my family, girlfriend and friends!

Acknowledgments

This work has been financed by the EU Commission, within the research contract GRACE No. NMP2-SL-2010-246203 coordinated by Univ. Politecnica delle Marche and having partners SINTEF, AEA srl, Instituto Politécnico de Bragança, Whirlpool Europe srl and Siemens AG.

The author would like to thank Dr. Paulo Leitao and Dr. Jose Luis Lima for their guidance, supervision and support during the project development. Moreover, he extends his gratitude to Lorenzo Stroppa, Prof. Nicola Paone, Univ. Politecnica delle Marche and Instituto Politécnico de Bragança for their collaboration in the experimental stage of the project.

Abstract

Adaptive and self-optimized behaviours in automated quality control systems based on computer vision and hence on digital image processing, constitute an approach that may significantly enhance the robustness, efficiency and flexibility of the industrial systems.

This work presents the development of an adaptive software system designed for the pre-processing of digital images captured in industrial production scenarios. The system is capable of characterizing and distinguishing between different image scenes, evaluating their current global quality, determining the best enhancement strategy to be executed and auto-adapting its functional parameters according to the actual generated data.

Real images captured in a washing machines production line are presented and used to test and validate the system algorithms. Experimental results demonstrate the robust performance of the system, the image quality enhancements achieved and the improvement of the recognition effectiveness of the posterior processing algorithms.

The contribution of the present work consists in the implementation of a dynamic scene recognition algorithm, based on a neural network architecture and a parametric indexes extraction process, that allows the execution of customized quality correction strategies and it is capable of considering the system's history in order to self-adapt. With this project it is intended to incur in the automated industrial quality control field by addressing a novel approach on adaptive pre-processing systems for quality enhancement purposes.

Keywords: Adaptive systems, Image pre-processing, Industrial quality control, Scene recognition.

Resumo

Os comportamentos adaptativos e auto-otimizáveis nos sistemas de controlo de qualidade baseados em visão artificial e, portanto, em processamento de imagem, constituem uma abordagem que poderia aumentar significativamente a robustez, eficiência e flexibilidade das respostas dos sistemas industriais.

Este trabalho apresenta o desenvolvimento de um sistema de software adaptativo desenhado para o pré-processamento de imagens capturadas em ambientes de produção industrial. O sistema é capaz de caracterizar e distinguir entre diferentes imagens, avaliar a sua qualidade global actual, determinar a melhor estratégia de melhoramento por executar e auto-adaptar os seus parâmetros funcionais de acordo com a informação gerada.

Imagens reais capturadas numa linha de produção de máquinas de lavar são apresentadas e usadas para testar e validar os algoritmos do sistema. Resultados experimentais demonstram o rendimento robusto da arquitectura, os melhoramentos da qualidade da imagem atingidos e o incremento da efectividade de reconhecimento dos algoritmos de processamento posteriores.

A contribuição do presente trabalho consiste na implementação de um algoritmo dinâmico de reconhecimento de cenários, baseado numa arquitectura de rede neural e num processo de extracção de parâmetros, que permite a execução de estratégias de correcção de qualidade e é capaz de considerar a história do sistema com o objectivo de auto-adaptar-se. Com este projecto pretende-se incorrer no campo do controlo de qualidade automatizado focando uma abordagem novedosa em sistemas de pré-processamento adaptativo com propósitos de melhoramento de imagem.

Palavras Chave: Sistemas adaptativos, Pré-processamento de imagem, Controlo de qualidade industrial, Reconhecimento de padrões.

Resumen

Comportamientos adaptativos y auto-optimizables en sistemas de control de calidad basados en visión artificial y, por tanto, en procesamiento de imagen, constituyen un abordaje que podría aumentar significativamente la robustez, eficiencia y flexibilidad de las respuestas de los sistemas industriales.

Este trabajo presenta el desarrollo de un sistema de software adaptativo diseñado para el pre-procesamiento de imágenes capturadas en escenarios de producción industrial. El sistema es capaz de caracterizar y distinguir entre diferentes imágenes, evaluar su calidad global actual, determinar la ejecución de la estrategia de mejoramiento óptima y auto-adaptar sus parámetros funcionales de acuerdo con la información generada.

Imágenes reales capturadas en una línea de producción de lavadoras de ropa son presentadas y utilizadas para verificar y validar los algoritmos del sistema. Los resultados experimentales obtenidos demuestran un rendimiento robusto de la arquitectura, mejoras tangibles en la calidad de la imagen y un incremento de la efectividad de reconocimiento de los algoritmos de procesamiento posteriores.

La contribución del presente trabajo consiste en la implementación de un algoritmo dinámico de reconocimiento de escenarios, basado en una arquitectura de red neuronal y en un proceso de extracción de parámetros, que permite la ejecución de estrategias de corrección de calidad y es capaz de considerar la historia del sistema con el objetivo de auto-adaptarse. Con este proyecto se pretende incursionar en el campo del control de calidad automatizado enfocando un abordaje novedoso en sistemas de pre-procesamiento adaptativo con propósitos de mejoramiento de imagen.

Palabras Clave: Sistemas adaptativos, Pre-procesamiento de imagen, Control de calidad industrial, Reconocimiento de escena.

Contents

List of Acronyms	xv
1 Introduction	1
1.1 Problem Statement	2
1.2 Objectives	3
1.3 Work Scope	4
1.4 Methodology	5
1.5 Document Organisation	5
2 Theoretical Framework	7
2.1 Introduction	7
2.2 Computer Vision Systems	7
2.3 Digital Image Manipulation in Computer Vision	8
2.3.1 Image Pre-processing and Processing Methods	9
2.3.1.1 Histogram Manipulation: Equalization and Expansion	10
2.3.1.2 Noise Reduction	11
2.3.1.3 Edges Enhancement	12
2.3.1.4 Unsharp Masking and High Boost Filtering	13
2.3.1.5 Morphological Operations	14
2.3.1.6 Contours Detection	14
2.3.1.7 Hough Transform	15
2.3.1.8 Geometric Matching	17
2.3.1.9 Pattern Matching	17

2.4	Neural Networks	19
2.4.1	Concept and Brief History	19
2.4.2	Perceptron Neuron and Multilayer Neural Networks	20
2.4.3	Neural Network Learning Process	21
2.4.3.1	Supervised Learning	22
2.4.3.2	Unsupervised Learning	22
2.4.4	Back Propagation Feed-Forward Networks	23
2.4.4.1	Phases	23
2.4.4.2	Learning Modes	24
2.4.4.3	Enhancement Heuristics	24
2.4.5	Neural Network Design Work Flow	25
2.5	Vision-based Industrial Applications	25
2.6	Summary	29
3	System Architecture	31
3.1	Introduction	31
3.2	Task Organisation Scheme	31
3.3	The Scene Recognition Module	34
3.3.1	Parameters and Global Quality Functions	34
3.3.1.1	Penalization Criteria	37
3.3.1.2	Weights Vector Optimization	39
3.3.2	System Learning	39
3.3.3	Complex Scene Identification	41
3.3.4	Advantages of the Scene Recognition Implementation	42
3.4	The Adaptive Quality Enhancement Module	44
3.4.1	Correction Strategies	44
3.4.1.1	Contrast Enhancement Strategy	44
3.4.1.2	Luminosity Correction Strategy	45
3.4.1.3	Sharpness Enhancement Strategy	45
3.4.2	Optimal Enhancement Strategy Determination	45
3.4.3	Quality and Parameters Updating	48

3.5	Summary	49
4	Experimental Results	51
4.1	Introduction	51
4.2	First Experimental Stage: System Performance and Quality	51
4.2.1	Interconnection Test	56
4.2.2	Statistics Summary	57
4.3	Second Experimental Stage: Processing Recognition	58
4.3.1	Experimental Scene 1	59
4.3.2	Experimental Scene 2	64
4.3.3	Statistics Summary	69
5	Conclusions and Future Work	71
5.1	Conclusions	71
5.2	Future Work	72

List of Figures

1.1	Sequential Production Line Layout	3
2.1	Image Digital Acquisition, Pre-processing and Processing Procedures	9
2.2	Histogram Manipulation. a) Original b) Equalization c) Expansion.	11
2.3	Signal Derivatives Analysis for Zero Crossing and Edge Detection	13
2.4	Unsharp Masking Blocks Diagram	13
2.5	Morphological Operations. a) Original Image b)Enhanced Image by Opening	15
2.6	Contour Detection. Sobel Method a) Input Image. b) Processed Image [12]	16
2.7	Geometric Matching a) Template b) Inspection Image	18
2.8	Pattern Matching (PM) a) Template b) Inspection Image c) Rotation d) Scal- ing	18
2.9	Neuron Model Diagram	20
2.10	Multilayer Neural Network Diagram	21
2.11	Predictive Control Functional Diagram. Adapted from [38]	27
3.1	Image Pre-Processing Task Scheme	32
3.2	Image Pre-processing Architecture	33
3.3	Experimental Indexes Behaviour Test	38
3.4	Feed-Forward Backpropagation Neural Network Schematic. Adapted from [21]	40
3.5	Simplified Schematic of the Recognition Process	41
3.6	Image Histogram Matching Method	43
3.7	Simplified Schematic of the Adaptive Quality Enhancement Algorithm	47

4.1	Experimental Image Scenarios Analysed	52
4.2	Experimental Scenario 1. a) Input Image b) Enhanced Image	53
4.3	Experimental Scenario 2. a) Input Image b) Enhanced Image	53
4.4	Experimental Scenario 3. a) Input Image b) Enhanced Image	54
4.5	Experimental Scenario 4. a) Input Image b) Enhanced Image	54
4.6	Experimental Scenario 5. a) Input Image b) Enhanced Image	55
4.7	Experimental Scenario 6. a) Input Image b) Enhanced Image	55
4.8	Experimental Scenario 7. a) Input Image b) Enhanced Image	56
4.9	Experimental Scenario 8. a) Input Image b) Enhanced Image	56
4.10	Geometric Matching Template used in the Experimental Scene 1	59
4.11	Scene 1. Contrast and Sharpness Enhancement Test 1. a) Input Image b) Enhanced Image	60
4.12	Scene 1. Contrast and Sharpness Enhancement Test 2. a) Input Image b) Enhanced Image	60
4.13	Scene 1. Recognition Enhancement. Score Improvement in Contrast and Sharpness Test	61
4.14	Scene 1. Experimental Matching Scores. Contrast and Sharpness Test	62
4.15	Scene 1. Luminosity Enhancement Test 1. a) Input Image b) Enhanced Image	62
4.16	Scene 1. Luminosity Enhancement Test 2. a) Input Image b) Enhanced Image	63
4.17	Scene 1. Recognition Enhancement. Score Improvement in Luminosity Test .	63
4.18	Scene 1. Experimental Matching Scores. Luminosity Test	64
4.19	Geometric Matching Template used in the Experimental Scene 2	64
4.20	Scene 2. Luminosity Enhancement Test 1. a) Input Image b) Enhanced Image	65
4.21	Scene 2. Luminosity Enhancement Test 2. a) Input Image b) Enhanced Image	65
4.22	Scene 2. Recognition Enhancement. Score Improvement in Luminosity Test .	66
4.23	Scene 2. Experimental Matching Scores. Luminosity Test	66
4.24	Scene 2. Contrast and Sharpness Enhancement Test 1. a) Input Image b) Enhanced Image	67
4.25	Scene 2. Contrast and Sharpness Enhancement Test 2. a) Input Image b) Enhanced Image	67

4.26 Scene 2. Recognition Enhancement. Score Improvement in Contrast and Sharpness Test	68
4.27 Scene 2. Experimental Matching Scores. Contrast and Sharpness Test	68

List of Tables

- 2.1 Functional Modules of Human and Computer Vision 8
- 2.2 Morphological Operations 14

- 3.1 Boundaries of the Weight Constraint Optimization 39

- 4.1 Some Important Image Parametric Indexes Variations 57
- 4.2 Image Quality Enhancement Summary 58
- 4.3 Summary of the Matching Score Averages and Standard Deviations 69

List of Acronyms

ANN	Artificial Neural Network
AV	Artificial Vision
BMP	Bitmap format file
BP	Back Propagation
CCD	Charge Coupled Device
CT	Computer Tomography
CV	Computer Vision
DR	Dynamic Range
E	Entropy
GM	Geometric Matching
GMDH	Group Method of Data Handling
HAVNET	Hausdorff-Voronoi Network
HL	High Level
HS	High Saturation level
LL	Low Level

LM	Luminosity Mean
LMS	Least Mean Square
LS	Low Saturation level
MAS	Multi-Agent System
MLP	Multilayer Perceptron
NN	Neural Network
PC	Personal Computer
PCA	Principal Components Analysis
PDP	Parallel Distributed Processing
PM	Pattern Matching
SOM	Self-Organizing Maps
TF	Transfer Function
TN	Tenengrad
V	Variance
VLSI	Very Large Scale Integration

Chapter 1

Introduction

As a direct consequence of the industry's pursuit of efficient and low-cost production methods, in addition to the growing customer's high-quality product demand, research on alternative techniques have been taken worldwide. Approaches based on robotics, computer vision and artificial intelligence arise as promising perspectives to address this challenge.

Automated product inspection and quality control in manufacturing processes, constitutes one of the resulting alternatives of the research efforts and comes up like a modern strategy boosted by the modern industry. Using computer vision mechanisms, this kind of procedures can be performed as non-invasive tasks and allow the valuable substitution, in terms of process dynamics and efficiency, of repetitive supervision operations executed by human workers [2].

In manufacturing companies with a high volume of goods production, the quality control becomes a thorough labour. In many cases, the production process cannot be stopped or disturbed for external procedures, not even the supervision ones. Moreover, product inspection routines may imply high accuracy or present operational difficulties due to environmental reasons or the process itself. Under these circumstances, vision-based systems demonstrate an added value and substantial profits by means of high zoom-image capturing, precision and flexibility.

Disturbances of several natures in the assembly plants, may affect the pre-set image acquisition configuration of the cameras located along the process lines. This fact has a direct consequence in the image quality, i.e., blurring or luminosity problems. Pre-processing procedures can be executed in order to enhance the image quality and facilitate the posterior

tasks of feature extraction and image recognizing [54].

The complexity of the situation increases when considering the industrial environmental conditions, namely time-varying parameters, changing production layouts, and product modifications. These variables reveal the limitations of fixed-configuration systems, because tedious changes in the current configurations may be required. In this point, it is evident that an adaptive behaviour is desirable in the context of effectiveness and performance. Consequently, it comes up the conception of a robust Intelligent Manufacturing System [30] capable of facing actual changing situations [35] by a self-adaptive and self-learning conduct.

In this work, the development and testing stage of an adaptive image pre-processing approach for production lines are described. The designed architecture is provided with adaptive properties based on neural networks, features extraction and image correction techniques, which allow the system to dynamically recognize and enhance the quality of given pictures and increase the effectiveness of the posterior processing algorithms.

1.1 Problem Statement

The problem addressed in this work concerns the adaptive image quality enhancement of images captured in production lines with automated quality control purposes.

Production layouts focused in the current approach consist in work stations disposed along a conveyor line, in which the products travel being transformed by sequential tasks executions. Figure 1.1 illustrates a generic topology of the assembly layout focused.

Quality control stations are also placed strategically along the lines. Generally, these are composed by robotic actuators, digital cameras and control programs interacting altogether with the aim of extracting information of the process.

The supervision stations capture several pictures of the product, either by changing the robot orientation or the camera parameters (e.g., magnification, focus, aperture, exposure time and electronic gain). These acquisitions variants are required in order to perform the evaluation of diverse items according to a defined quality policy. Consequently, the resulting acquired images may be composed by different elements or present dissimilar parametric indexes. Then, in terms of image parametrization, those variants can be considered like independent image scenarios.

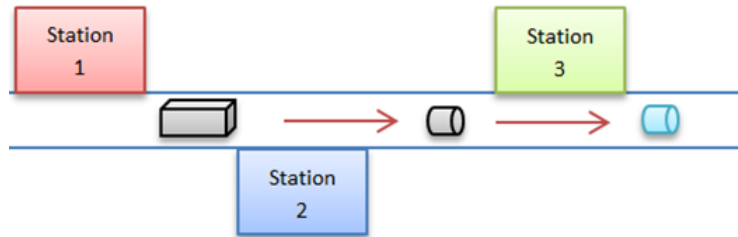


Figure 1.1: *Sequential Production Line Layout*

Naturally, the quality optimization of images belonging to different scenes requires the execution of customized strategies that fit each situation. The objective is to develop one software system capable of detecting different scenarios, self-adapt to the current circumstances and determining the optimal pre-processing strategy for the analysed image.

In the current work, real images corresponding to washing machines production lines are used. This issue endows the work with an applied and specific character and allows testing the system under a manufacturing layout that accomplishes the stated features.

1.2 Objectives

The general objective pursued in the present project is:

- To develop an adaptive image pre-processing system for the enhancement of digital images captured in industrial production scenarios.

Then, the successful accomplishment of the stated general objective implies the attainment of a series of specific objectives, namely:

- Apply digital image manipulation methods for quality enhancement.
- Implement a neural network scheme oriented to image recognition.
- Design a system architecture capable of distinguishing between different images scenarios, executing customized quality enhancement strategies and self-adapting.
- Codify the resulting architecture in an adequate programming language.

- Test and validate the performance of the implemented system by analysing real pictures proceeding from a washing machines production line.
- Analyse the recognition effectiveness enhancement by implementing an experimental detection algorithm.

1.3 Work Scope

The present project develops a functional adaptive image pre-processing architecture for quality control in industrial production scenarios, specifically washing machines assembly lines.

The final system version must be able to accurately analyse and recognize image scenes, evaluate the current global image quality and execute customized correction strategies for quality enhancements.

As an initial approach, the system architecture is to be sequential, executing pre-processing tasks on a single image at a time. Additionally, sequential image batch processing is to be addressed in the architecture.

The quality criteria to be implemented is based on a parameter extraction process and a global quality function evaluation. An adequate ensemble of parametric indexes must be defined in order to correctly characterized the image and estimate its global quality.

Enhancement strategies to be developed are to correct quality detractions relative the image contrast, sharpness and luminosity.

Adaptive behaviours and system's history analysis must be addressed in the design with the purpose of adapt and self-optimized to the current conditions of the work environment.

Resulting image quality enhancements must be quantified numerically and a significant improvement of effectiveness of the posterior processing algorithms due to the pre-processing execution is to be demonstrated.

The project remains open for future modifications such as parallel processing and the exploration of new parametric indexes for deep and detailed image descriptions and differentiations.

1.4 Methodology

In order to accomplish the objectives stated in the project, the steps presented below were followed:

- Establish an ensemble of parametric indexes and an evaluation criteria in order to quantify the quality of a given image.
- Define a pre-processing modular architecture and the respective component module functions.
- Endow the designed architecture with adaptive features and intelligent behaviours by means of neural networks fundamentals.
- State the correction strategies to be implemented for image quality enhancement and the image manipulation methods to be used in each one.
- Implement the designed system in an adequate programming language. As a first approach the implementation on MATLAB® is addressed.
- Test and evaluate the system performance and execution times under diverse scenes and with different image quality problems.
- Develop an interconnection of the the system with alternate tools or systems running on different platforms. Initially an interconnection with a image processing system running on LabVIEW® is explored.
- Design a test recognition algorithm in order to analyse the effectiveness enhancement of the identification methods, achieved due to the pre-processing stage. As a first approach, a recognition program is codified on LabVIEW® by using the NI Vision® utilities.

1.5 Document Organisation

The document is organised as follows: Chapter 2 introduces the problem's theoretical framework and presents the state of the art, addressing for this purpose, the topics of dig-

ital image manipulation, neural networks and vision-based quality control systems in the industry. Chapter 3 presents the system architecture focusing on the development of the scene recognition and the adaptive quality enhancement algorithms. Experimental results concerning the overall system performance and the image quality enhancement benefits are discussed in Chapter 4. Finally, Chapter 5 points out the conclusions and the future work.

Chapter 2

Theoretical Framework

2.1 Introduction

By means of the interaction of computer vision techniques and neural networks algorithms, it is pretended to implement smart systems conceived for optimizing solutions in an extended range of applications in the automated industry field. Several benefits related to the performance, efficiency and economical profitability of the process can be achieved by means of this implementation.

This chapter presents a series of concepts and background fundamentals related to the topics of digital image manipulation, neural networks and industrial quality control systems, base subjects of the study case addressed in the present work.

2.2 Computer Vision Systems

Computer Vision (CV) is a branch of artificial intelligence that studies the methods of acquisition, manipulation and analysis of images, which usually correspond to objects of the real world, with the aim of extracting and generating information capable of characterising and parameterising an image scene [36]. Vision-based models are inspired in the biological human vision system and pursue to simulate its capabilities and behaviours by means of electronic devices and software algorithms. Table 2.1 presents a comparison of the common

functionalities of the compared systems [23].

