Quality Inspection of Textile Artificial Textures Using a Neuro-Symbolic Hybrid System Methodology

VIANEY GUADALUPE CRUZ SÁNCHEZ[1], OSSLAN OSIRIS VERGARA VILLEGAS[2],
GERARDO REYES SALGADO[1], HUMBERTO DE JESÚS OCHOA DOMÍNGUEZ[3]
[1]Computer Science Department
Centro Nacional de Investigación y Desarrollo Tecnológico (cenidet)
Interior Internado Palmira s/n, Col. Palmira, P. C. 62490, Cuernavaca, Morelos
MEXICO
[2]Industrial and Manufacturing Department, Electrical and Computer Department
[3]Electrical and Computer Engineering Department
Universidad Autónoma de Ciudad Juárez
Avenida del Charro No. 450 Norte, Zona Pronaf, P. C. 32310, Ciudad Juárez, Chihuahua
MEXICO
{vianey, greyes}@cenidet.edu.mx, {osslan, hochoa}@ieee.org

Abstract: - In the industrial sector there are many processes where the visual inspection is essential, the automation of that processes becomes a necessity to guarantee the quality of several objects. In this paper we propose a methodology for textile quality inspection based on the texture cue of an image. To solve this, we use a Neuro-Symbolic Hybrid System (NSHS) that allow us to combine an artificial neural network and the symbolic representation of the expert knowledge. The artificial neural network uses the CasCor learning algorithm and we use production rules to represent the symbolic knowledge. The features used for inspection has the advantage of being tolerant to rotation and scale changes. We compare the results with those obtained from an automatic computer vision task, and we conclude that results obtained using the proposed methodology are better.

Key-Words: - Computer Vision, Neuro-Symbolic Hybrid Systems, Artificial Neural Networks, Production Rules

1 Introduction

The Neuro-Symbolic Hybrid System (*NSHS*) is a system inspired in the human behavior, due that is based on the way that humans can obtain a solution for a simple problem. In order to simulating this behavior the NSHS are built integrating the advantages offered by Artificial Neural Networks (*ANN*) and symbolic representations (e. g. Production Rules (*PR*)).

The NSHS offers several advantages that have been used to solve problems as: control, image processing, monitoring and pattern recognition [1] [2]. The ANN has been used to obtain and process the numeric knowledge, whereas the PR has been used to express the symbolic knowledge of a Human Expert.

The NSHS is a very active research area in artificial intelligence [3]. The great challenge is: *How to select the better hybrid architecture?* [4]. There are several works that attempt to explain the different types of Hybrid Systems [5], [6]. There exist a set of criteria that allow classify the NSHS. However, this classification is not enough to answer the question.

We develop a NSHS and a set of strategies based on several criteria proposed by [7]. The methodology developed is used to solve quality inspection problems and the strategies allow us to select the best Neuro-Symbolic architecture type in order to find a good solution for an specific problem.

The quality inspection problems always have the numeric or symbolic knowledge incomplete [8][9]. If we can integrate both knowledge types, the system can offer better results. The methodology was tested with a problem of quality inspection on artificial textures [10]. Quality inspection is very important in the manufacturing industry, due it is necessary to maintain quality standards. In the following sections we present a description of the methodology and the tests and results obtained for the textures inspection.

2 Proposed Methodology

The role of a computer vision system (*CVS*) for an automatic inspection process is very important, due it can achieves the quality inspection of any piece. The typical process of a CVS has six stages [11]. Nevertheless, during the process, each stage presents several difficulties that sometimes are impossible to solve using only one computer vision technique.

The NSHS has demonstrated its effectiveness in several computer vision systems [12], [13]. However, typically the NSHS has been implemented ignoring some details that have a direct correspondence with the selected hybrid architecture.

We create a NSHS Methodology to select the hybrid architecture suitable for a specific problem, composed by eight stages: a) problem description, b) problem decomposition, c) approach type selection, d) technique base building, e) technique selection, f) strategies selection, g) architecture type implementation and h) results evaluation, (See figure 1).

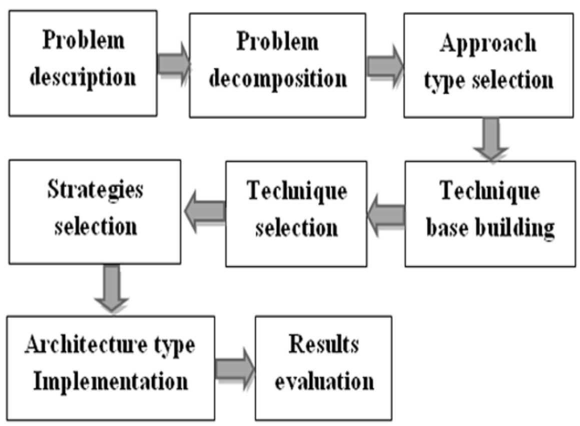

Fig. 1. The eight phases of the proposed methodology.

2.1 Problem description

This stage is very important in order to understand the necessities of the problem.

- *Problem to solve.* The expert defines the problem to be solved. In a lot of visual inspection problems the solutions are focused into maintain certain quality level.
- *Goal definition.* Define what we need: better efficiency, better execution, speed, etc.
- *Locate the problem in a stage of the computer vision process.* It is necessary to locate the problem in the CVS to delimit it.

In the figure 2 we show the problem description phase. We can see that we have a computer vision problem which needs to be described in order to understand the problem. Later, we need to locate the problem at any stage of computer vision process.

2.2 Problem decomposition

The main goal is to divide the problem in a set of sub-problems that allow us to solve them easily. The problem decomposition is made by the expert in the particular domain. A tree is used to represent the set of simplest task detected (see figure 3).

After decomposition, we obtain important information of each sub-task. The information is integrated by assigning a name to each task, detecting the problem that is necessary to solve and the stage of computer vision process where each task is located.

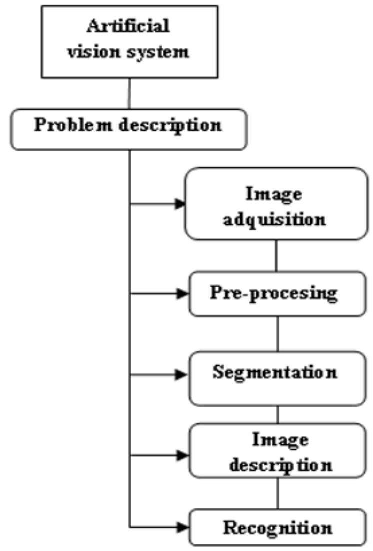

Fig. 2. Problem description phase.

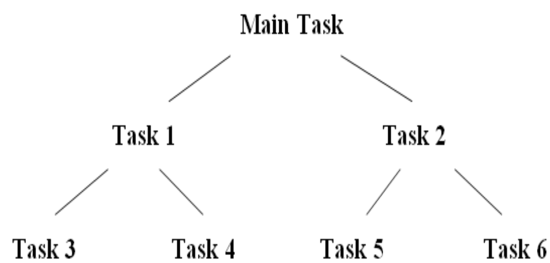

Fig. 3. Task decomposition tree.

2.3 Approach type selection

We select an approach type corresponding to each sub-task obtained.

The selection is based on five characteristics: training, compressible representation, generalization, inference mechanism and numeric data. In order to select the approach type we need:

- *Obtaining the problem desirable characteristics for each task*. Each task has a problem to solve and a set of characteristics desirables to find its solution.
- *Selecting the starting approach type of the characteristics selected.* If we need training, generalization or if we have numeric data, then we need a numeric approach. If we need a comprehensible representation or an inference mechanism, then the approach is symbolic. If we need characteristics of both approaches then we select a neuro-symbolic approach.

2.4 Technique base building

The main goal is offer a set of techniques available during the technique selection phase.

- *Obtaining a set of techniques*. The base can be integrated by the techniques developed by the computer vision expert or some commercial software.
- *Classifying the techniques according to the problem type [14]*. We classify each technique in some category as it is shown in Table 1.
- *Describing the abilities of each technique.* In the same form than the human being, the techniques have some specific abilities that help us to solve one problem better than others.

Table 1. Technique classification for different problem types [14].