Table 2.1: *Functional Modules of Human and Computer Vision*

Task	Human Vision	Computer Vision
Visualization	Passive by reflection of light from opaque surfaces	Passive and active using electromagnetic, particulate and acoustic radiation
Image Formation	Refractive optical system	Various systems
Control of Irradiance	Muscle-controlled pupil	Motorized apertures, filter wheels, tunable filters
Focusing	Muscle-controlled change of focal length	Autofocus systems based on various principles of distance measurements
Irradiance Resolution	Logarithmic sensitivity	Linear sensitivity, quantization between 8 and 16 bits; logarithmic sensitivity
Tracking	Highly mobile eyeball	Scanner and robot-mounted cameras
Processing and Analysis	Hierarchically organized parallel processing	Serial processing still dominant

2.3 Digital Image Manipulation in Computer Vision

The digital image manipulation process refers to the consecutive procedures of image *pre-processing* and *processing* which take place sequentially after the image acquisition phase.

The pre-processing tasks are in charge of enhancing the image quality for the posterior processing stage which manages the execution of exploratory methods in order to extract qualitative/quantitative information of the scene under study. Figure 2.1 presents a graphic schematic that illustrates the processes described above.

As it is possible to observe, pre-processing tasks are performed to make the picture acquire an adequate quality Q . Notice that in the case that the Q value does not accomplish the reference value, pre-processing strategies will be executed recursively or a new acquisition will be requested. Once a correct quality value is accomplished, the respective processing tasks are executed applying features extraction methods.

In a general way, image manipulations methods of pre-processing and processing allow

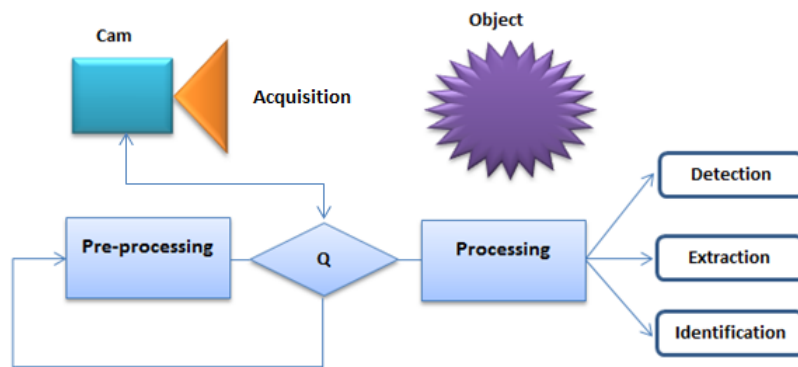


Figure 2.1: *Image Digital Acquisition, Pre-processing and Processing Procedures*

to obtain quality enhancements namely, smoothing, noise filtering, contrast improvement, region segmentation, resizing, feature extraction, image parametrization, surfaces detection, object identification, among others.

2.3.1 Image Pre-processing and Processing Methods

The image pre-processing is a procedure with a relevant added value during the subsequent features extraction and data-analysis processes in computer vision systems. Basically, the pre-processing objective consists in adjusting, improving or emphasizing the information present in a digital image or in a part of it, this with the purpose of improving the proper performance of the processing algorithms.

Pre-processing techniques such as histogram equalization and adjustment, filtering or morphological operations, significantly enhance the obtained results in the vision-based systems.

Correction strategies to be executed depend on the acquisition phase and the scene conditions. In some industrial production scenarios, it is possible to partially control the scene environment of the image acquisition. Using this resource, it is likely to facilitate the pre-processing tasks by means of the adjustment on parameters such us intensity, directionality, object reflexivity, surrounding space, heat, spectral distribution, sensor response and camera parameters.

This approach address image pre-processing methods for noise reduction and contrast

and sharpness enhancement. Correction strategies studied are based on histogram manipulation, spatial convolution filtering, frequency-domain algorithms, morphological operations and binary logic. The next sections present the implemented methods.

2.3.1.1 Histogram Manipulation: Equalization and Expansion

The histogram of a monochromatic digital image with L tonal intensity values is a function $h(x) = nx$, where nx is the number of times that the x value appears in the image. Commonly, the histograms are normalized by dividing the tonal frequency occurrence values between the total image pixels number, in this way the histogram represents an estimation of the occurrence probability of each value in the image.

Therefore, using the histogram information is likely to infer image features as brightness, contrast, affecting noise, sharpness, among others. The histogram manipulation by means of punctual transformations allows to enhance the image contrast and definition. One representative example of this kind of transformations is the histogram *equalization*, algorithm that consists in a proportional function to the accumulated histogram of the original image that pursues to uniform the values of the new histogram (see equation 2.3.1).

Let $p_r(r)$ and $p_s(s)$ be the histograms of the input and output images. Admit that r and s denote the normalized Grey-level values belonging to the interval $[0, 1]$. Then, the histogram equalization seeks a transformation of the type $s = T(r)$ such that $p_s(s) = r \cdot c$, with c constant. Thus, the processed (output) image is obtained by mapping each pixel with level r_k in the input image into a corresponding pixel with level s_k in the output image, as shown in equation 2.3.1 [18].

$$s_k = T(r_k) = \sum_{j=0}^k p_r(r_j) \quad 0 \leq r_k \leq 1 \quad (2.3.1)$$

The standard histogram equalization is highly used in image pre-processing with the aim of uniforming brightness scales. Alternative algorithms of this technique exist, for instance: the *adaptive*, *dynamic* and *duallistic sub-image* histogram equalization methods [17]. These algorithm variations offer an added value in terms of noise reduction and luminance shift.

Another method usually applied in the histogram manipulation is the *expansion*. This

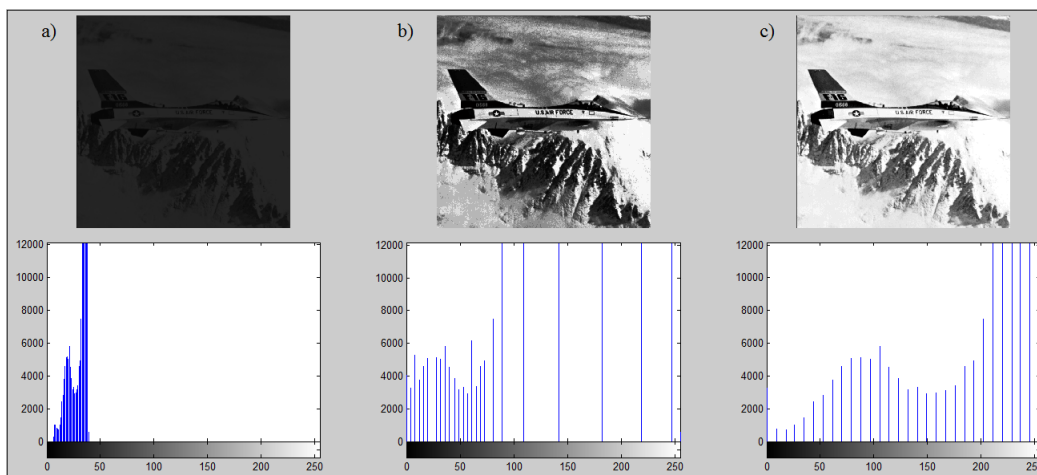


Figure 2.2: *Histogram Manipulation. a) Original b) Equalization c) Expansion.*

technique modifies the pixels values, so that the minimum and the maximum values will be defined limits, commonly the first and the last value of the used intensity scale. Equation 2.3.2 shows the corresponding mathematical operation of the expansion:

$$out\ Pixel = (in\ Pixel - in\ Min) \cdot \frac{(out\ Max - out\ Min)}{(in\ Max - in\ Min)} \cdot out\ Min \quad (2.3.2)$$

where $(out\ Pixel, in\ Pixel)$ are the corresponding new and original values of each pixel, and $(out\ Max, out\ Min) - (in\ Max, in\ Min)$ are the maximum and minimum values of the the new and the original grey-scales, respectively.

Figure 2.2 shows the results obtained by means of the execution of the histogram equalization and expansion methods.

2.3.1.2 Noise Reduction

The aim of the noise reduction methods is to suppress the affecting noise while preserving important object details in an image. Noise reduction is one of the most common techniques performed during the image pre-processing tasks, consequently a large number of approaches on the topic exist in the literature [41] [46] [48] [57].

The main noise source in digital images takes place during the acquisition phase and the subsequent data transmission process. The most common types of noise usually detected are

aleatory, gaussian, “salt and pepper” and multiplier [18].

A highly used technique in this field consists in calculating an average of several images acquired from the same scene. The larger is the number of images considered for the average, the better the obtained results will be.

Another algorithm widely used with noise reduction purposes comprises the application of filters in the spatial domain. These filters have the objective of improving the signal-to-noise ratio preserving the anatomical characteristics of the component objects by means of the convolution of the image with predefined masks or kernels. Some filters of this kind are: *uniform, average, median, mean, gaussian, Kuwahara, bilateral, alpha-trimmed, harmonic*, among others [19]. Their application permits to smooth the image, remove noise components and detect or enhance specific details.

2.3.1.3 Edges Enhancement

There are in turn a series of filters designed to enhance and stress the image edges, which are particularly important for the features extraction tasks during the processing stage. These filters operate based on the image derivatives (see Figure 2.3), considering characteristic numerical values shown through the image according to the type of region, i.e., constant zones, ramps, grey transitions, edges, among others. The areas with accentuated variations present derivatives of great magnitude, while the zones with soft transitions or uniform tendency are characterized by small or null magnitudes.

The performance of some tasks such as image segmentation, boundary detection, object recognition and so on, depend on the success of the edges characterization and edge preservations [42]. Some examples of edge enhancement filters that can be implemented for image sharpness improvement purposes are the Laplacian [7] and the Laplacian of Gaussian (LoG) [29].

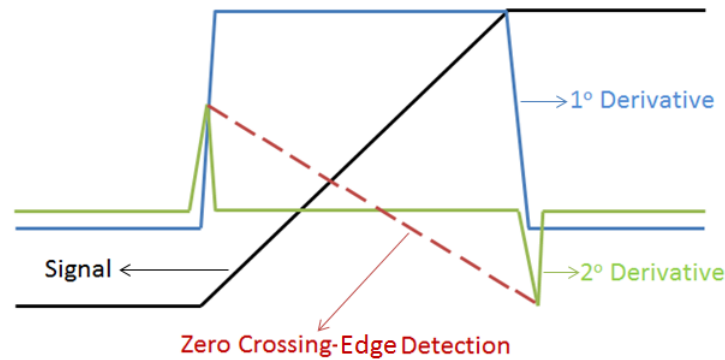


Figure 2.3: *Signal Derivatives Analysis for Zero Crossing and Edge Detection*

2.3.1.4 Unsharp Masking and High Boost Filtering

The unsharp masking method is a sharpening operator that emphasizes high frequency contents to enhance edges and other details in an image [52]. The technique consists in subtracting an unsharp or smoothed version of an image from the original image, as shown in Figure 2.4 and equation 2.3.3.

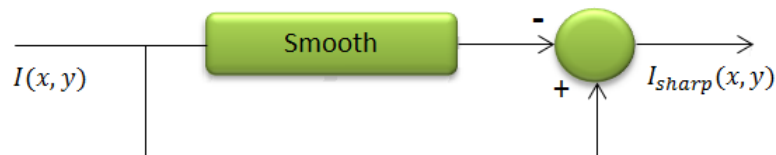


Figure 2.4: *Unsharp Masking Blocks Diagram*

The smoothing process is usually performed by means of the application of low-pass filters. The Gaussian filter is widely used in this kind of pre-processing tasks.

In the other hand, the high boost filtering corresponds to a generalization of the unsharp masking method. Assigning a positive weight constant K ($K > 1$) to the original image and performing some algebraic operations, as shown in the equations 2.3.3 and 2.3.4, the resulting mathematical expression is obtained in 2.3.5. By means of high boost filtering it is likely to enhance high frequency elements while still keeping desired low frequency components.

$$I_{HB}(x, y) = K \cdot I(x, y) - I_{Smoothed}(x, y) \quad (2.3.3)$$

$$I_{HB}(x, y) = (K - 1) \cdot I(x, y) + I(x, y) - I_{Smoothed}(x, y) \quad (2.3.4)$$

$$I_{HB}(x, y) = (K - 1) \cdot I(x, y) + I_{Sharp}(x, y) \quad (2.3.5)$$

2.3.1.5 Morphological Operations

There exist some methods based on binary logic that allow to preserve the essential image object features, removing noise, irrelevant details, or simply not important information. These techniques are denominated morphological operations and the fundamental operators are *erosion* and *dilation* [36]. Then, the combination of these operators originate two additional ones called *opening* and *closing* [6]. Defining A as an image and B as a structuring element, the corresponding mathematical expressions are presented in Table 2.2.

Table 2.2: *Morphological Operations*

Operation	Mathematical Expression
Dilation	$A \oplus B = \delta_B(A) = \{x \mid B_x \cap A\} \neq \emptyset$
Erosion	$A \ominus B = \Psi_B(A) = \{x \mid B_x \subseteq A\}$
Opening	$A \circ B = (A \ominus B) \oplus B$
Closing	$A \bullet B = (A \oplus B) \ominus B$

Figure 2.5 presents an example of the application of morphological operations in noise remotion tasks.

2.3.1.6 Contours Detection

Most edge and contour detection methods are based on intensity derivatives and work on the assumption that edges occur where there is a discontinuity in the intensity function or a very steep intensity gradient in the image. Derivative based approaches can be categorized

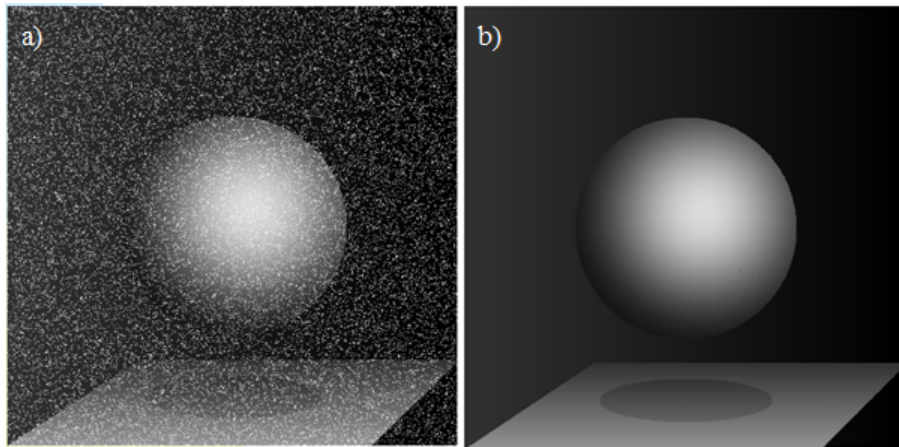


Figure 2.5: *Morphological Operations. a) Original Image b) Enhanced Image by Opening*

into two groups, namely first and second order derivatives. Commonly, the derivatives are estimated by means of mask convolutions [12].

Using derivatives estimation, it is likely to determine the magnitude and direction of the gradients through the image. Then, the maximum values computed correspond to the zones with higher probabilities of being edges and image contours.

There exists a extensive range of methods for edge detection that implement algorithms based on the assumptions stated above. Some examples of the most widely used in image processing are: Prewitt, Sobel [1], Marr-Hildreth [4] and Canny [40].

Figure 2.6 illustrates a typical application of image contour detection performed by means of the Sobel edge detection method.

2.3.1.7 Hough Transform

Hough Transform is a standard image analysis tool for finding curves that can be defined in a parametrical form such as lines and circles [5]. It is method for automatic detection of collinear points, highly used in the recognition of parametric surfaces in binary digital images.

Hough's method is based on the fact that certain curves can be completely described by few parameters. For instance, straight lines in a plane are fully determined by two parameters, i.e., in cartesian coordinates their gradient m and their y-axis offset b , or in polar coordinates

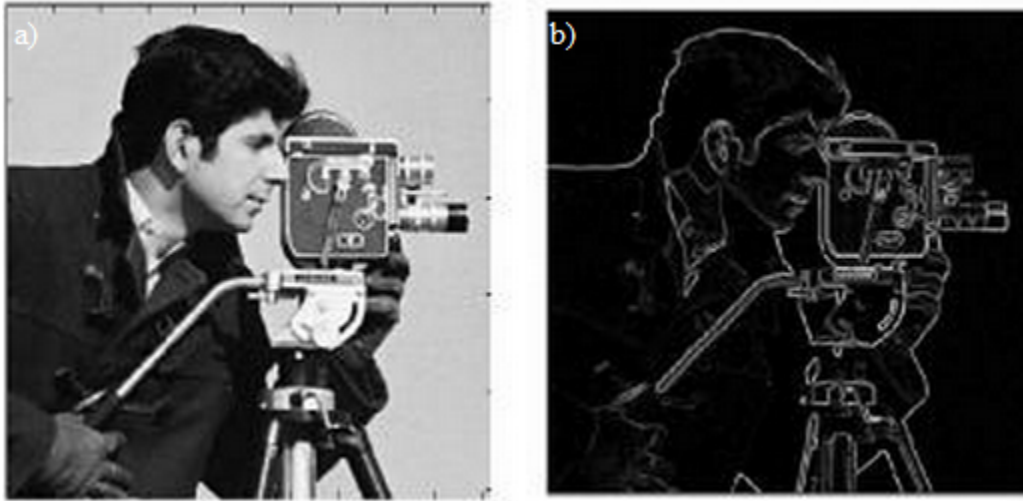


Figure 2.6: *Contour Detection. Sobel Method* a) Input Image. b) Processed Image [12]

their magnitude ρ and their angular orientation θ , as shown respectively in equations 2.3.6 and 2.3.7.

$$y = mx + b \quad (2.3.6)$$

$$\rho = x \sin(\theta) + y \cos(\theta) \quad (2.3.7)$$

The first step of the method is the formation of a two-dimensional parameters space, i.e. it is consider a matrix whose columns represent a finite number of different values for the first parameter and, whose rows represent the corresponding values for the second. Hence a straight line is allocated to each element of the matrix. Note that such a matrix can only represent a finite number of lines, suitable for the case of the digital images of finite size.

Then a counter is allocated to each point in the parameter scene, i.e., to each element of the parameters matrix (accumulator). The counters are initialized to zero before the start of the transformation.

The actual transformation step now consists of running through all the pixels of the input image and to increment those accumulator cells which represent straight lines going through the pixel which is currently being looked at.

The result of the transformation is a filled accumulator. All those accumulator cells whose

values are larger than one, represent a straight line which contains at least two pixels of the original image. The larger the cell value, the more pixels are contained in the line represented by the cell. Hence the aim of the analysis is to determine the maximum in the accumulator, as those represent the significant straight lines in the original image [9].

Hough's method logic can be extended to recognise other parametric surfaces in binary images, this is the case of ellipses.

2.3.1.8 Geometric Matching

Geometric matching (GM) refers to the methods applied for locating regions in digital images that match a determined template of a reference pattern based on geometries. GM algorithms are particularly suitable to locate objects that are characterized by distinct shape information.

The GM logic lies in searching for instances of the template in each inspection image and calculating a score for each match. The score value relates how closely the template resembles the located matches.

The calculation of the matching score takes into consideration factors such as: the number of geometric features in the template image that matched the target, the individual scores obtained from matching template features to their corresponding features in the inspection image and the score obtained by comparing the edge strengths of the curves in the template image to the edge strengths of the corresponding curves in the inspection image [22].

Geometric matching techniques are based on image processing methods such as: curve extraction, edge detection and the generalized Hough matching. Common applications of this kind of algorithms include gauging, inspection, alignment and sorting.

Currently, GM tools are capable of finding template matches regardless of lighting variation, blur, noise, occlusion, and geometric transformations such as shifting, rotation, or scaling of the template. Figure 2.7 illustrates an example of the GM principle.