Problem type	***Description***
Diagnosis	Inferring malfunctions of an object from its behaviour and recommending solutions.
Classification	Assigning an object to one of the defined classes.
Selection	Recommending the best option from a list of possible alternatives.
Prediction	Predicting the future behaviour of an object from its behaviour in the past.
Clustering	Dividing a heterogeneous group of objects into homogeneous subgroups.
Optimisation	Improving the quality of solutions until an optimal solution is found.
Control	Governing the behaviour of an object to meet specified requirements in real-time.
Data concise presentation	Producing compact description for a set of data. It can be numeric, graphic or "if-then" rules.

- *Assigning the approach type to each technique.* It is necessary to assign the approach type to each technique in order to facility its selection. The technique base is very important in the methodology because offer a set of alternatives available during the technique selection phase.

2.5 Technique selection

The goal is to select a set of alternatives available in the technique base to solve each task.

- *Obtaining as input the approach type.* It is necessary to know the approach type of the task that is necessary to solve.
- *Obtaining the alternatives available according to the approach*. We need to obtain the set of techniques contained on the approach requested.
- *Analyzing and selecting the technique according to some criteria*. First, we need to locate the task into the computer vision process in order to delimit the number of alternatives that can be used during the corresponding stage (preprocessing, segmentation, etc.). Second, we need to obtain the techniques that allow us to solve the problem identified. Third, if we do not have the technique, we need to verify the abilities of the techniques to solve the problem. We will obtain the best alternative to solve the task. At the final, we know the approach and the technique necessary to solve the particular problem.

2.6 Strategies selection

This stage consists on selecting the integration level between the connectionist and symbolic module.

The NSHS methodology proposed allows finding a "suitable" solution (not necessary "exact"), according to mental model theory. The integration strategies were based on [7]. The criteria and its corresponding classification are shown in Table 2 [7]. We not know *what is the hybrid architecture type* necessary to obtain a solution.

Table 2. NSHS classification.

Criteria	***Classification***
Integration type	Unified approach, Semi-hybrid, Hybrid.
Integration Mode	Chain, Sub treatment, Meta treatment, Co- treatment.
Coupling Grade	Weak couple, Middle couple, Strong couple.
Knowledge Transference	S→C, S←C, S↔C.

2.6.1 Integration type

The integration type is classified according to the hybridation type: a) *Unified approach*, attempts to integrate the symbolic systems properties into connectionist systems and vice versa, b) *Semi hybrid approach,* is used to achieve translations and c) *Hybrid approach,* may exist many symbolic and connectionist modules integrated to each other.

The process to select the integration type is:

- *Knowing the problem modules*. It is necessary to know the modules and the approach that we will use in each module respectively (symbolic/connectionist).
- *Identify the hierarchy among modules.* We need to know the priority of the tasks. *If* the task *n+1* depends of the output *n, Then,* do the task *n. Repeat until* the goal is obtained.
- *Identify the functions*. We can have three main functions. The *knowledge insertion* use a set of rules as initial structure of an ANN, also called knowledge compilation, the *knowledge refinement* where the knowledge is improved trough a learning algorithm, and the *knowledge extraction* that allows extract rules from an ANN.
- *Deduce the integration type.* The integration type depends on the problem necessities:
  If we have a symbolic module that needs some properties of the connectionist approach, or if we have a connectionist module that needs some properties of the symbolic approach, *then* we have a unified integration.
  If the problem needs to make some function and we have a connectionist module and a symbolic knowledge base, or if we have a symbolic module and a numeric knowledge base *then* we have semi-hybrid integration.
  If we have two or more modules and the modules perform some neuro-symbolic function *then* we have hybrid integration. If we have a lot of modules the integration cost will be higher as it is shown in figure 4.

  If number of modules ↑ ***then*** integration cost ↑.

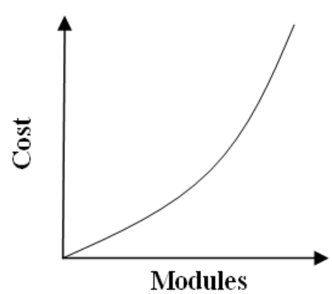

Fig. 4. Correspondence between number of modules and integration cost.

2.6.2 Integration Mode

Represents the reason why the neural module and symbolic module are configured in relation of one to other and the full system.

Chain, it is when two modules operate in sequence. One is the main processor and it is assisted by another module, acting like pre or post processor. The relationship between modules is input/output.

Sub-treatment, in this integration mode, one module is subordinate of another module to achieve some function. The main module decides at what moment calls it and how it uses its output. *Meta-treatment*, it is when a module solves the problem and the other plays a meta-level role such as take the control or improving results and the *Co-treatment*, in this mode both modules are equals in the problem solution process.

The process to select the integration mode is:

- *Knowing the problem modules*. It is necessary to know the modules and the approaches that we will use in each module (symbolic/connectionist).
- *Locate the system type.* According to the integration mode classification there exist a correspondence between the system type and the architecture. *Sequential system*, this has the correspondence with the chain treatment architecture, the modules achieve in sequence and with order. In this system type a module achieves its function and finish before the next module. The data are passed from one to other directly [17]. *Embedded system* has a correspondence with the sub-treatment architecture. These are systems where there is a main module that has control on another module called subordinate. This system type performs call from one module to another. The subordinate module act like subroutines. *Control systems*, these systems have the correspondence with the meta-treatment architecture. These systems can solve problems like control, directional control, post operational control, etc. *Collaborative system*, this system has a correspondence with the co-treatment architecture; allows share the same knowledge and the same processing charge to solve the problem.
- *Assigning the role.* A module can assume some role inside the system. *Main module*: has the responsibility of take the final decisions. This module can have subordinate modules that need to control. *Subordinate module*: has the responsibility to achieve some action requested by the main module. *Control module*: has the responsibility of monitoring the results obtained by the main module. *Cooperative module*, shares the same responsibility with one or more modules.
- *Locate the modules in some integration mode.* The modules can be located in some architecture or specific configuration as is shown in Table 3.

Table 3. Integration mode selection.

Architecture Type	***System Type***	***Module Type***
Chain	*Sequential*	Main – subordinate →
Sub-treatment	*Embedded*	Main – subordinate →,↔
Meta-treatment	*Control*	Main – control
Co-treatment	*Collaborative*	Collaborative - Collaborative

According to [14] the architecture type will be influenced for the available computational recourses.

If complexity of architecture ↑ ***then*** recourses ↑

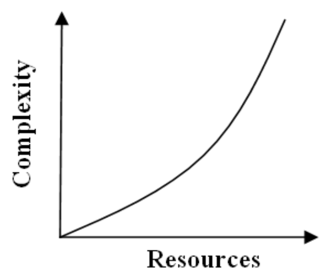

Fig. 5. Correspondence between the complexity of architecture and resources.

2.6.3 Coupling grade

Define the interaction force between two modules. The classification of different grades is made through a progressive level that runs from extreme to another.

This classification consists of three levels: *Weak couple*, the different modules are connected by a simple relation of input/output, and the communications are unidirectional, *Medium couple*, the interactions among modules are more flexible, due are bidirectional; it does not treat simply of input/output relationship, but rather each module can influence on the operation of another, *Strong couple*, the knowledge and data are shared among modules through internal structures in common. The process to select the coupling grade is:

- *Knowing the modules of the problem*. It is necessary to know the modules and the approach that we will use in each module (symbolic/connectionist).
- *Detect the communication lines among the modules*. It is necessary to know what is the dependence among modules, *if the output of the task A is the input of the task B, then there is a communication line (interconnection) in both tasks, where the grade of dependence will be defined according to the problem necessities.*
- *Identify the problem necessities.* There are some factors that have influence to decide the coupling grade, system total cost and performance, see figure 6.

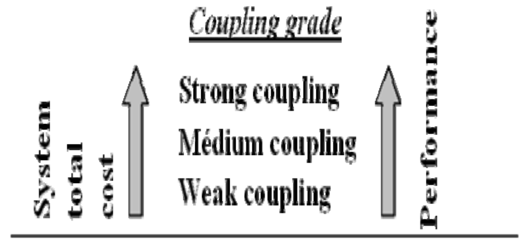

Fig. 6. Coupling grade according to system requirements.

If coupling grade ↑ ***then*** system total cost ↑
If coupling grade ↑ ***then*** better performance ↑

- *Identify the dependence type among the modules.* The dependence can be: *Data*, the modules are communicated through parameters. *Stamp*, the modules are communicated through a data structure. *Control*, it is the control grade of a module over other. *Extern*, it is when two modules depend on extern environment. *Common*, two modules have common coupling if they made the reference to the same data global area. *Contend*, when one module modifies to another.
- *Deciding the type and coupling grade.* In order to select the coupling types according to coupling grade is necessary considerer the classification shown in Table 4.