2.3.1.9 Pattern Matching

Pattern matching (PM) refers to those methods designed for locating image instances that match a determined pattern template based on intensity properties. PM algorithms

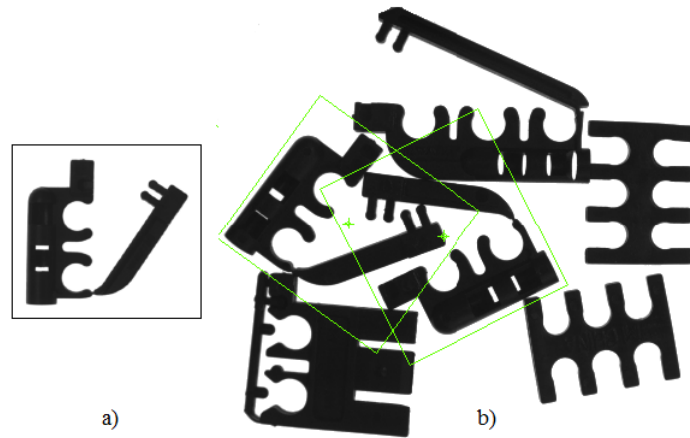


Figure 2.7: Geometric Matching a) Template b) Inspection Image

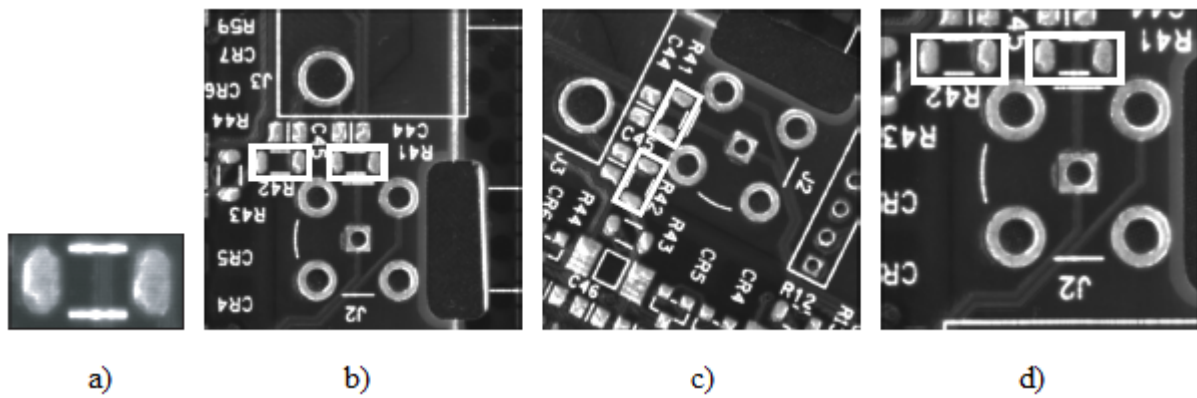


Figure 2.8: Pattern Matching (PM) a) Template b) Inspection Image c) Rotation d) Scaling

use the pixel intensity information present in the template image as the primary feature for matching. As the GM algorithms, PM calculates matching scores in order to quantify the accuracy of the match.

Generally, the techniques used in PM procedures involve methods such as curve normalized cross-correlation extraction, scale-rotation invariant matching and pyramidal matching [22]. Figure 2.8 presents a graphic example of the a recognition procedure performed by PM.

2.4 Neural Networks

2.4.1 Concept and Brief History

Neural Networks (NNs), also known as Artificial Neural Networks (ANNs), are massively parallel interconnected networks of simple (usually adaptive) elements and their hierarchical organizations which are intended to interact with the objects of the real world in the same way as biological nervous systems do [24].

The ANNs are an attempt at modeling the information processing capabilities of nervous systems and perform cognitive tasks, such as learning and optimization [43][34]. In fact, they present a processing model similar to the brain, i.e., knowledge is acquire experientially by a learning process, stored by means of connections (synaptic weights) [20] and may lead to solve complex information processing problems that would be difficult to achieve using traditional symbolic approaches.

The earliest work on neural networks dates back to 1943 when McCulloch and Pitt presented their M-P model of an artificial neuron, affirming that a network composed by multiple neurons might determine any computable function.

During the 1950s and 1960s novel approaches on single-layer neural networks were taken; among them highlights the development of the LMS algorithm [16] and the formulation of the Adaptive Linear Element (ADALINE) [58] by Widrow and Hoff.

Research on the field continued with successful results on pattern recognition and classification until 1969, when Minsky and Peper demonstrated the incapability of the perceptron to solve simple functions such as the logical XOR. Such limitations led to the decline of research on neural networks during the 1970s.

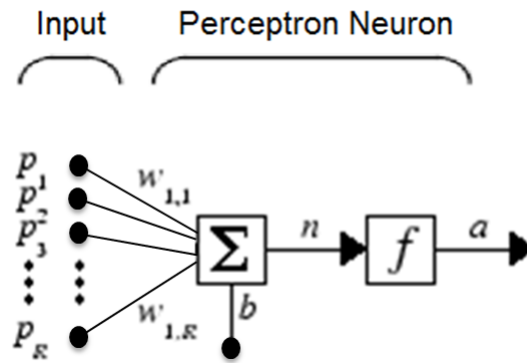


Figure 2.9: Neuron Model Diagram

In the early 1980s, research initiatives resurged largely due to successful contributions made by Hopfield on learning algorithms for multilayer neural networks.

Currently, diverse neural networks architectures exist, for instance: multilayer feedforward networks, radial basis function networks, adaptive resonance theory models, self-organizing feature mapping networks and others [56].

2.4.2 Perceptron Neuron and Multilayer Neural Networks

A perceptron neuron is a nonlinear, parametrized, bounded function with variables termed inputs and outputs [13].

The output of the neuron is the result of a series of distinct functional operations. First, a number R of inputs $\{p_i\}$ are multiplied by scalar weights $\{w_i\}$. Second, the obtained product is added to an additional constant bias b to form the net input n . Finally, the net input is passed through the transfer function f which produces the output a , as shown in equation 2.4.1 and illustrated in Figure 2.9.

$$a = f(n) = f\left(b + \sum_{i=1}^R w_i \cdot p_i\right) \quad (2.4.1)$$

Note that w_i and b are both adjustable scalar parameters of the neuron. The central idea of neural networks is that such parameters can be adjusted so that the network exhibits some desired or interesting behavior. Thus, it is likely to train the network to do a particular job

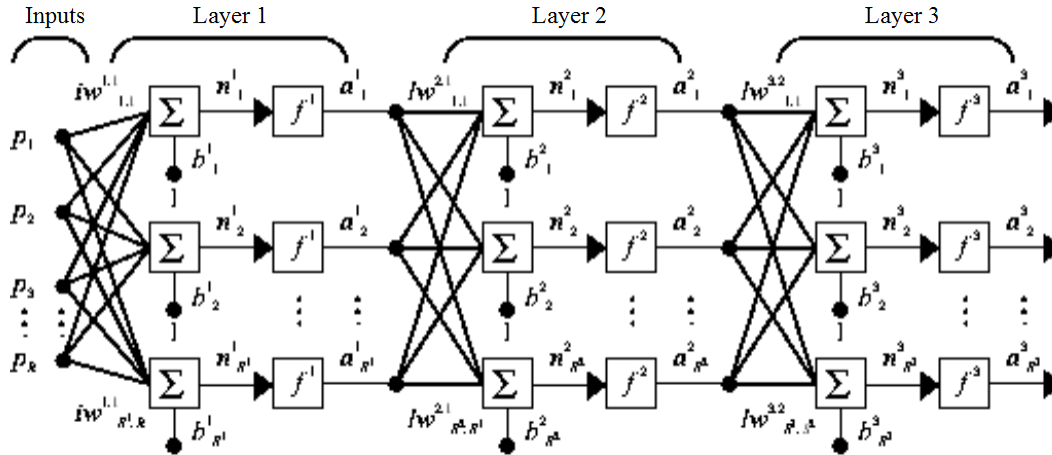


Figure 2.10: Multilayer Neural Network Diagram

by adjusting the weight or bias parameters [21].

The neuron model concept can be extended into more complex structures of interconnected neurons grouped into layers, i.e., *multilayer neural networks*, also known as *multilayer perceptrons* (MLP) [45]. In this architecture, the layer that produces the network output is called *output layer*, while the rest are denominated *hidden layers*. For instance, the three-layer network shown in Figure 2.10 has one output layer (layer 3) and two hidden layers (layers 1 and 2).

Several transfer functions (TFs) can be implemented for the neural network layers activation, being the most common in the literature the *linear*, *tan-sigmoid* and *log-sigmoid* types [31].

Certainly, the properties of the functions make each one specially adequate for specific purposes; for instance, the linear TF is widely used in the final layer of multilayer networks that are used as function approximators, while the sigmoid TFs are commonly used in the hidden layers, mainly because their differentiable properties.

2.4.3 Neural Network Learning Process

The neural network learning stage is a process by which the network's parameters (i.e., synaptic weights and bias levels) are adapted through a continuous process of stimulation

relative to the environment in which the network is embedded.

The type of learning is determined by the method in which the parameter changes take place, i.e., by supervised or unsupervised methods [45].

2.4.3.1 Supervised Learning

Supervised learning methods assume the availability of a labeled set of training data constituted by N input–output examples pairs as shown in equation 2.4.2:

$$T = \{x_i, d_i\}_{i=1}^N \quad (2.4.2)$$

where x_i is the inputs vector, d_i is the desired target vector and N the sample size.

Given the training sample T , the requirement is to compute the parameters of the neural network, so that the actual output y_i of the neural network due to x_i is close enough to d_i for all i in a statistical sense. For instance, the minimization of mean-square error is widely used as an the index of performance (see equation 2.4.3).

$$E(n) = \frac{1}{N} \sum_{i=1}^N (d_i - y_i)^2 \quad (2.4.3)$$

The Supervised learning is widely implemented in feed-forward neural networks by means of back-propagation algorithms, and in radial-basis function networks.

2.4.3.2 Unsupervised Learning

Turning next to unsupervised learning, the adjustment of the network's parameters is carried through the use of neurobiological principles such as Hebbian learning and competitive learning.

According to Hebb's postulates, the change in synaptic weight Δw_{ji} of a neural network can be defined by equation 2.4.4:

$$\Delta w_{ji} = \eta x_i y_j \quad (2.4.4)$$

where η is the learning-rate parameter, x_i is the input (presynaptic) signal and y_j the output

(postsynaptic) signal.

By contrast with supervised learning, there are no explicit target outputs or environmental evaluations associated with each input; rather the unsupervised learner brings to bear prior biases as to what aspects of the structure of the input should be captured in the output [49].

Unsupervised learning methods are used for applications such as Self-Organizing Maps (SOM), clustering and Principal Components Analysis (PCA).

2.4.4 Back Propagation Feed-Forward Networks

The *Back Propagation (BP)* algorithm is by far the most widely used method in multi-layer feedforward architectures [51]. The basic idea of the algorithm is that the learning process is composed of positive communication of the signal and back propagation of the error. In the following sections the BP phases, learning modes and enhancement heuristics are discussed.

2.4.4.1 Phases

The training process by using the back-propagation algorithm is accomplished in two phases:

- *Forward Phase.* During this phase the network's parameters are fixed and the input signal is propagated through the network layer by layer. The forward phase finishes with the computation of the error signal given in equation 2.4.5.

$$e_i = d_i - y_i \tag{2.4.5}$$

- *Backward Phase.* During the second phase, the error signal e_i is propagated through the network in the backward direction, hence the name of the algorithm. It is during this phase that adjustments are applied to the network's parameters so as to minimize the error e_i in a statistical sense.

2.4.4.2 Learning Modes

The BP learning process may be implemented in two modes, namely, sequential mode or batch mode.

1. *Sequential mode*: In this mode of BP learning, adjustments are made to the network's parameters on an example-by-example basis, i.e., the weights and biases are updated after each input has been presented. The sequential mode is best suited for pattern classification.

2. *Batch mode*: In this second mode of BP learning, adjustments are made to the network's parameters on an epoch- by-epoch basis, where each epoch consists of the entire set of training examples. Stated in other words, the weights and biases are only updated after all the inputs and targets have been presented. The batch mode is best suited for nonlinear regression.

2.4.4.3 Enhancement Heuristics

The BP algorithm is simple to implement and computationally efficient in that its complexity is linear in the synaptic weights of the network. However, a major limitation of the algorithm is that it does not always converge and can be slow, particularly when it is necessary to deal with a difficult learning task that requires the use of a large network.

Then, in order to improve the performance of the algorithm it is likely to invoke the following list of heuristics:

- Use neurons with antisymmetric activation functions in preference to nonsymmetric activation functions.
- Shuffle the training examples after the presentation of each epoch; an epoch involves the presentation of the entire set of training examples to the network.
- Follow an easy-to-learn example with a difficult one.
- Pre-process the input data so as to remove the mean and decorrelate the data.
- Incorporate prior information into the network design whenever it is available.
- Split the set of training examples into two parts: an *estimation* subset used for training of the model and *validation* subset used for evaluating the model performance.

2.4.5 Neural Network Design Work Flow

The work flow for the neural network design process has seven primary steps [21], namely:

- Collect data
- Create the network
- Configure the network
- Initialize the weights and biases
- Train the network
- Validate the network
- Use the network

After a neural network has been created, it needs to be configured and then trained. Configuration involves arranging the network, so that it is compatible with the studied problem. Once the network has been configured, the adjustable network parameters need to be tuned, so that the network performance is optimized. This tuning process is referred as training the network. Configuration and training require that the network be provided with example data.

2.5 Vision-based Industrial Applications

Modern computer vision architectures based on advanced image pre-processing and processing methods, constitute a emergent technological resource boosted by an important portion of control systems in industrial applications such as: robotics, surveillance systems, processes automation, tele-guided mechanisms, supervision and quality control, among others.

Nowadays, it is likely to find companies dedicated to the development and implementation of vision-based inspection systems; this is the case of *MAPS Informática Industrial*¹, company who put on the market the application *VISIOMAPS*®.

¹<http://www.maps.es>

VISIOMAPS® is a system capable of processing images in real-time and executing multiple configurable functions, e.g., packing and labeling supervision, prints control and measures pieces inspection. The system presents a flexible design conceived to work under different industrial environments and operates in a PC-embedded architecture with an specialized VLSI processor.

Between the main system image processing specifications, the architecture offers: 256 levels grey-scale, 336 x 286 image resolution, BMP image format, color detection and customizable definition of coordinates windows in the image.

As it is possible to observe, this industrial system presents configurable and flexible features, however the architecture does not address self-adapting and self-optimizing behaviours, fact that limits its functionality in dynamic industrial systems. Moreover, the image resolution used in the system is low, compromising the accuracy of high-precision tasks.

In the same line of industrial product inspection, [28] exposes the implementation of an automatic vision-based assortment system for corks classification. The system is capable of analysing and distinguishing defects present on the surface of corks (e.g.:cracks, random-shaped holes and others) by means of processing algorithms such as morphological operations, spatial convolution filtering, histogram manipulation, image binarisation and edge detection. Structural features and properties such as size, form and position of the defects are addressed in the approach with the purpose of standardize the corks classification.

The approach combines hardware and software components in a modular architecture divided into acquisition and processing modules. The system performs the image acquisitions by means of a linear CCD camera, transmits data over an Ethernet based network and finally, processes the image to extract quantitative information using a PC system.

Another project of interest in the industrial inspection field is presented in [10]. This project describes the development of a CV system designed for the supervision of the bottling and packing processes in a beer industry. Using processing algorithms such as segmentation, filtering and contour detection; the system is capable of verifying if the bottles are correctly placed in boxes according to the color of their tops, checking if the boxes are fully populated and detecting if elements with superficial defects exist. The respective system architecture was developed in C++, using the library of image processing and analysis Open CV. The final version of the system was installed in a real production factory with satisfactory results.

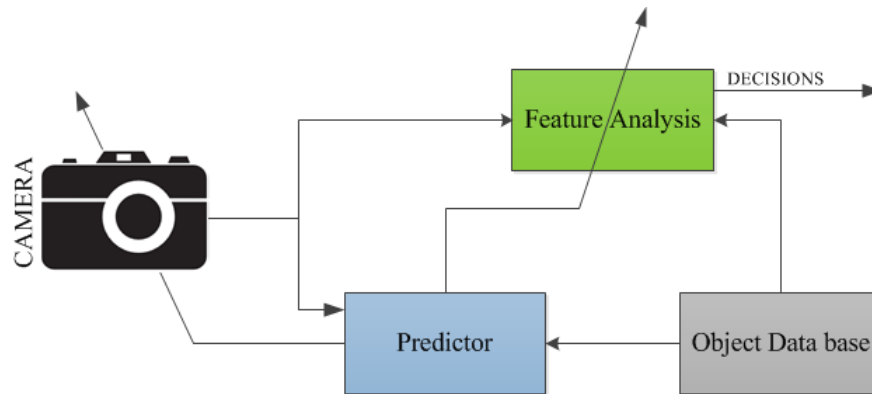


Figure 2.11: *Predictive Control Functional Diagram. Adapted from [38]*

There exist some other real-time vision systems based on predictive control methods. In [38], it is addressed the implementation of a system with sequential data base analysis and predictive control strategies. By means of a closed control loop, the system is capable of controlling in advanced the image acquisition parameters of the camera. This architecture scheme presents an important particularity i.g., control decisions strategies based on the previous system information stored in the data base, allow the system to self-adapt to the environment. In the other hand, the detection results presented in this project do not accomplish the accuracy requirements of a rigorous control system. It is fundamental to significantly improve the image pre-processing and processing methods. Figure 2.11 presents the schematic of the describe architecture.

In [35], it is presented an novel approach that compiles a series of single numerical indexes related to image properties such as contrast, luminosity and sharpness, as well as a global quality function. By means of the determination and optimization of these values, the control system performs different strategies in order to regulate the acquisition parameters of the camera, according to the surrounding conditions. The quality indexes and the global quality function stated in the approach, are adapted in the present work in order to be used in the evaluation of the pre-processing strategies that will be implemented. More details of the methods presented in [35] are discussed in Section 3.3.1.

Regarding to the topic of image enhancement methods, [14] develops an adaptive image enhancement system for variable illumination conditions optimized for face recogni-

tion applications. The approach exposes the implementation of adaptive region-based image pre-processing methods for contrast and edges enhancement. The image enhancement scheme proposed is based on novel manipulation techniques such as: regional histogram equalization applied on low-frequency coefficients to minimize illumination variations under different lightning conditions, and enlargement of the high-frequency coefficients for highlighting the image edges.

Then, in the field of image recognition systems based on neural networks, [25] presents the development of a work on medical image recognition by using GMDH-type neural networks. The approach is based on the automatic organization of neural networks architectures performed by the heuristic self-organization method. The recognition neural network can identify components (in this case the structure of the lungs in an X-ray CT) using as input variables some statistics parameters such as: mean, standard deviation, variance, minimum and maximum levels and dynamic range. The obtained results were accurate and constitute one useful method for image medical recognition.

Also, it is possible to find in the literature approaches that combine AV systems with NN architectures in industrial recognition tasks. For instance, [55] addresses the application of CV and NN algorithms in touching characters recognition of industry products. The work presents a method which consists in rotating the character image to the horizontal position, using morphological operations to cut the touching characters and finally, identifying the resulting characters by means of a back-propagation neural network.

After executing the character identification labour, the system updates the neural network training set, by including the recent identified characters. This action allows the architecture to consider the previous system's history for next identifications.

In the same thematic of NNs and processing methods for image identification, [47] develops a AV identification system based on the segmented Fourier transform (which gives the image a rotational and translational invariant structure) and HAVNET neural networks [44]. This kind of NNs are specially designed for two-dimensional binary pattern recognition. The approach addresses the identification of grey-scale fingerprints using image processing techniques such as: edge enhancement, histogram equalization, Laplacian filtering and thresholding.

2.6 Summary

Along this chapter, fundamental concepts about computer vision, image manipulation and neural networks, as well as the state of the art on computer vision-based systems in the industry were addressed.

As it was pointed out in section 2.2, computer vision systems are artificial approaches inspired in the biological systems of vision. By means of electronic devices and software algorithms, the CV pretends to simulate the main biological vision features in order to endow artificial entities, such as robots and guided mechanisms, with vision capabilities.

Turning to the image manipulation field, section 2.3 exposes an extended series of quality enhancement methods widely used in image pre-processing and processing procedures. For instance, image manipulation techniques such as histogram manipulation, noise reduction, edges enhancement, unsharp masking and morphological operations are examined and explained.

Then, section 2.4 introduces the neural networks field presenting the fundamentals relative to the architecture, learning methods, training algorithms and implementation methodology of multi-layer neural networks in recognition applications.

Finally, section 2.5 presents a review of the state of the art on projects that dynamically combine the previous fields and present novel approaches in the development of industrial quality control applications.

Chapter 3

System Architecture

3.1 Introduction

A changing manufacturing environment characterized by the aggressive competition on a global scale and rapid changes in process technology requires the development of up-gradable system paradigms capable of integrating new functionalities and reconfiguring according to the new processes needs [15].

Consequently, in the context of modern industrial manufacturing, modularity and flexibility are system key-features in order to accomplish the current requirements of the production companies and achieve cost-effective and rapid changes responses within the diverse manufacturing processes.

3.2 Task Organisation Scheme

The pre-processing system operation takes place just after the image acquisition process and before the image processing stage, as illustrated in Figures 3.1 and 3.2. The proposed system architecture for the pre-processing comprises two main modules, namely, a scene recognition module and an adaptive quality enhancement module.

The scene recognition module is in charge of determining which type of picture scenario is being processed based on the evaluation of the incoming image indexes. Meanwhile, the

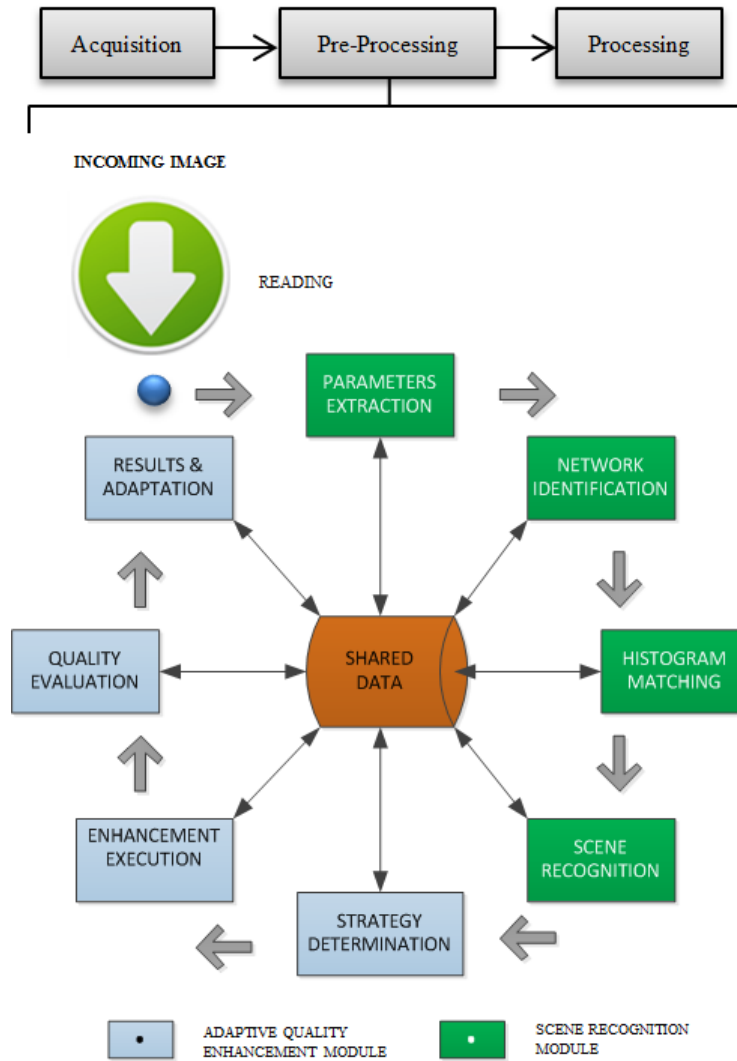


Figure 3.1: Image Pre-Processing Task Scheme

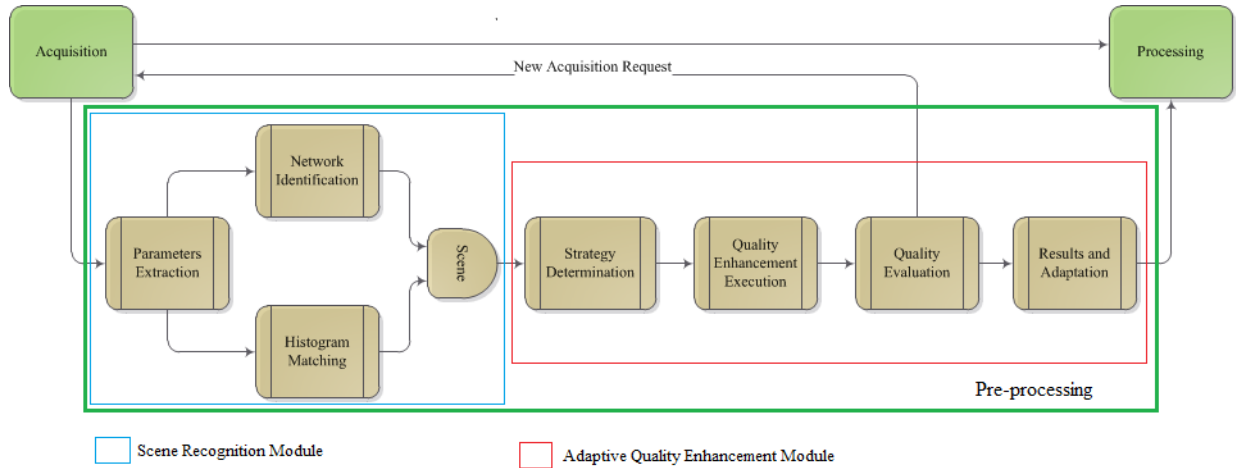


Figure 3.2: Image Pre-processing Architecture

adaptive quality enhancement module manages the dynamic choice and execution of the optimal correction strategies for the current image in progress.

The corresponding required tasks for the image analysis and manipulation are performed by the sequential execution of different routines running on a shared-data logical framework, i.e., the program variables and structures are successively used, shared and modified by the routines calls.

A strategically task organization scheme is proposed in order to accomplish the image quality correction. In sequential routines order, the process is (see Figure 3.1): *incoming image reading, parameters extraction, network identification, histogram matching, scene recognition, strategy determination, enhancement execution, quality evaluation and adaptation, and results display*. The routines execution works as follows: first four tasks are performed by the scene recognition module and the remaining five ones are executed by the adaptive quality enhancement module.

The proposed architecture is designed to execute sequential processing, i.e., while program routines are running a no-interrupt flag is set to indicate that the program is currently busy and cannot accept a new incoming picture.

The modular design allows the system to be easily implemented and interconnected with an extended range of tools running on different platforms, e.g., C++, MATLAB[®], LabVIEW[®] and JAVA. Diverse scripting methods between these programming languages cur-

rently exist, allowing the execution of multiple routines calls and interconnections. Among others, interconnected tools could be an image acquisition unit, a feature extraction processing module and or a control multi-agent system (MAS) [11].

3.3 The Scene Recognition Module

Along this section, the scene recognition module development is presented. In first instance, a series of indexes and evaluation functions are defined and discussed. Then, the recognition task is explained based on the execution of a specific learning process powered by neural networks. Finally some examples of the advantages achieved by means of the application of scene recognition techniques in industrial systems are presented.

3.3.1 Parameters and Global Quality Functions

In order to determine the relative quality of an acquired image, it is necessary to define an evaluation criteria based on numerical computations. For this purpose, an adequate set of parameters that allows the quantification of properties such as contrast, luminosity and sharpness, must be defined. In addition, the establishment of an appropriate quality function that relates the individual indicators, assigning the proper significance to each one, is fundamental for this task.

The application of the functions set allows the system to parametrise specific scenes and obtain information related to the image properties. Meanwhile, the global quality function combines the contributions of the single parameters to penalize those values far from the desirable ones. Then, the strategy consists in minimizing this function and obtaining a single numerical value that represents a normalized image quality. A brief summary of a set of quality functions and parameters adapted from the approach developed in [35] is presented below.

Starting with a measure of the image sharpness and definition, it is addressed the *Tenengrad* parameter (TN). The TN's logic lies in the principle that focused pictures present defined contours characterized by high intensity variations. Therefore, its numerical calculation is based on the maximization of the gradient scheme, i.e., determining the gradient at each

image point and summing up the square values of the magnitudes. High TN values are related with well-defined images and hence it is intended to increase this parameter for quality enhancement purposes. Defining W and H as the dimensions of the image, the corresponding TN expression is given by the equation 3.3.1.

$$TN = \sum_{x=1}^W \sum_{y=1}^H G(x, y)^2 \quad \forall G(x, y) > Th \quad (3.3.1)$$

The gradient magnitude is obtained by the convolution of the Sobel operators (i_x , i_y) [1] with the image pixels $I(x, y)$ in the respective x and y directions, like it is showed in the equation 3.3.2.

$$G(x, y) = \sqrt{i_x * I(x, y)^2 + i_y * I(x, y)^2} \quad (3.3.2)$$

For noise impact reduction purposes, a numerical threshold (Th) is defined and only the magnitudes greater than this value are considered in the computation of TN. In this case, the threshold estimation is performed using the *Otsu's method* [39].

Alternative adaptive strategies for the TN calculation were tested, e.g., the adaptive Tenengrad estimation through weight allocation scheme [53], however longer execution times were obtained and consequently they were not implemented in the final system version.

In the same way, it is possible to define a *Flatness* (FL) parameter which provides information about those zones characterized by constant intensity tendency. Basically, FL counts those gradient pixels considered flat, i.e., their magnitudes are smaller than a certain suitable threshold Th_1 , see equations 3.3.3 and 3.3.4. The FL value reduction is desirable in order to improve the image definition.

$$FL = \sum_{x=1}^W \sum_{y=1}^H px \cdot (G(x, y)) \quad (3.3.3)$$

$$px \cdot (G(x, y)) = \begin{cases} 1 & \text{if } G(x, y) < Th_1 \\ 0 & \text{otherwise} \end{cases} \quad (3.3.4)$$

Generally, the flatness threshold computation is associated with the Tenengrad threshold

in a proportional relation of the type $Th_1 = k.Th$, and is selected conveniently based on the image noise level.

Other simple parameters can be included in order to obtain measures of the image brightness. For instance, it is likely to determine the higher and lower intensity magnitudes in the picture and their corresponding numerical difference, namely *low level* (LL) and *high level* (HL) and *dynamic range* (DR). It could also result important to know the number of pixels that present those saturated intensities, in this case defined respectively as *low saturation* (LS) and *high saturation* (HS) indexes. In this case, a wide dynamic range as well as a reduced number of saturated pixels is desirable.

Moreover, an *average luminosity* (LM) value can be computed as showed in the following equation:

$$LM = \frac{1}{T} \sum_{i=0}^{L-1} l_i \cdot O_i \quad (3.3.5)$$

where O_i represents the number of pixels with l_i intensity level, T the total considered pixels and L the number of tonal values (e.g., 256). The LM optimal value corresponds to the mean value of the grey level scale used (in this case 128).

Analogously, the statistical *variance* (V) of the grey level distribution is calculated according to the equation 3.3.6. This index gives a measure of how the distribution spreads from the mean luminosity value.

$$V = \frac{1}{T} \sum_{x=1}^W \sum_{y=1}^H [I(x, y) - LM]^2 \quad (3.3.6)$$

In addition, one last parameter based on the image histogram is included. This corresponds to the *entropy* value (E), which deals with the histogram and the intensity distribution uniformity. Defining $P(i)$ as the frequency of occurrences of the i level, normalized with respect to the total number of pixels, the E computation is defined as shown in the equation 3.3.7.

$$E = - \sum_{i=0}^{L-1} P(i) \ln(P(i)) \quad P(i) \neq 0 \quad (3.3.7)$$

Finally, all the previous indexes are combined in one quality penalization function Q . By this method, it is possible to assign different weights $\vec{w} = [w_1, w_2, \dots, w_{10}]$ to the single parameters values and therefore penalize those no desirable variables tendencies spreading from the optimal values. By means of this definition, it is attainable to estimate the normalized relative quality of the images processed. The corresponding Q expression is presented in the equation 3.3.8.

$$Q = w_1 \cdot TN^{-2} + w_2 \cdot FL^2 + w_3 \cdot E^2 + w_4 \cdot LS^2 + w_5 \cdot HS^2 + w_6 \cdot LL^2 \cdot LS + w_7 \cdot (255 - HL)^2 + w_8 \cdot [(255 - (HL - LL))]^2 + w_9 \cdot LM^2 + w_{10} \cdot (1 - V)^2 \quad (3.3.8)$$

The Q function tuning-up process requires an extensive set of experimental tests. In other words, the definition of the weights values is not based on direct criteria but on practical experiences. Using trial results, it is likely to observe certain index tendencies and get an approach about their global importance in the image quality. Certainly, some parameters can be more important than others, therefore their respective weights must be greater. As exposed in section 3.3.1.2, a constraint optimization method based on local minimums is included in the system for determining the weight values.

An interesting experiment consists in observing the effect of the luminosity level variation on the parameters behaviour. Partial experimental results of this test¹ are presented in Figure 3.3.

A quick graphic analysis of the data allows pointing out some relevant conclusions, for instance, that the TN and LM indexes are more significant than the E parameter; hence their respective weights must reflect this preponderance relation in the quality function.

3.3.1.1 Penalization Criteria

By means of the Q function (equation 3.3.8) is likely to penalize those behaviours in the parametric indexes that somehow affect the image quality. A detailed list of penalized tendencies is presented below.

¹Data gathered within the EU FP7 GRACE project (<http://www.grace-project.org>).

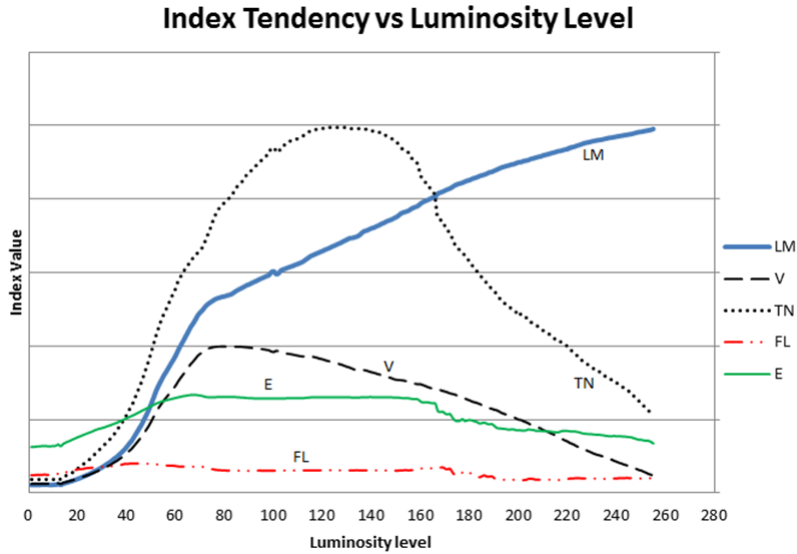


Figure 3.3: *Experimental Indexes Behaviour Test*

- Low Tenengrad values.
- Large amounts of flat pixels.
- High histogram entropy values.
- Large amounts of saturated pixels, i.e., pixels with the highest or lowest grey scale values.
- Dynamic ranges far from the maximum scale excursion (HL-LL).
- Luminosity distribution averages² far from the interval center of the used grey scale.
- Distribution variances far from the optimal value 1.
- Highest and lowest pixels values in the image different to the maximum and minimum values of the grey scale.

²In this approach the luminosity distribution average is normalized in the interval [-1,1] to be evaluated in the Q function.

Table 3.1: Boundaries of the Weight Constraint Optimization

Lower Boundary	Weight	Upper Boundary
0.5	w_1	0.9
0.3	w_2	0.7
0.1	w_3	0.2
0.2	w_4	0.3
0.2	w_5	0.3
0.2	w_6	0.3
0.2	w_7	0.4
0.4	w_8	0.6
0.3	w_9	0.4
0.1	w_{10}	0.2

3.3.1.2 Weights Vector Optimization

The weights determination task for the image quality calculation is a problem bounded by constraints. Basically, according to the preponderance of the single parametric indexes and the required scale normalization for each one of them, a set of constrained ranges must be defined. Then, by means of optimization methods based on iterative local minimization, it is likely to determine the ideal values for the respective quality function weights $\vec{w} = [w_1, w_2, \dots, w_{10}]$.

In this approach, it is intended to perform the weights optimization labour by using the function *fmincon* [32] through the *interior-point method* [8]. By using results gathered within a set of experimental tests, the respective constraints boundaries are stated as shown in table 3.1.

3.3.2 System Learning

Learning is a capability of entities with a certain level of intelligence. Nowadays, advances in computer science and engineering allow the implementation of artificial intelligence behaviours [50] and self-learning systems. Usually, these are based on neural networks and genetic algorithms. In fact, the recognition module developed in this work, operates by means of a feed-forward backpropagation neural network like the one showed in Figure 3.4.

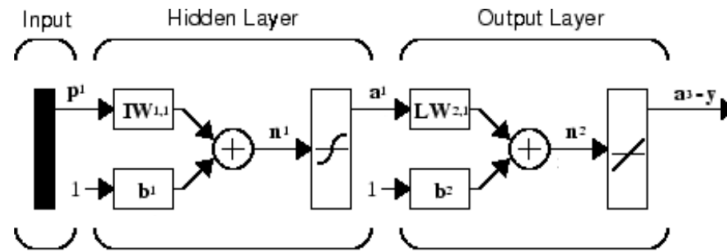


Figure 3.4: Feed-Forward Backpropagation Neural Network Schematic. Adapted from [21]

A neural network is a system with inputs and outputs related by non-fixed mathematical criteria. This kind of systems can be used to implement controllers with weights or internal parameters determined by a self-learning process. A multi-layer neural network can learn how to identify dynamic characteristics [37] and perform pattern identification with low error margins.

In order to teach a neural network how to respond to determined situations, it is necessary to provide specific examples of the possible occurrence scenarios. Stated in another way, initial reference domains composed by the typical system inputs and outputs (targets) must be defined.

In the image pre-processing system case, a set point routine is executed during the set-up phase of the program. Along this routine, some high quality pictures corresponding to the different likely scenarios are provided. A parameters extraction process is performed and a set of the resulting image indexes constitute the network's inputs (see subsection 3.3.3). Note that the image pixels themselves are not considered directly as inputs, but the calculated image parameters. Then, the respective scenario's number represent the corresponding system's targets (i.e., single numerical values: 1,2,3...n). Figure 3.5 presents a brief graphic illustration of the network's inputs, outputs and operation.

The designed network is implemented using a feed-forward multi-layer architecture consisting of an input layer, a single hidden layer and an output layer, as shown in the topology of Figure 3.4. The size of the hidden layer is determined by an iterative process until obtaining a desirable robust performance. For the case study, 11 nodes (neurons) was considered as an appropriated size. The training process is based on the backpropagation algorithm and the *trainlm* function [21]. Log-Sigmoid and Linear transfer functions are used respectively

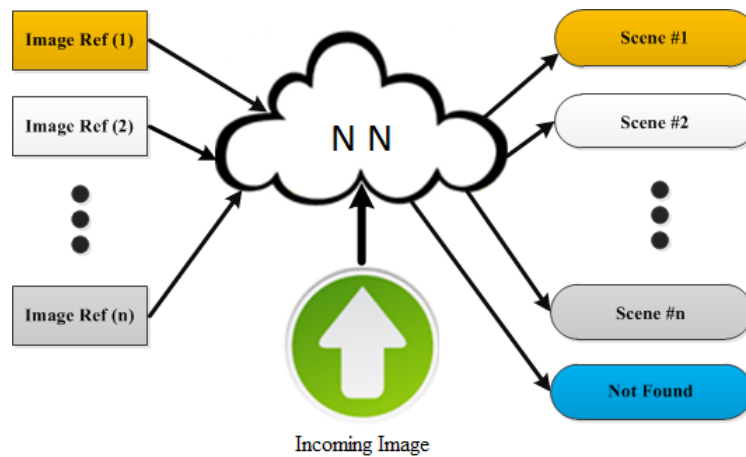


Figure 3.5: *Simplified Schematic of the Recognition Process*

like activations for the hidden and output layers.

Once the network is trained, the system is able to receive, recognize and associate pictures with a respective scene. Notice that the incoming images could present indexes that may differ from the ones given in the set point routine. Here is when the neural network intelligence is required in order to identify the coincidences and determine by a matching procedure which scene number better describes the image properties. The algorithm is designed for accepting as effective matches those with a coincidence probability of at least 80%. In case that the incoming image does not belong to any of the pre-defined scenarios, the system is capable of informing that no match was found (see Figure 3.5).

3.3.3 Complex Scene Identification

The indexes' set defined in Section 3.3.1, brings detailed information about the image parametric composition. Nevertheless, it can occur that images belonging to different scenes present similar parametric values. For instance, if the Grey-level scales in different image scenarios show homogeneous components, several individual parameters such as HL, LL, HS, LS and LM might be similar between the images. Clearly, this issue constitutes a problem for the scene recognition module performance, because close parametric similarities may lead to False-Positive or True-Negative results in the neural network matching procedure.

To solve this problem, linear combinations between the different indexes are explored to define the neural network's input vector. The resulting input must contain singular information about the contrast, sharpness and luminosity of the scene. The contributions of the single parameters in the vector are tuned up by means of constant weights k_i , reinforcing those parameters which allow the differentiations between the image scenarios. After an extensive set of experimental tests, the selected input vector used in this approach is defined is shown in equation 3.3.9. Note that the five components of the \overrightarrow{NN} vector correspond to the size of the neural network's input layer (i.e., 5).

$$\overrightarrow{NN} = (k_1 \cdot TN, k_2 \cdot (HL - LL), k_3 \cdot (HS - LS), k_4 \cdot E, k_5 \cdot V) \quad (3.3.9)$$

Moreover, an additional method is proposed in order to improve the accuracy of the scene recognition module in the case of facing complex matching situations. The technique addressed is based on the comparison of the histogram shape patterns, i.e., the incoming image's histogram is compared point-by-point with histograms of the reference images with the aim of detecting pattern coincidences.