Table 4. Coupling types according to coupling grade.

Coupling grade	***Coupling type***
Weak coupling	Data coupling, Stamp coupling
Medium coupling	Control coupling, Extern coupling, Common coupling
Strong coupling	Contend coupling

2.6.4 Knowledge transference

The transference may be classified according to the exchange direction.

From symbolic to connectionist, the symbolic knowledge is transferred from a symbolic module and it is integrated to a connectionist module (S →C).

From connectionist to symbolic, the knowledge acquired by learning in connectionist net may be explained in a form of symbolic rules (S→C), Bilateral transfer, the knowledge can be transferred in both senses: symbolic and connectionist (S↔C).

Usually include compilation mechanism and rules extraction starting from the nets.

The process to select the transference is:

- *Knowing the system approach type*. It is necessary to know the approach types (symbolic/connectionist) due that not in all cases the symbolic module will has a module.
- *Knowing the modules of the problem*. Detect the modules that are going to compose the hybrid system. In the case of the semi-hybrid system only we have a symbolic or connectionist module.
- *Identify the knowledge transference lines*. The communication lines that indicate the knowledge transference form between modules.

  If the system has a connectionist module but there is a relationship between the symbolic approach and connectionist module, where the connectionist module depends of the initial symbolic knowledge, *then* the transference is unidirectional, from symbolic to connectionist.

  If the system has two modules connectionist and symbolic, but only one need the output of another and not in a reciprocal form, *then* the transference is unidirectional.

  If the system has a connectionist module but there is a relationship between the symbolic approach and the connectionist module and vice versa, *then* the transference is bidirectional.

  If the system has two or more modules connectionist and symbolic and both need its output in reciprocal form, *then* the transference is bidirectional.

2.7 Architecture type implementation

This phase consist on implement the system proposed by the NSHS methodology. The goal is obtain an alternative to find a solution in visual inspection problems or any problem where the Neuro-symbolic approach can give some solution.

2.8 Results evaluation

The goal is to analyze the results obtained with the NSHS methodology implemented. In order to do this, we considered the factors that have influence on the architecture type selection.

3 NSHS methodology implementation

All the objects in the real world have a surface that can reflect light in a way that depends on the structure of the surface. That way of reflecting light is known as the visual texture of the objects and gives information about the material that the object is composed of (wood, water, steel, wool, etc) and some properties (roughness, regularity, brightness, homogeneity, etc) that inform about the state (wet, clean, liquid, etc) of the object.

The quality inspection in artificial textures is a very important problem treated by [10]. That work, verifies the quality in artificial textures textiles based in several features. These characteristics allow classify the textures in good or bad quality. The results obtained with the changes contained in the images show the necessity to implement others techniques to solve the problem [10].

Phase 1: *Problem description*. The problem consists on the artificial textures recognition. The techniques used in [10] were effective; however, we hope that with the use of other techniques as neuro-symbolic approach, we will obtain a better solution in the recognition percentage.

The goal is to prove the NSHS methodology in order to obtain a better recognition results through the use of the expert knowledge and the computer vision system.

The problem is *located in the recognition stage*. This stage is the last stage in a lot of computer vision systems, see figure 7.

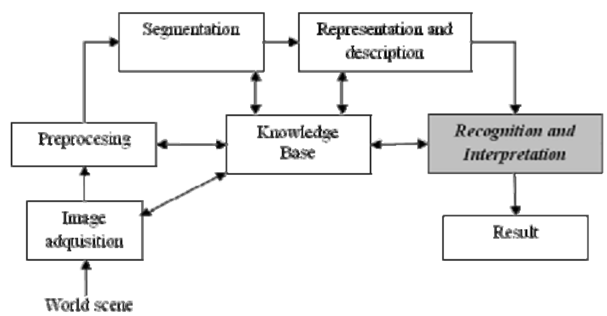

Fig. 7. Recognition stage in the computer vision process.

Phase 2: *Problem decomposition.* The problem was decomposed in two tasks (see figure 8).