The resulting partial matches are quantified by calculating a matching score, using the mean-square error (see equation 2.4.3) between the images' histograms contours. Then, defining a suitable threshold, the algorithm may consider the matching score value like an effective or a false match, this depending of the desired matching precision.

The gathered information is used for the system to reinforced the neural network decisions when tight-matching cases are presented. Figure 3.6 shows a graphic representation of the histogram matching method.

3.3.4 Advantages of the Scene Recognition Implementation

The recognition module allows the system to implement customized image procedures. For instance, having a previous knowledge about the possible scenarios permits to define areas of interest in the image and just manipulate the corresponding pixels. This fact has an important value in terms of execution time and system's performance.

Besides, scene recognition can compensate the negative influence of environmental factors on the image acquisition phase. For example, consider the case of an industrial company

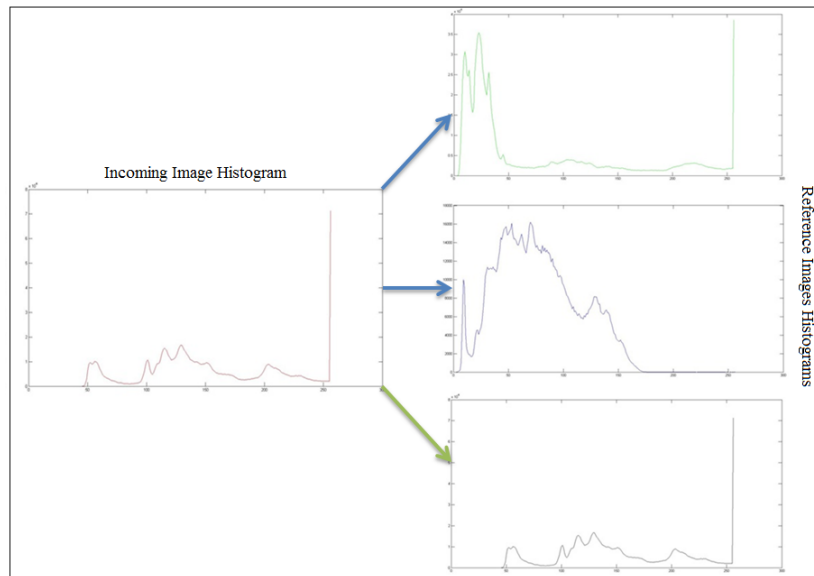


Figure 3.6: *Image Histogram Matching Method*

with several automated production lines working continuously during the day and the night. Naturally, the illumination conditions variations through the course of the day will have impact on the image quality of the acquisitions. This circumstance might affect the efficiency of the recognition algorithms. Then, by implementing a scene recognition module it is likely to performed customized pre-processing strategies so that the luminosity variations do not affect the vision quality control process.

Another example of the application of scene recognition in industrial vision systems is found in quality stations composed by fixed cameras constellations. Suppose that each camera of the constellation captures pictures of one specific component of the supervised objects on a production line that carries products that may present structural variations, for instance the size of a component. The limitation of this layout is that the processing algorithms could decrease their performance when facing random product variations. Then in this case, the scene recognition makes possible to execute customized procedures by having knowledge of the product type is being processed increasing the system performance.

3.4 The Adaptive Quality Enhancement Module

After the image recognition process, the system counts with useful information to perform specific quality enhancement methods and update the current knowledge of the surrounding industrial environment. In this section the correction strategies implemented and their adaptive process are presented.

3.4.1 Correction Strategies

To obtain conclusive results from the processing methods, it is usually necessary to implement pre-processing tasks such as contrast correction, noise reduction, filtering and morphological operations. Those techniques can be adequately grouped in different operating sequences, which tend to correct specific image detractors. Hence, according to the picture parameters analysis, it is likely to determine the eventual image problem and execute the most appropriate strategy for that case.

The current adaptive quality enhancement module implements three quality strategies, namely: contrast enhancement, luminosity correction, and sharpness enhancement. The functional structure of each one and the applied methods are based on a quality-execution time balance that comes from an industrial system's restriction, i.e., the available execution time is reduced and a determined minimum quality level is required. Particularly in the industrial application case study, the available time for multiple image supervisions running on a PC powered with multi-core Intel I5 @ 2.4 GHz rounds 20 s, then it is defined that the pre-processing system dispose of approximately 2 s per image to execute the required correction routines. A brief explanation of the implemented strategies is presented below.

3.4.1.1 Contrast Enhancement Strategy

Contrast correction strategies seek to enhance the apparent visual quality and emphasize specific image features [27]. In this approach, the task is performed by adaptive histogram equalization [33] and morphological filters. The first procedure adapts the intensity scheme defining the image, but might tend to increase the noise. Consequently in a second step, morphological opening and closing operations are performed to reduce the noise without

affecting the highlighted regions. The TN, FL,V, LL and HL parameters are analysed to detect contrast issues.

3.4.1.2 Luminosity Correction Strategy

Image luminosity problems may difficult the identification of the structural scene components. For solving this matter, the designed luminosity strategy performs a histogram adjustment. This method tunes up the intensity scale, correcting dark and overexposure regions. Luminosity detractions can be identified by the E, HL, LL, HS, LS and LM indexes.

3.4.1.3 Sharpness Enhancement Strategy

Sharpness is close related with image focusing and defined edges. Well-focused images will show distinct edges [3] facilitating the contour detection. For the sharp enhancement strategy purposes, blind deconvolution methods [26] were tested with successful quality results. However, the execution time required by these procedures considerably exceeds the available task time.

Laplacian filtering was also addressed in the sharpness enhancement approach, nevertheless the corresponding resulting images did not satisfy the defined quality criteria. Finally, unsharp masking [18] was tested, obtaining a good quality-time performance; consequently this method was chosen for the strategy implementation. In this case, the TN, FL and E parameters are related with sharpness deficiencies.

3.4.2 Optimal Enhancement Strategy Determination

The adaptive quality enhancement module is responsible for selecting the optimal correction strategy for the incoming image problem. For this purpose, a comparison between the indexes of the reference scene and the current image must be performed. The resulting values reveal some parametric tendencies which are compared with defined patterns. Then, based on a max coincidence matching criterion, it is likely to find the main quality problem. If the program identifies multiple problems affecting the image in an equal way, the corresponding correction strategies will be performed sequentially.

Defining \vec{V} (eq. 3.4.1) like the current image's indexes vector, a set of indexes patterns is established as shown in equations 3.4.3–3.4.5. The set contains an ensemble of three vectors exposing the typical tendencies of the single parameters of \vec{V} in the specific cases of contrast, luminosity and sharpness problems. The pattern vectors' components are assigned respectively with -1 or 1 depending if the index value is smaller or greater than the corresponding reference, and 0 if the parameter is not considered for the addressed quality problem.

$$\vec{V} = (TN, FL, E, HSN, LM, V, HL, LL, LSN) \quad (3.4.1)$$

$$\overrightarrow{Vref} = (TN^*, FL^*, E^*, HSN^*, LM^*, V^*, HL^*, LL^*, LSN^*) \quad (3.4.2)$$

$$Contrast = [-1, 1, 0, 0, 0, 1, -1, 1, 0] \quad (3.4.3)$$

$$Luminosity = [0, 0, 0, 1, 0, 0, -1, 1, 1] \quad (3.4.4)$$

$$Sharpness = [-1, 1, 1, 0, 0, 0, 0, 0, 0] \quad (3.4.5)$$

$$\vec{P} = \vec{V} - \overrightarrow{Vref} \quad (3.4.6)$$

Now, let's define \vec{P} (equation 3.4.6) like the difference between \vec{V} and the quality reference vector \overrightarrow{Vref} (equation 3.4.2). By a matching procedure, the patterns set is compared with the corresponding \vec{P} vector of the image in progress. Finally, the matching values obtained in the comparison procedure are normalized and the resulting max coincidence determines which strategy or strategies must be execute to achieve the quality enhancement.

Once the correction strategies are executed, the new image quality is calculated and the program estimates if it was possible to enhance the image and achieve a quality in the range of the expected values, otherwise, a new image acquisition will be requested. Figure 3.7 presents a flux diagram describing the algorithm stages.

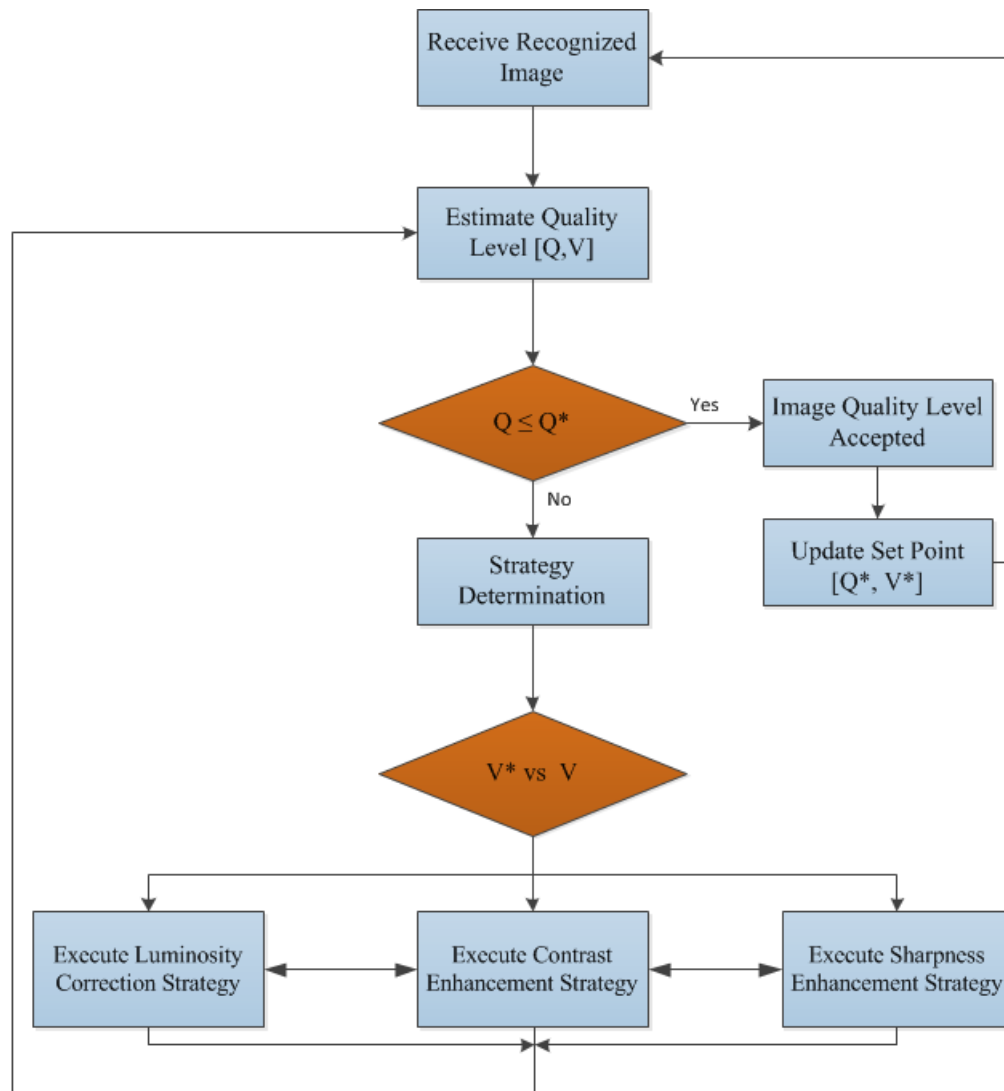


Figure 3.7: *Simplified Schematic of the Adaptive Quality Enhancement Algorithm*

3.4.3 Quality and Parameters Updating

Robustness in industrial systems demands an extensive and updated knowledge of the production data. This information is fundamental for adapting to dynamic process changes and performing the most appropriate actions to achieve optimized results.

At the beginning of the designed system execution, the required reference scenes are composed just by a reduced ensemble of single images. Hence, the available information about the operation environment and the quality aspects are limited.

To solve this issue, the adaptive enhancement module considers the history of the processed pictures. Specifically, the quality $Q_{enhanced}$ and parametric indexes' vector $\vec{V}_{enhanced}$ of those processed images which accomplish the quality criteria are considered to update the corresponding scene reference quality $Q_{ref(old)}$ and parameters vector $\vec{V}_{ref(old)}$. Thus, the respective new scene references $Q_{ref(new)}$ and $V_{ref(new)}$ are obtained, as shown in equations 3.4.7 and 3.4.8.

The updating contribution of a single enhanced image is controlled by a defined factor of preponderance, i.e., a scalar weight or percentage c_i . Conveniently, this constant is set to a small value assigning a greater preponderance to the history of the system. This consideration is fundamental in applications with large history records, e.g., industrial assembly lines that analyse hundreds or thousands of pictures per day.

$$Q_{ref(new)} = c_1 \cdot Q_{ref(old)} + c_2 \cdot Q_{enhanced} \quad (3.4.7)$$

$$\vec{V}_{ref(new)} = c_1 \cdot \vec{V}_{ref(old)} + c_2 \cdot \vec{V}_{enhanced} \quad (3.4.8)$$

The update process refines the acceptance quality threshold and makes the strategies converge to the optimal ones; hence the system becomes more strict and accurate. For instance, if in one specific moment the program applies determined strategies to enhance the current image quality, it is likely than in a posterior instant (after processing a significant sample of the scenario's images) the correction strategies for that scene would vary. Furthermore, the reference quality range tends to narrow while more pictures are analysed. In this case, it is possible that in the present instant the system discards enhanced images that in a previous

moment were accepted.

3.5 Summary

Along this section, the main details relative to the system architecture, its components and algorithms are addressed. In first instance, the pre-processing task scheme is described and discussed, followed by the presentation of the proposed pre-processing architecture.

Afterwards, the first system's operative module (i.e., the scene recognition module) is presented and explained. In this subsection, it is introduced a set of image parametrical indexes and global quality functions, explaining their fundamental role in the image quality estimation and in the system's neural network learning process. Then, the scene recognition method used in the neural network is explained in detail as well as the histogram matching algorithm applied in the identification of complex scenes.

Finally, the second system's module (i.e., the adaptive quality enhance method) is addressed. In this particular, the implemented quality correction strategies, the optimal strategy determination algorithm and the quality and image parameters updating procedure were focused.

Chapter 4

Experimental Results

4.1 Introduction

Along this chapter it is presented the experimental test and validation phase of the proposed approach.

In first instance, the overall system performance, as well as the image quality enhancements achieved are evaluated. Then, in a second stage, it is presented a study concerning the effectiveness of the posterior processing algorithms after the execution of the pre-processing system.

4.2 First Experimental Stage: System Performance and Quality

Along the first experimental stage, the developed adaptive pre-processing system was tested to verify its performance and robustness under diverse image environments. For to this aim, an extensive amount of pictures from different scenarios were processed and analysed. The component modules responses as well as the system's overall performance were evaluated during the pre-processing of BMP format images with a 1280 x 1024 resolution.

In this approach, the architecture was codified and tested under MATLAB® R2011b. A significant part of the design is based on the Neural Network, Global Optimization and

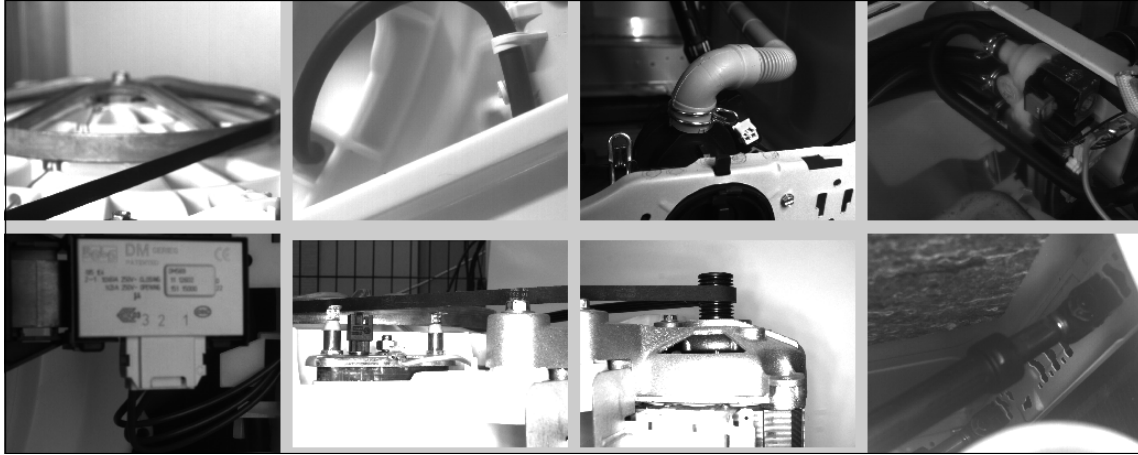


Figure 4.1: *Experimental Image Scenarios Analysed*

Image Processing toolboxes.

Experimental tests were performed using two different multi-core processors, namely, an Intel I7 @ 1.6 GHz and an Intel I5 @ 2.4 GHz. The execution times provided below for the different tasks correspond to the arithmetic mean of the obtained results.

Defining a three known scenarios ensemble, i.e., a system input domain composed by three reference scenes (batch mode learning), the program was simulated. An initial set point routine was executed spending a time of 4.7 s.

In the testing of the scene recognition module, samples proceeding from eight differentiated scenarios were examined. As result, the algorithm correctly recognized and identified 99.7% of the sample images (up to 1800 images analysed), i.e., either when the picture was effectively matched as a known scene or when accurately no possible match was found. The execution time spent for this task rounds 500 ms. Figure 4.1 shows the experimental image references of the studied scenarios.

In the other hand, the adaptive quality enhancement module correctly determined and performed customized correction strategies according to the image indexes analysis and the respective problem identification. After several executions, the system showed an adaptive behaviour converging to the optimal strategies for the specific scenarios. The execution time spent for this task was 1.1 s.

Figure 4.2 illustrates the image quality enhancement achieved in the first experimental

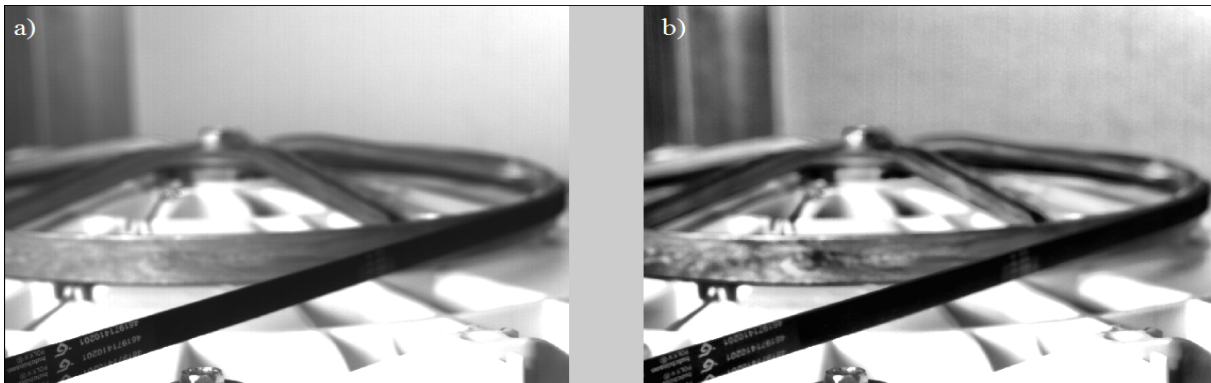


Figure 4.2: *Experimental Scenario 1. a) Input Image b) Enhanced Image*

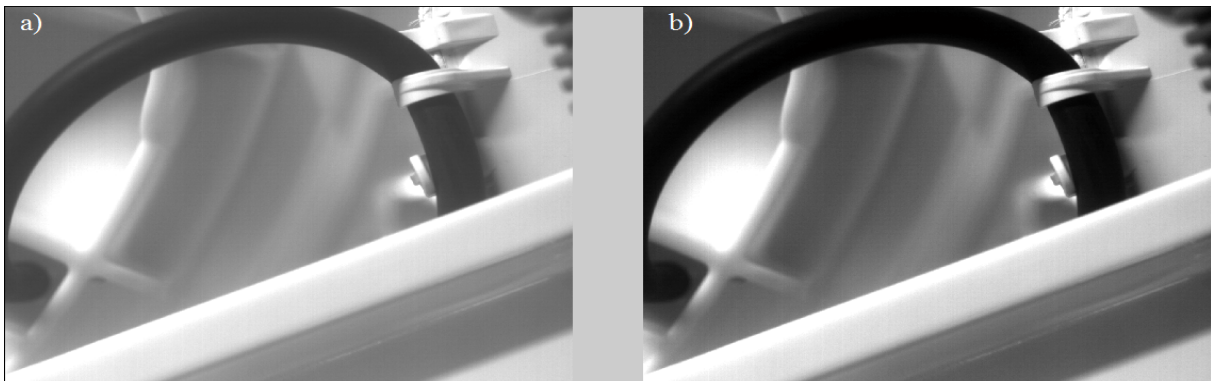


Figure 4.3: *Experimental Scenario 2. a) Input Image b) Enhanced Image*

scenario by performing a sequential execution of the contrast and the sharpness strategies. In this scene, the visual improvement might result slight, however in numerical terms the image has been significantly enhanced. Parametrically speaking, TN increased 28.8%, FL decreased 3.7%, HS reduced 2.4%, while LM improved 9.1%. Consequently, the global quality enhancement obtained was 95.8%.

Regarding to the scenario number two, in Figure 4.3 it is possible to observe the results obtained after the performance of the luminosity correction strategy. In this case, TN increased 72.8%, FL reduced 18.6% and DR increased 18.4 %; all this for a global quality enhancement of 94.3%.

Then, Figure 4.4 presents the pre-processing results achieved in one image belonging to the third scenario. For this scene, the contrast correction strategy was executed obtaining

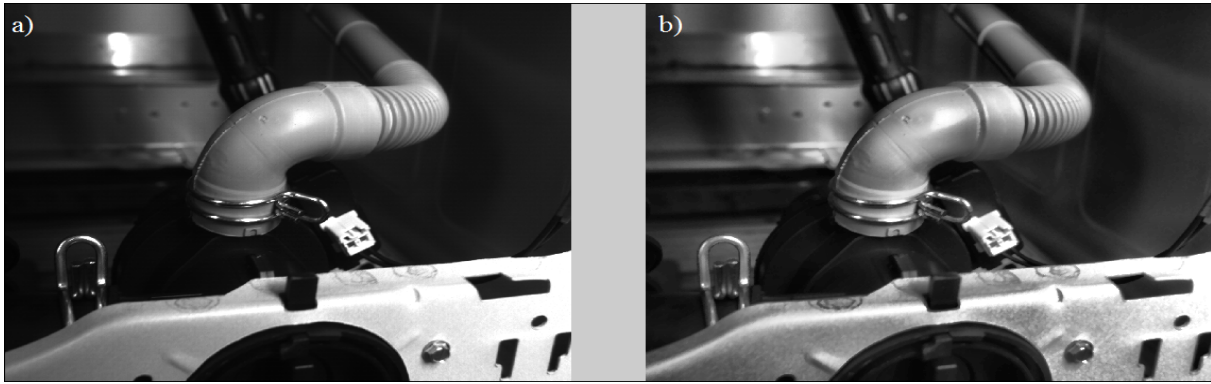


Figure 4.4: *Experimental Scenario 3. a) Input Image b) Enhanced Image*

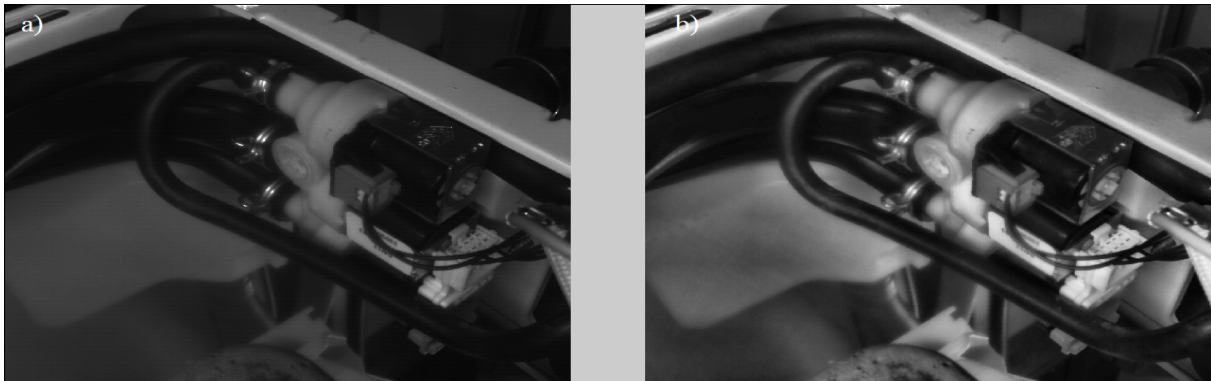


Figure 4.5: *Experimental Scenario 4. a) Input Image b) Enhanced Image*

the following numerical parameters improvements: FL was reduced in 3.2%, HS and LS decreased 15.2% and 94.4% respectively, and LM improved 32.2%. Hence, the overall quality enhancement achieved was 26.8%.

Figure 4.5 shows a quality enhancement performed by means of the contrast strategy execution on an image belonging to the scenario number four. Clearly, the corrected intensity distribution makes the structural image components clearer and more defined for processing purposes. In this case, TN presented an increment of 88.6%, FL decreased 5.9%, HS was reduced 39.7% and LM improved 31.4%. Those changes imply an enhancement of 80.7% in the global quality function.

For the fifth scenario, the pre-processing system applied sharpness and contrast correction strategies. Figure 4.6 allows to appreciate the grey scale distribution correction and the

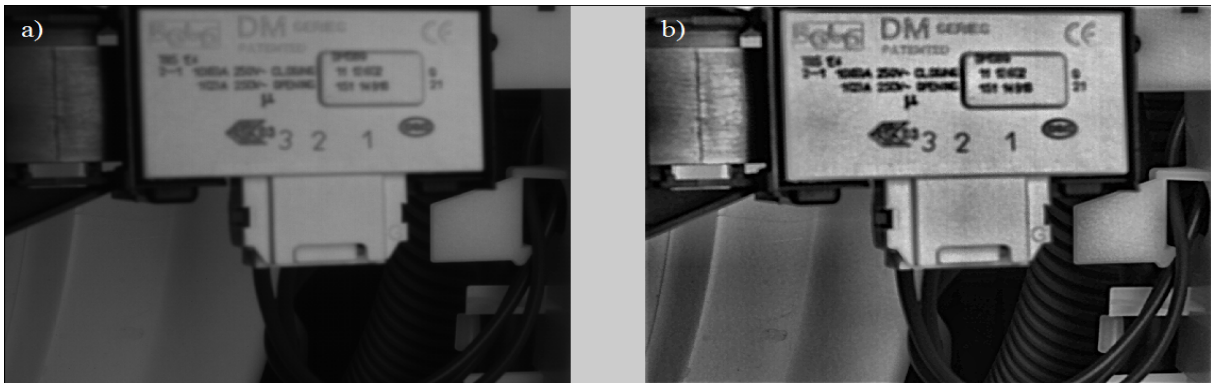


Figure 4.6: *Experimental Scenario 5. a) Input Image b) Enhanced Image*

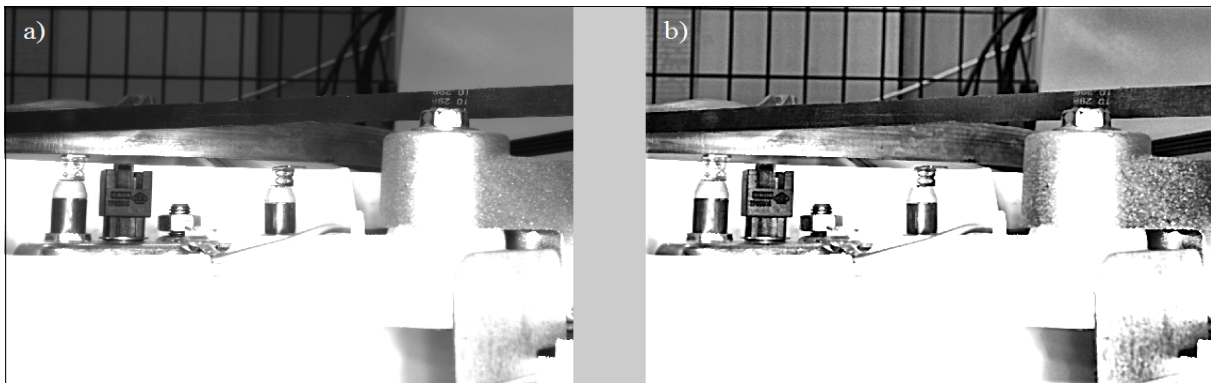


Figure 4.7: *Experimental Scenario 6. a) Input Image b) Enhanced Image*

improvement of the image edges and the definition of the scene. In numerical terms the quality enhancement can be quantified as: TN increased 84.5%, FL was reduced in 8.5%, and LM and DR improved 72.8% and 36.5%, respectively. Finally, the overall improvement achieved was 97.63%.

In regard to the scenario number six, Figure 4.7 shows an example of the quality enhancement achieved in this test. For this experimental image, TN increased 19.4%, FL was reduced in 8.2%, while V improved 14.2% for a total global enhancement of 81%.

In the seventh scenario, the application of the sharpness strategy allowed to obtain an edge enhancement and a definition improvement, as it is possible to observe in Figure 4.8. Then, in this case TN incremented in 71.7 %, LM improved 12.5%, HS reduced 2.4% and DR was expanded in 3.5% for a global quality enhancement of 74.3% .

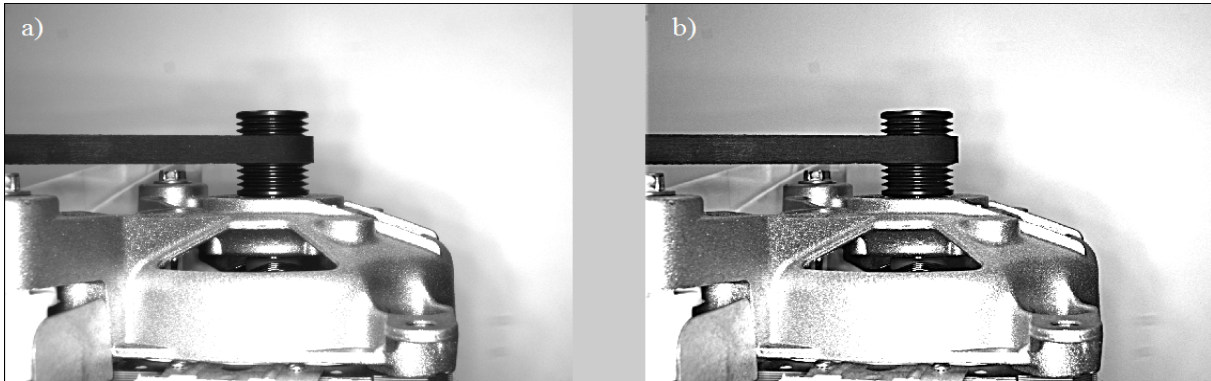


Figure 4.8: *Experimental Scenario 7. a) Input Image b) Enhanced Image*

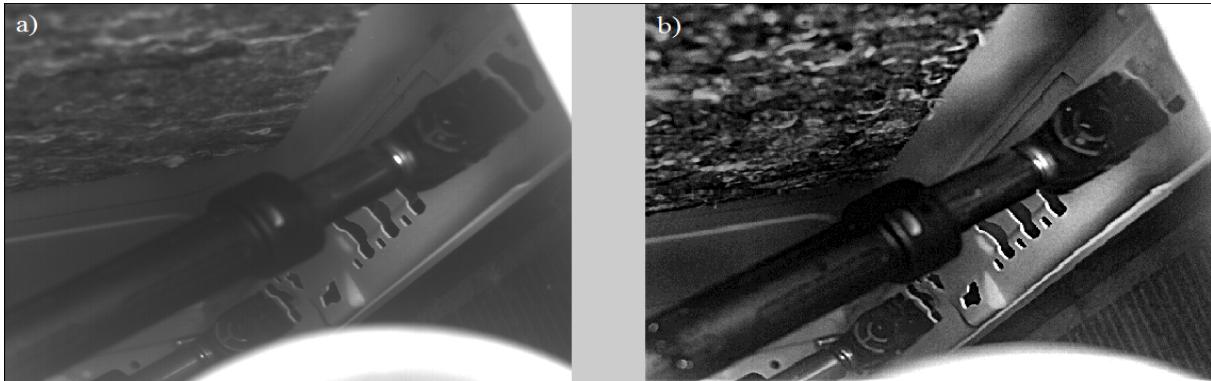


Figure 4.9: *Experimental Scenario 8. a) Input Image b) Enhanced Image*

Finally, for the last scenario, the system determined to apply the sequential execution of the luminosity and sharpness strategies, obtaining the following results: TN increased 90.3 %, FL was reduced in 9.5%, LM improved 10.3%, DR was expanded in 13.7% for an overall enhancement of 90.3%. Figure 4.9 presents the visual results achieved for the experimental image analysed.

4.2.1 Interconnection Test

An additional test was run in order to verify the capability of the architecture to interconnect with other platforms. The results were satisfactory; a successful interconnection with negligible latency was performed with LabVIEW® by means of a MATLAB script.

4.2.2 Statistics Summary

In order to present the statistics of the pre-processing procedure, the designed system generates a *DATA.xls* file containing the initial and final parameters/quality values, as well as the respective variations percentages and tendency graphics. This information can be used to exchange information with other systems (e.g., acquisition, processing or control platforms), with the aim of analysing the current conditions and taking eventual actions regarding to the process.

Table 4.1 exposes a brief summary of the percentage variations of some important image parameters belonging to the sample images of section 4.2, and Table 4.2 presents the global quality enhancements achieve within the different scenarios. Note that the positive numerical values represent index improvements, while the negative ones indicate undesirable index changes.

Finally, the overall system process was executed in 1.6 s with accurate recognition (99.7% effective matchings) and noticeable enhancement results. Considering that for the washing machines production line case study the available time to perform the pre-processing is 2 s, the execution time obtained correctly accomplished the temporal and quality requirements.

Table 4.1: *Some Important Image Parametric Indexes Variations*

Sample from Scenario	TN (%)	FL (%)	LM (%)	DR(%)	HS(%)
1	28.8	3.7	9.1	6.3	2.4
2	72.8	18.6	10.9	18.4	-6.2
3	-4.1	3.2	32.2	0.4	15.2
4	88.6	5.9	31.4	2.3	39.7
5	84.5	8.5	72.8	36.5	34.5
6	19.4	8.2	-2.6	5.2	0.2
7	71.7	12.5	10.1	3.5	2.4
8	90.3	9.5	10.3	13.7	-3.7

Table 4.2: *Image Quality Enhancement Summary*

Scenario	Initial Quality	Final Quality	Enhancement (%)
1	126.7	6.1	95.2
2	858.8	17.3	97.9
3	9.2	6.9	25.0
4	17.4	3.4	80.5
5	5964.9	90.4	98.5
6	32.5	6.4	80.3
7	21.9	5.1	76.8
8	584.2	52.7	90.9

4.3 Second Experimental Stage: Processing Recognition

In the second experimental stage, it is analysed the performance improvement of the processing recognition algorithms due to the quality enhancement achieved by the adaptive pre-processing system execution.

In order to measure this performance enhancement, an inspection procedure based on geometrical matching is implemented on LabVIEW®. For testing purposes, experimental pictures exhibiting different scene conditions (i.e., contrast, luminosity and sharpness variations) are captured by a robot-camera system in a lab environment. The respective hardware configuration used for the image acquiring process is composed by: an articulated robot *DENSO vs-6577G*, a CCD camera *Prosilica GS1380* and a ring light *Metaphase MB-RL205*.

The recognition program designed implements a geometrical matching function that seeks to match a determined object or pattern in the image and generates a matching score in the range of $[0,1000]$, this in order to quantified the effectiveness of the detection. Note that 1000 corresponds to a perfect match, while 0 means that no possible match was found. Additionally, a second score related to the pixels intensity matching (PM) is generated with the aim of complementing the information provided by the first score and identify False-Positive or True-Negative detections.

The testing procedure is focused on the analysis of two experimental scenes which expose different components of a washing unit that must be verified for quality control requirements.

The details about each test, the achieved image enhancement and the processing results are presented in the sections that follow.

4.3.1 Experimental Scene 1

The first experimental set of tests is intended to detect an edge-defined structural object through samples images belonging to the scene number one. The template for the geometric matching used along the experimental procedure is presented in figure 4.10 and an example of the inspection scene is presented in figure 4.11 a).

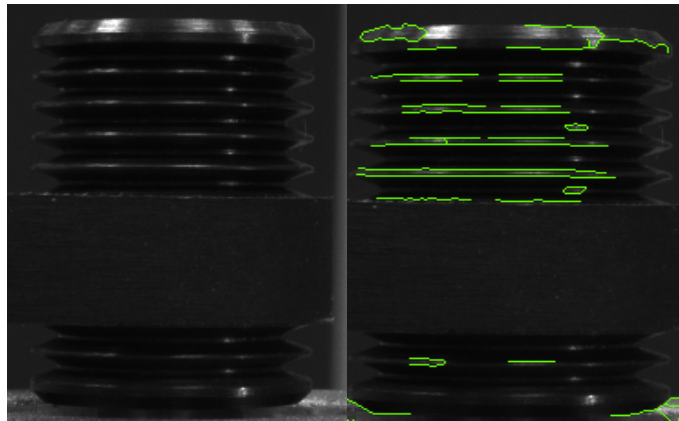


Figure 4.10: *Geometric Matching Template used in the Experimental Scene 1*

The complexity of the recognition task lies on the possible quality detractions that the image can suffer during the acquisition phase and the associated repercussion on the edge detection effectiveness. Therefore, by changing the surrounding environmental conditions of the washing unit, it is possible to experimentally reproduce images with several variations of contrast, luminosity and sharpness. The resulting samples are used to analyse the object detection performance before and after the pre-processing execution. In each test at least a group of 20 sample images is analysed.

For instance, during the study of the algorithm performance on pictures with contrast and sharpness detractions, 6 tests were executed. Figures 4.11 and 4.12 show sample images corresponding to two variations of a quality problem on the area of interest and the resulting

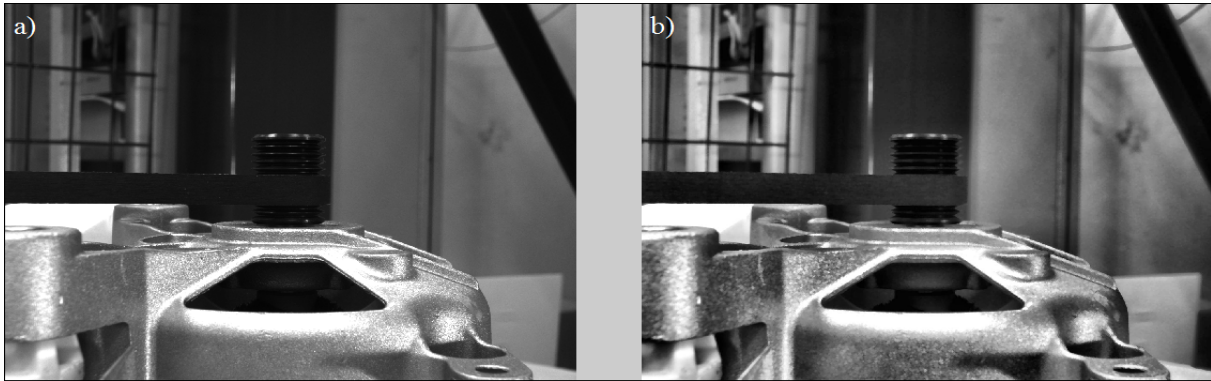


Figure 4.11: Scene 1. Contrast and Sharpness Enhancement Test 1. a) Input Image b) Enhanced Image



Figure 4.12: Scene 1. Contrast and Sharpness Enhancement Test 2. a) Input Image b) Enhanced Image

enhancements achieved by means of the adaptive pre-processing tasks. Notice that each figure corresponds to a separated test.

Following the test procedure, the recognition algorithm was applied to each sample image belonging to the test-group, and the respective matching scores were registered. Figure 4.13 presents an example of the scores obtained *before* (Test1) and *after* the image pre-processing execution (Test1-E). Note how the scores corresponding to the pre-processed images are significantly higher and how the average score increased, passing from 365.77 to 849.77 for an overall improvement of 48.4%.

Notice besides in the scatter graphic of Figure 4.13, the elevated number of null score

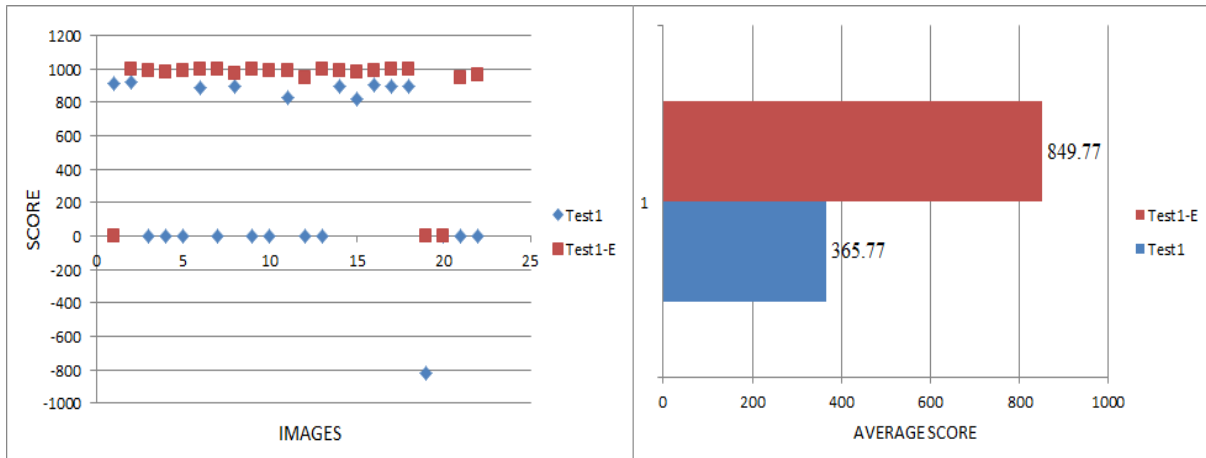


Figure 4.13: *Scene 1. Recognition Enhancement. Score Improvement in Contrast and Sharpness Test*

values (no effective matches found), as well as one negative score (false matching) obtained for the case of non-pre-processed images, and the consecutive reduction of these cases achieved by means of the system execution. In fact, those recognition limitations are due to image quality detractions (in this case poor contrast and blurring), which affect the edge detection and the surface parametrization tasks executed by the recognition algorithms.

Figure 4.14 sums up the results obtained in the six contrast-sharpness tests ran, presenting the score averages and the respective standard deviations associated to each one. The obtained graphics allow to observe significant increments of the matching averages and reductions of the standard deviations in the images after the pre-processing execution.

Note that in the case of the test number five, the score average obtained for the images without pre-processing exhibits a negative value. This particular is related with the presence of False-Positive and True-Negative detections in the test.

In an analogous way, a set of seven luminosity tests was performed. Figures 4.15 and 4.16 present two samples of luminosity detractions and the contribution of the image pre-processing system on the quality enhancement.

In figure 4.17, it is possible to observe the single scores obtained for each image of one luminosity test-group, as well as the average score of the test. Once again the matching improvement is noticeable for the images after the pre-processing procedure. In this specific

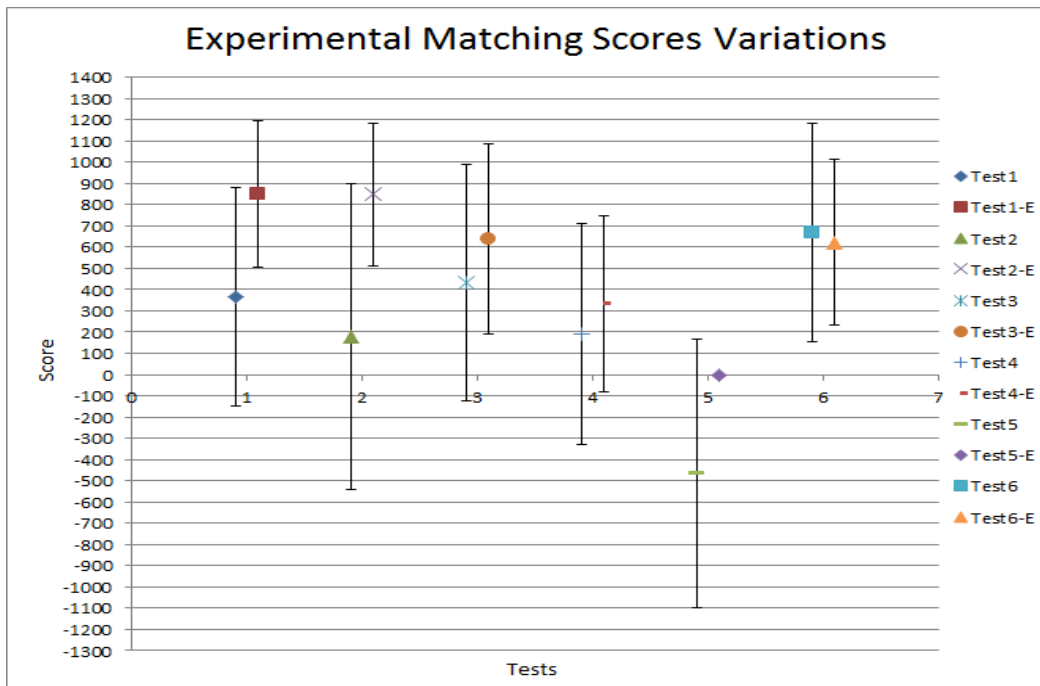


Figure 4.14: Scene 1. Experimental Matching Scores. Contrast and Sharpness Test

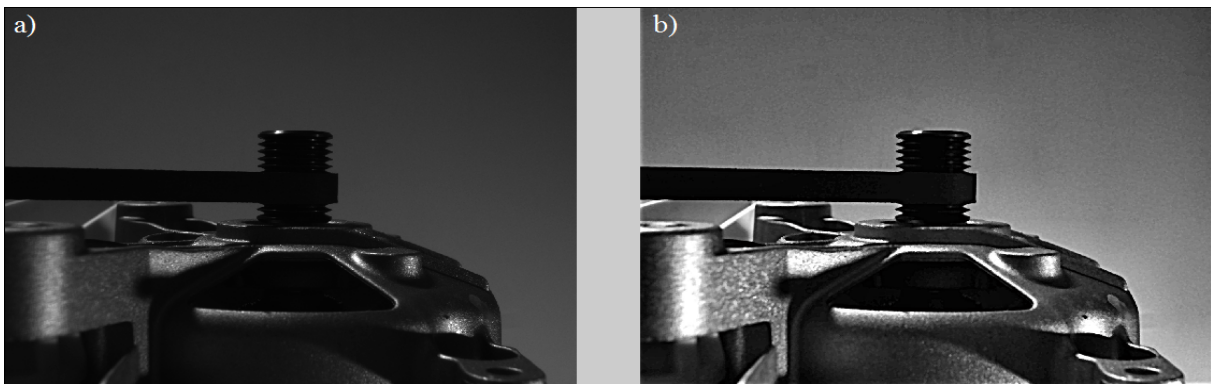


Figure 4.15: Scene 1. Luminosity Enhancement Test 1. a) Input Image b) Enhanced Image



Figure 4.16: Scene 1. Luminosity Enhancement Test 2. a) Input Image b) Enhanced Image

test the score passed from 486.41 to 920.18 what implies an overall enhancement of 43.77%.

This test also exhibits several cases of recognition limitations (null scores in Test1) related with deficiencies in the luminosity of the scene, namely: underexposure or overexposure.

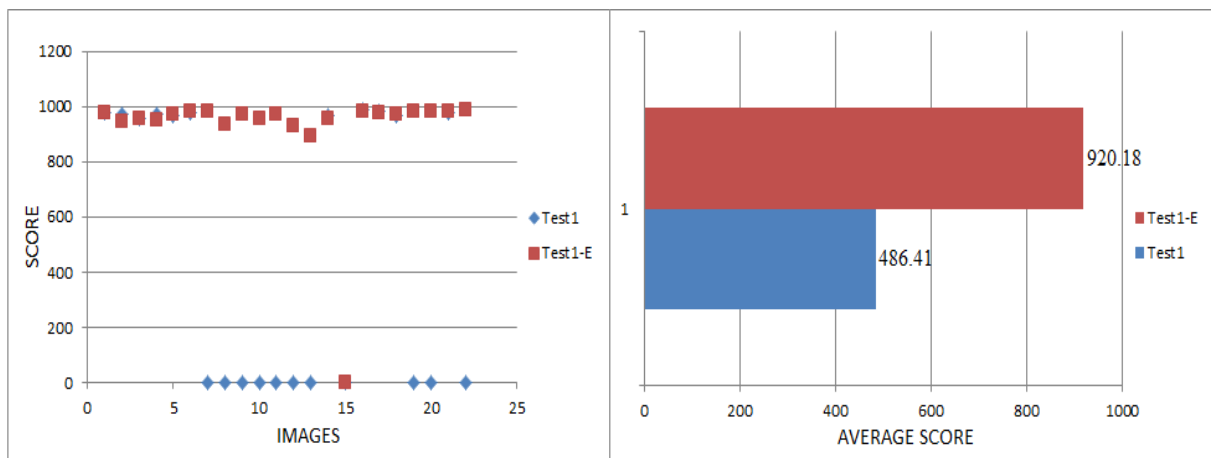


Figure 4.17: Scene 1. Recognition Enhancement. Score Improvement in Luminosity Test

A summary of the luminosity tests results is presented in figure 4.18. As it is possible to notice, the global performance of the identification algorithm is higher and presents less variability in the particular of images after the pre-processing execution.

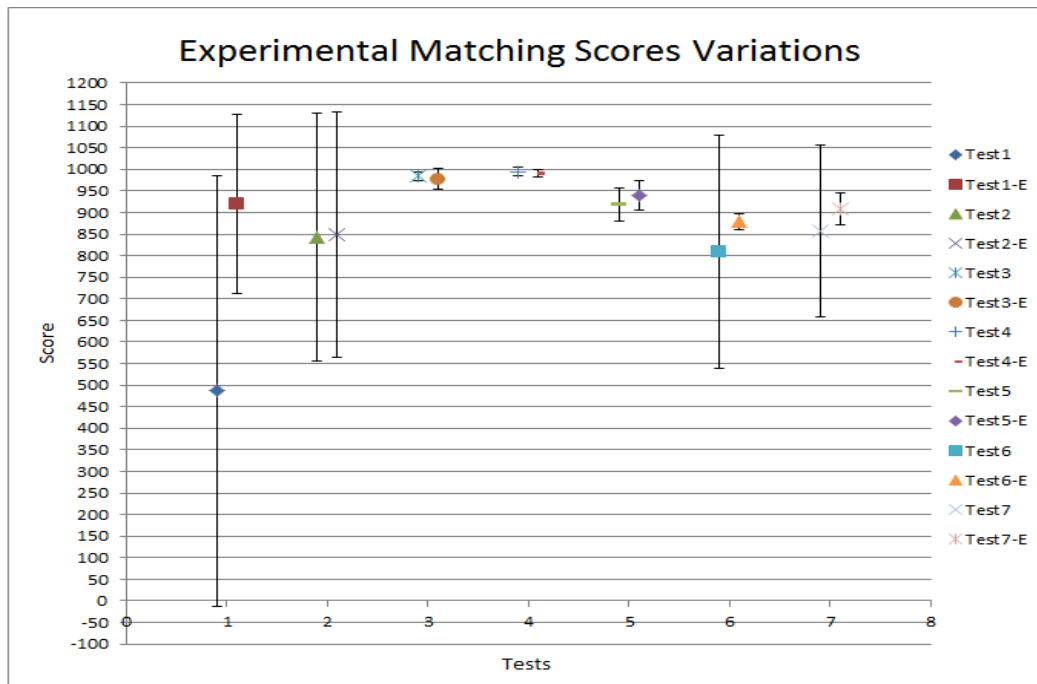


Figure 4.18: Scene 1. Experimental Matching Scores. Luminosity Test

4.3.2 Experimental Scene 2

Turning to the second experimental scene, the set of tests performed in this occasion seek to quantify the scores obtained in the geometrical matching of the template shown in figure 4.19, along an ensemble of experimental images like the one presented in figure 4.20 a).

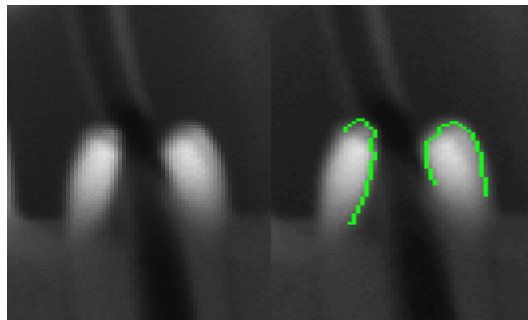


Figure 4.19: Geometric Matching Template used in the Experimental Scene 2

The test procedure starts analysing the case of scene luminosity problems in the image and their effect on the matching algorithm, for which a set of five tests is studied. Figures 4.20 and 4.21 show the results obtained in the stage of quality enhancement pre-processing for some sample images belonging to this scene.

Once applied the pre-processing tasks on the sample, the recognition algorithm is executed in the totality of images. Partial results of one test are shown in figure 4.22, which let to appreciate a higher data repeatability and an average score increment of 17.54% in the sample segment with pre-processing.

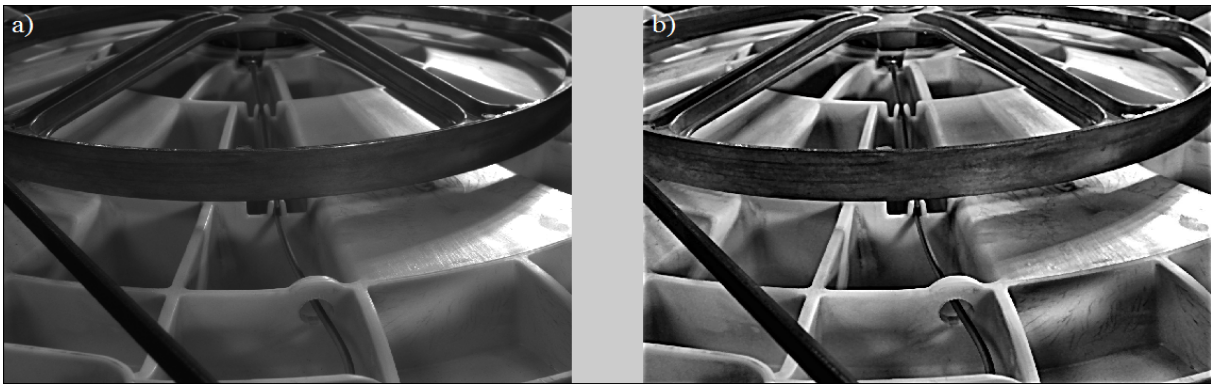


Figure 4.20: Scene 2. Luminosity Enhancement Test 1. a) Input Image b) Enhanced Image

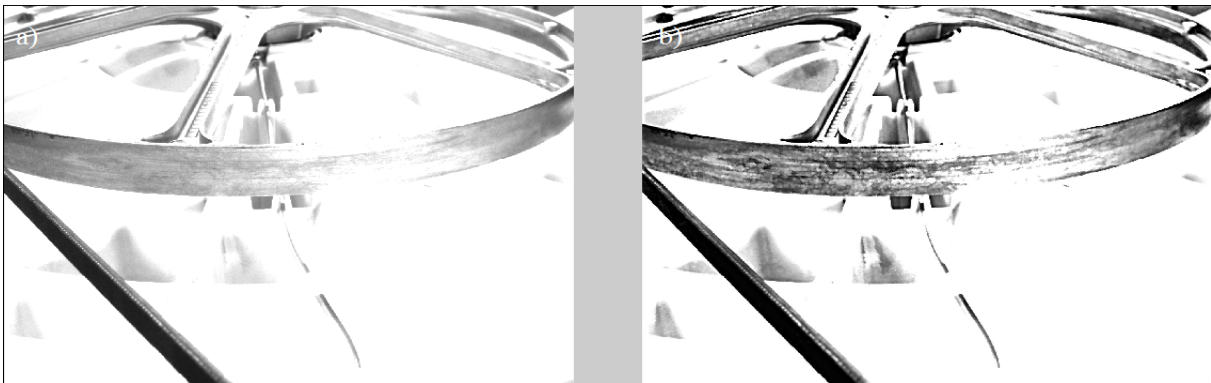


Figure 4.21: Scene 2. Luminosity Enhancement Test 2. a) Input Image b) Enhanced Image

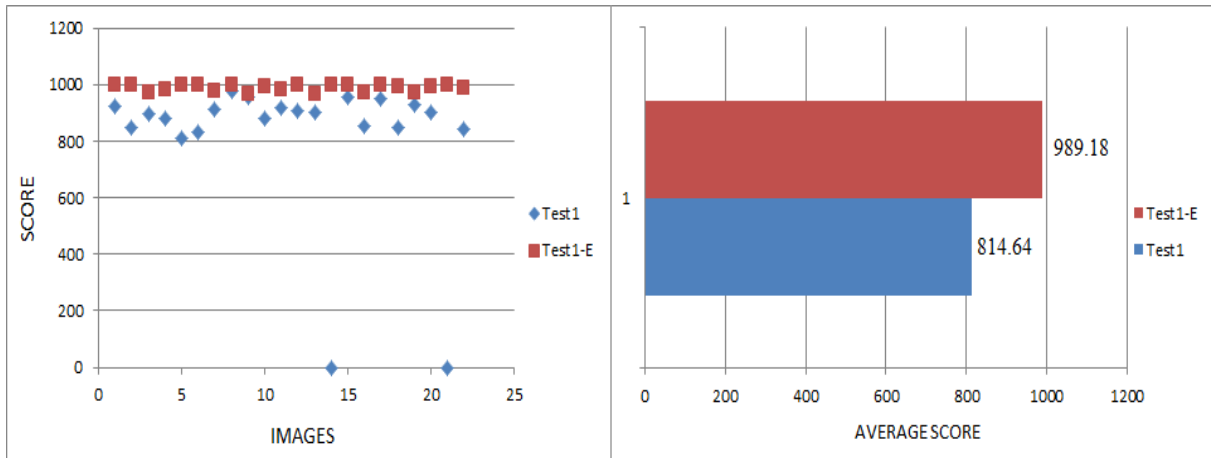


Figure 4.22: Scene 2. Recognition Enhancement. Score Improvement in Luminosity Test

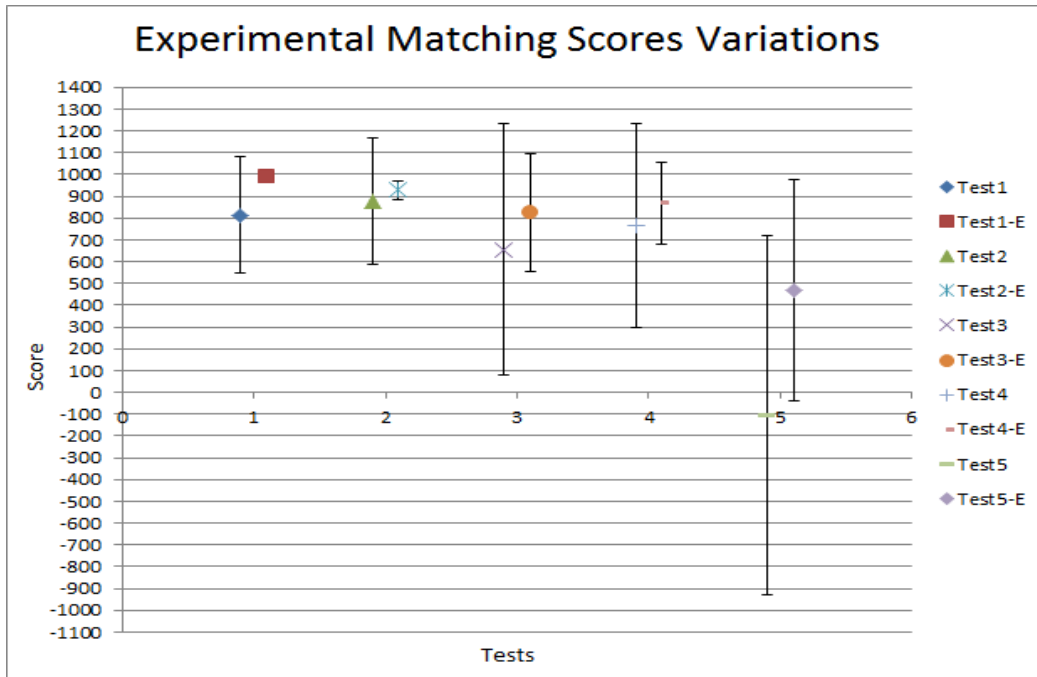


Figure 4.23: Scene 2. Experimental Matching Scores. Luminosity Test

Figure 4.23 presents a summary of the results achieved in the five luminosity tests performed. The tendency found in this case is similar that the one observed in previous tests,

i.e., the scores obtained in images after the pre-processing stage exhibit higher matching averages, more data repeatability and consequently smaller standard deviations.

Finally, it is addressed the repercussions of contrast and sharpness detractions on the identification of the second experimental scene. Figures 4.24 and 4.25 show some examples of the quality enhancements achieved during the pre-processing execution.

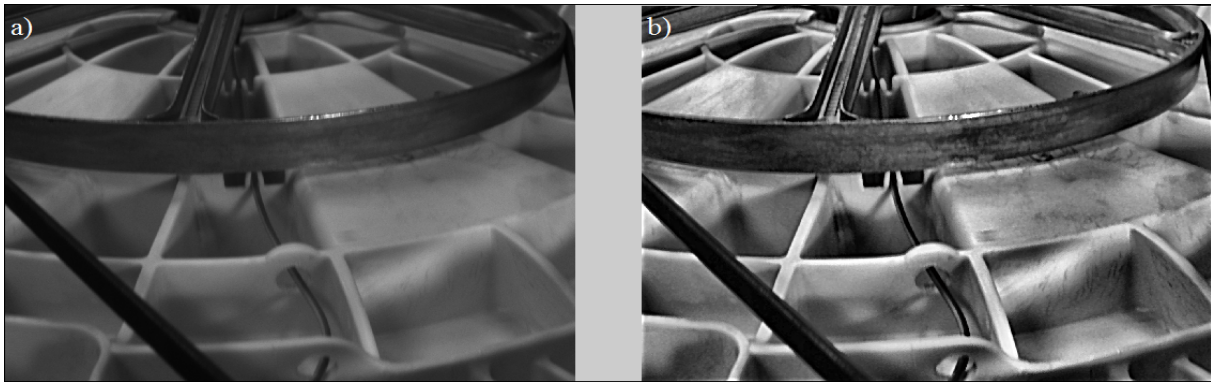


Figure 4.24: Scene 2. Contrast and Sharpness Enhancement Test 1. a) Input Image b) Enhanced Image

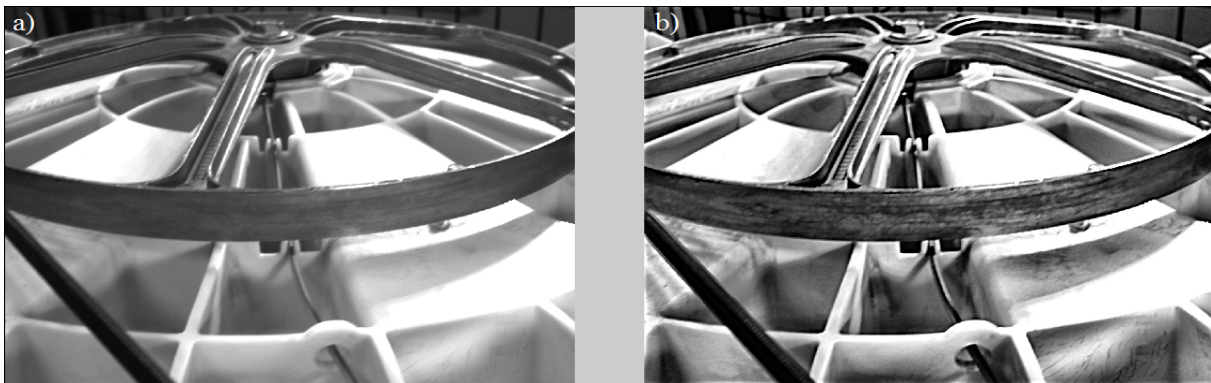


Figure 4.25: Scene 2. Contrast and Sharpness Enhancement Test 2. a) Input Image b) Enhanced Image

Figure 4.26 presents detailed information about the matching scores registered in one of the five tests ran. In this particular case the pre-processing execution allowed the a 12.46% improvement in the overall average score.

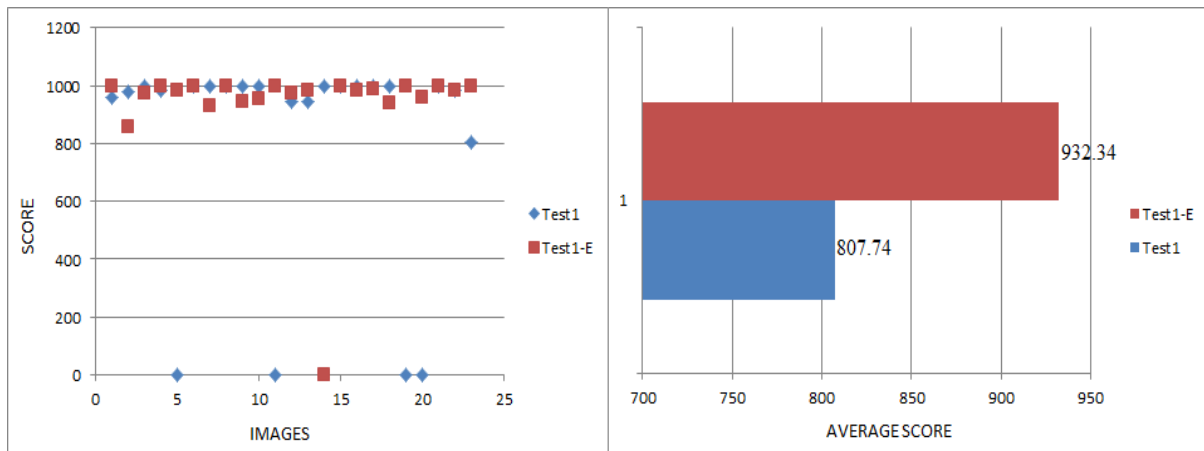


Figure 4.26: Scene 2. Recognition Enhancement. Score Improvement in Contrast and Sharpness Test

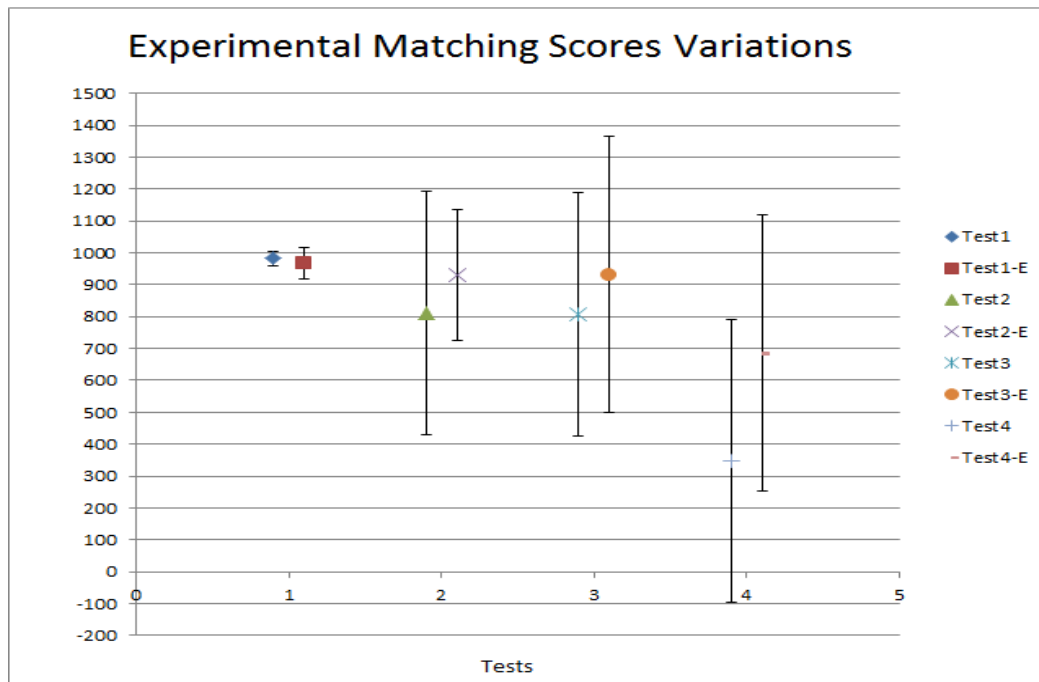


Figure 4.27: Scene 2. Experimental Matching Scores. Contrast and Sharpness Test

The final results of the contrast and sharpness set of tests are presented in figure 4.27. One

more time, the matching scores achieved after the execution of the pre-processing strategies present a higher arithmetic average and less data variability.

4.3.3 Statistics Summary

Table 4.3 presents an overall summary of the results obtained during the second experimental stage of the project. The report shows information related to the global matching scores and the respective standard deviations achieved in several tests performed *before* and *after* the pre-processing system execution on the experimental scenes described in the subsections 4.3.1 and 4.3.2.

The simultaneous increment of the matching score averages and diminution of the standard deviations obtained, allow to enhance the reliability, confidence and robustness of the detection systems. For instance, if a threshold is defined in order to accept or not detected matchings as true, the action of the pre-processing tasks can significantly boost the threshold value guaranteeing an smaller range of detection uncertainty, fact that leads to a significant reduction of false-positive and true-negative detections and a consequent enhancement of the inspection process efficiency.

Table 4.3: *Summary of the Matching Score Averages and Standard Deviations*

		Luminosity Test		Contrast & Sharpness Test	
		<i>before</i>	<i>after</i>	<i>before</i>	<i>after</i>
Exp. Scene 1	Score	856.79	912.91	228.53	549.12
	Deviation	192.42	116.29	576.11	322.61
Exp. Scene 2	Score	601.47	815.52	737.64	879.21
	Deviation	485.37	204.30	307.34	280.26

Chapter 5

Conclusions and Future Work

5.1 Conclusions

The problem addressed in this thesis comprehends the development of an adaptive system for the pre-processing of images captured in industrial production scenarios. In this approach, real pictures from a washing machines production line are used for the testing and validation stages. The work implementation lies on the context of automated systems for quality control purposes.

The system comprises two main operational units namely, a scene recognition module and an adaptive quality enhancement module. The first module is responsible for the image scene identification, while the second one performs the customized selection and execution of correction strategies of contrast, sharpness and luminosity according to the image quality problems.

The experimental results obtained show a robust system performance with short execution times. The required initial set point routine was performed in approximately 4.7 s (note that this routine is executed just one time during the program's set-up phase). The overall pre-processing system execution time rounds 1.6 s, which is an appropriate value according to the 2 s application's available interval. Note that those times correspond to the enhancement process of images with a 1280 x 1024 resolution. Eventually, a considerable time reduction may be achieved if specific image areas of interest are defined.

Moreover, customized correction strategies were accurately applied with significant image quality enhancements. The continuous program execution allowed the system collecting information about the scenarios quality deficiencies and thus selecting the optimal enhancement methods for each image scene.

Additional recognition tests demonstrate a significant increment of the scores averages and a diminution of the associated standard deviations, facts that imply an overall increment of the acceptance threshold that leads to the reduction of false matchings and consequently more confident and robust identification processes.

The successful implementation of a dynamic scene recognition algorithm, that allows the execution of customized quality correction strategies, it is capable of considering the system's history in order to self-adapt and considerably enhances the effectiveness of the identification tasks, constitutes a contribution with an added value for the automated industrial quality control field.

Currently, an interconnection of the designed architecture with an acquisition interface, a feature extraction system and a multi-agent control platform, is being developed with promising results.

5.2 Future Work

As future work, the system architecture could be modified into a multi-thread prototype capable of performing parallel indexes extraction and correction strategies executions. Moreover, additional image indexes can be addressed in order to increase the available image data and its respective parametrization. This issue might allow the system to process and recognize precisely images between similar scenarios.

Besides, experimental correlations between the indexes behaviours and the identification scores can be explored, this in order to approximate a criteria or function that might be useful for the implementation of additional quality correction strategies.

Finally, possible codifications of the architecture on other computational platforms (e.g., Python, C++, among others) may be developed with the aim of achieving shorter execution times.

Bibliography

- [1] T. Acharya and A.K. Ray. *Image Processing: Principles and Applications*. Wiley-Blackwell, 2005.
- [2] M.A. Aguilar, A.J. Arguelles, and C. Yanez. A real time artificial vision implementation for quality inspection of industrial products. *Proceedings of the Conference on Electronics, Robotics and Automotive Mechanics*, pages 277–282, 2008.
- [3] S. Ball. Generating images with enhanced depth of field by creating a seamless mosaic from a sequence of images with different focal settings. Technical report, 2004.
- [4] B. Bing. *Feature extraction and matching in content-based retrieval of functional Magnetic Resonance Images*. PhD thesis, Rutgers The State University of New Jersey-New Brunswick, 2009.
- [5] A. Bouridane. *Imaging for Forensics and Security: From Theory to Practice*. Springer, 2009.
- [6] Y.S. Boutalis, K. Tsirikolias, B.G. Metrtzios, and I.T. Andreadis. Implementation of morphological filters using coordinate logic operations. *Elsevier Pattern Recognition*, 35(1):287–198, 2002.
- [7] A.C. Bovik. *Handbook of Image and Video Processing*. Academic Press, 2005.
- [8] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
- [9] T. Braunl, S. Feyrer, W. Rapf, and M. Reinhardt. *Parallel Image Processing*. Springer-Verlag Berlin Heidelberg, 2001.

- [10] M. Campos. Inspection of bottles crates in the beer industry through computer vision. *Proceedings of the Annual Conference on IEEE Industrial Electronics Society*, pages 1138–1143, 2010.
- [11] P. Castellini, C. Cristalli, M. Foehr, P. Leitao, N. Paone, I. Schjolberg, J. Tjonnas, C. Turrin, and T. Wagner. Towards the integration of process quality control using multi-agent technology. *Proceedings of the 37th Annual Conference on IEEE Industrial Electronics Society*, pages 421–426, 2011.
- [12] V. Das, R. Vijayakumar, N.C. Debnath, J. Stephen, et al. *Information Processing and Management: International Conference on Recent Trends in Business Administration and Information Processing*. Springer, 2010.
- [13] G. Dreyfus. *Neural Networks: Methodology and Applications*. Springer-Verlag Berlin Heidelberg, 2004.
- [14] S. Du and R.K Ward. Adaptive region-based image enhancement method for robust face recognition under variable illumination conditions. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(9):1165–1175, 2010.
- [15] H.A. ElMaraghy. Flexible and reconfigurable manufacturing systems paradigms. *International Journal of Flexible Manufacturing Systems*, 17(4):261–276, 2005.
- [16] S. Gallant. *Neural Network Learning and Expert Systems*. MIT Press, 1993.
- [17] R. Garg, B. Mittal, and S. Garg. Histogram equalization techniques for image enhancement. *International Journal of Electronics and Communication Technologies*, 2:107–111, 2011.
- [18] R.C. Gonzalez and R.E Woods. *Digital Image Processing*. Prentice Hall, Upper Saddle River, New Jersey, 2002.
- [19] R.C. Gonzalez, R.E Woods, and S.L Eddins. *Digital Image Processing using MATLAB®*. Prentice Hall, Upper Saddle River, New Jersey, 2004.

- [20] S. Haykin. *Neural Networks: A Comprehensive Foundation, 2nd ed.* Englewood Cliffs, NJ: Prentice Hall, 1999.
- [21] M. Hudson, M. Hagan, and H. Demuth. *Neural Network Toolbox™ User's Guide R2011b.* MathWorks Inc, 1992–2011.
- [22] National Instruments™. *NI Vision Concepts Manual.* National Instruments Corporation, 2000–2008.
- [23] B. Janhe and H. Haussecker. *Computer Vision and Applications.* Academic Press Inc., 2000.
- [24] T. Kohonen. An introduction to neural computing. *Neural Networks*, 1:3–16, 1988.
- [25] T. Kondo. Medical image recognition by using logistic gmdh-type neural networks. *Proceedings of the 40th SICE Annual Conference*, pages 259–264, 2001.
- [26] D. Krishnan. Automatic contrast enhancement for low contrast images: A comparison of recent histogram based techniques. *Proceedings of the IEEE Computer Vision and Pattern Recognition*, pages 233–240, 2011.
- [27] R. Lakshmanan, M. Nair, M. Wilscy, and R. Tatavarti. Automatic contrast enhancement for low contrast images: A comparison of recent histogram based techniques. *Proceedings of the International Conference on Computer Science and Information Technology*, pages 269–276, 2008.
- [28] J.L. Lima and P.G. Costa. A modular approach to real-time cork classification using image processing. *Proceedings of the IEEE Conference on Emerging Technologies and Factory Automation*, 2:361–368, 2005.
- [29] S. Liqin. Edge detection on real time using log filter. *Proceedings of the International Symposium on Speech, Image Processing and Neural Networks*, 1:37–40, 1994.
- [30] I. Lopez, J.I. Lopez, and R. Osorio. Robot vision methodology for assembly manufacturing tasks. *Proceedings of the Conference on Electronic, Robotics and Automotive Mechanics*, pages 289–294, 2007.

- [31] P. Maria-Sanchez. *Neuronal Risk Assessment System for Construction Projects*. Expert Verlag, 2005.
- [32] MathWorks. *Optimization Toolbox™ User's Guide R2012a*. MathWorks Inc, 1990–2012.
- [33] MathWorks. *Image Processing Toolbox™ User's Guide R2011b*. MathWorks Inc, 1993–2011.
- [34] B. Muller, J. Reinhardt, and M.T Strickland. *Neural Networks: An Introduction*. Springer-Verlag Berlin Heidelberg, 1995.
- [35] V. Murino, G.L. Foresti, and C. Regazzoni. Adaptive camera regulation for investigation of real scenes. *IEEE Transactions of Industrial Electronics*, 43(5):588–600, 1996.
- [36] S. Nagabhushana. *Computer Vision and Image Processing*. New Age International Pvt Ltd Publishers, 2009.
- [37] D.H Nguyen and B. Widrow. Neural networks for self-learning control systems. *IEEE Control System Magazine*, 10(3):18–23, 1990.
- [38] V. Ortman. Real-time object recognition based on active vision and sequential analysis. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 6:3325–3328, 1999.
- [39] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9:62–66, 1979.
- [40] J Park, H.C. Chao, M. Obaidat, and J. Kim. *Computer Science and Converge*. Springer, 2012.
- [41] S. Patil. Noise reduction in images using statistical filtering. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 3:1142–1145, 1982.
- [42] A. Rezvanian, K. Faez, and F. Mahmoudi. A two-pass method to impulse noise reduction from digital images based on neural networks. *Proceedings of the International Conference on Electrical and Computer Engineering*, pages 400–405, 2008.

- [43] R. Rojas. *Neural Networks: A Systematic Introduction*. Springer-Verlag Berlin Heidelberg, 1996.
- [44] R.G. Rosandich. Havnet: A new neural network architecture for pattern recognition. *Neural Networks*, 10(1):139 – 151, 1997.
- [45] I.W. Sandberg, J.T Lo, C.L Fancourt, J.C Principe, S. Katagiri, and S.Haykin. *Nonlinear Dynamical Systems: Feedforward Neural Network Perspectives*. John Wiley and Sons, Inc, 2000.
- [46] D.V. Satish. Image enhancement and noise reduction using wavelet transform. *Proceedings of the Midwest Symposium on Circuits and Systems*, 2:989–992, 1997.
- [47] V.A. Sujana and M.P. Mulqueen. Image identification using the segmented fourier transform and competitive training in the havnet neural network. *Proceedings of the International Conference on Image Processing*, 1:489–492, 2001.
- [48] P. Tugba. Salt and pepper noise reduction in images using adaptive fuzzy filter. *Proceedings of the Signal Processing and Communications Applications Conference*, pages 1–4, 2012.
- [49] R.A. Wilson and F.C. Keil. *The MIT Encyclopedia of the Cognitive Sciences*. MIT Press, 2001.
- [50] L. Xian. Artificial intelligence and modern sports education technology. *Proceedings of the International Conference on Artificial Intelligence and Education*, pages 772–776, 2010.
- [51] W. Xianmin. Research of neural networks in image anomaly identification application. *Proceedings of the International Conference on Computational and Information Sciences*, pages 133 – 136, 2011.
- [52] X. Xiao and X. Zhang. An improved unsharp masking method for borehole image enhancement. *Proceedings of the International Conference on Industrial Mechatronics and Automation*, 2:349–352, 2010.

- [53] Y. Yao, B. Abidi, and M. Abidi. Digital imaging with extreme zoom: System design and image restoration. *Proceedings of the IEEE International Conference on Vision Systems, New York*, page 52, 2006.
- [54] G. Ying, L. Miao, Y. Yuan, and H. Zelin. A study on the method of pre-processing for recognition of crop diseases. *Proceedings of the International Conference on Advanced Computer Control*, pages 202–206, 2009.
- [55] F. You. The application of computer vision algorithms in touching characters recognition of industry products. *Proceedings of the International Conference on Mechanic Automation and Control Engineering*, pages 523–527, 2010.
- [56] M.L Zhang and Z.H Zhou. Multilabel neural networks with applications to functional genomics and text categorization. *IEEE Transactions on Knowledge and Data Engineering*, 18(10):1338 – 1351, 2006.
- [57] X.P. Zhang. Space-scale adaptive noise reduction in images based on thresholding neural network. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 3:1889–1892, 2001.
- [58] X.S. Zhang. *Nonconvex Optimization and its Applications: Neural Networks in Optimization*. Springer, 2000.