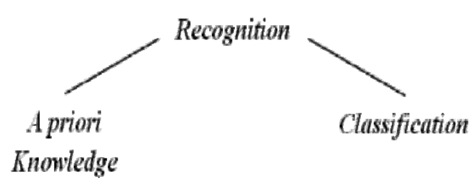

Fig. 8. Decomposition of the main task.

Task 1: A Priori knowledge. This task will use the human expert knowledge in order to explain when a texture is good or bad.

Task 2: Classification. This task will make the classification with tolerance to the image changes.

Phase 3: *Approach type selection.* We need to identify the problem desirable characteristics for each task. For the task 1, we need a compressive representation of the human expert criteria to decide when a texture is good or bad. For task 2, we need characteristics such as generalization that allows treating with numeric knowledge.

The task 1 needs a symbolic approach, while the task 2 needs a connectionist approach, in order to solve the recognition task the approach is neuro-symbolic.

Phase 4: *Technique base building*. In table 5 we present the results of implementing this phase.

Table 5. Characteristics of the available techniques.

Technique name	*Approach type*	*Ability*	*Classification*
Alvot	Numeric	Work with information lost	Classification
Testors	Numeric	It obtain the main characteristics	Selection
ANN	Numeric	Work with noise	Classification
Rules	Symbolic	Compressive knowledge representation	Classification, Data concise presentation

Phase 5: *Technique selection*. We have four available techniques. The Alvot technique was used in the previous work [10] and it has problems with the changes in the textures images, so, this is rejected. The rules are used to classify but it has difficulties with the generalization, then to solve the classification problem we select the ANN. The production rules are used to acquire the priori knowledge of the human expert.

Phase 6: *Strategies selection.* The results obtained are:

Integration type: There are two functions: knowledge insertion (compilation) and refinement, then the *system* is considered *hybrid* because there is knowledge integration, symbolic and numeric.

Integration mode: The problem has two modules, one has the main role and the other has the subordinate role and the system type is sequential, then the *architecture type* is *Chain.*

Coupling grade: We have a unidirectional communication line; the output of the symbolic module is the input of the numeric (connectionist) module. The coupling is a data type, because the rules are compiled to the ANN without be the rules modified, then the *Coupling grade* is *Weak.*

Knowledge transference: In this problem we have two modules: symbolic and numeric. This is a hybrid system that allows only the *unidirectional knowledge transference.* We need to make the transference from the symbolic module to the connectionist module.

Phase 7. *Architecture type implementation.* We implemented a hybrid system, with Chain architecture, a Weak coupling and unidirectional knowledge transference.

Phase 8: *Results evaluation*. The results obtained are shown in section 4.

4 Test and Results

For the tests we use a set of 510 textures for the learning stage and 600 textures for the recognition. The textures were selected randomly from [15], [16], [17] and modified to prove the *recognition capacity*. The figure 10 shows an example of the different textures used.

The textures selected were considered as the main pattern to build a piece of textile. We simulate changes in the color for the texture repetition.

The quality is based mainly in the conservation of colors of the texture images. From the knowledge of the experts we consider a good quality texture if we have a variation of colors between 1 and 10 %, all other textures out of that range are considered as bad quality textures.

Figure 9 shows an example of bad quality and god quality textures.

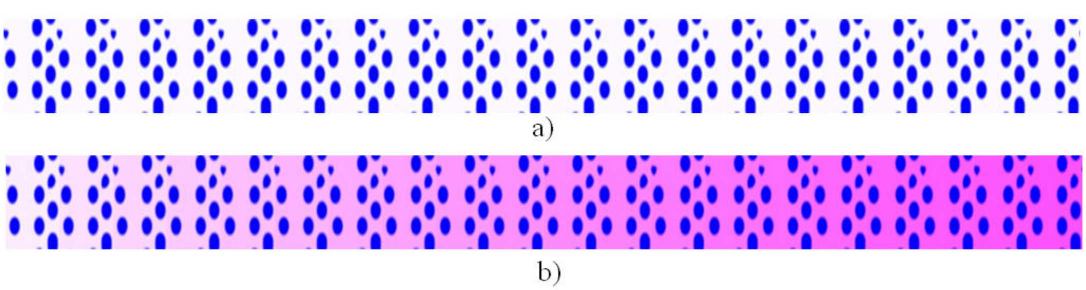

Fig. 9. Textiles, a) Good quality and b) bad quality.

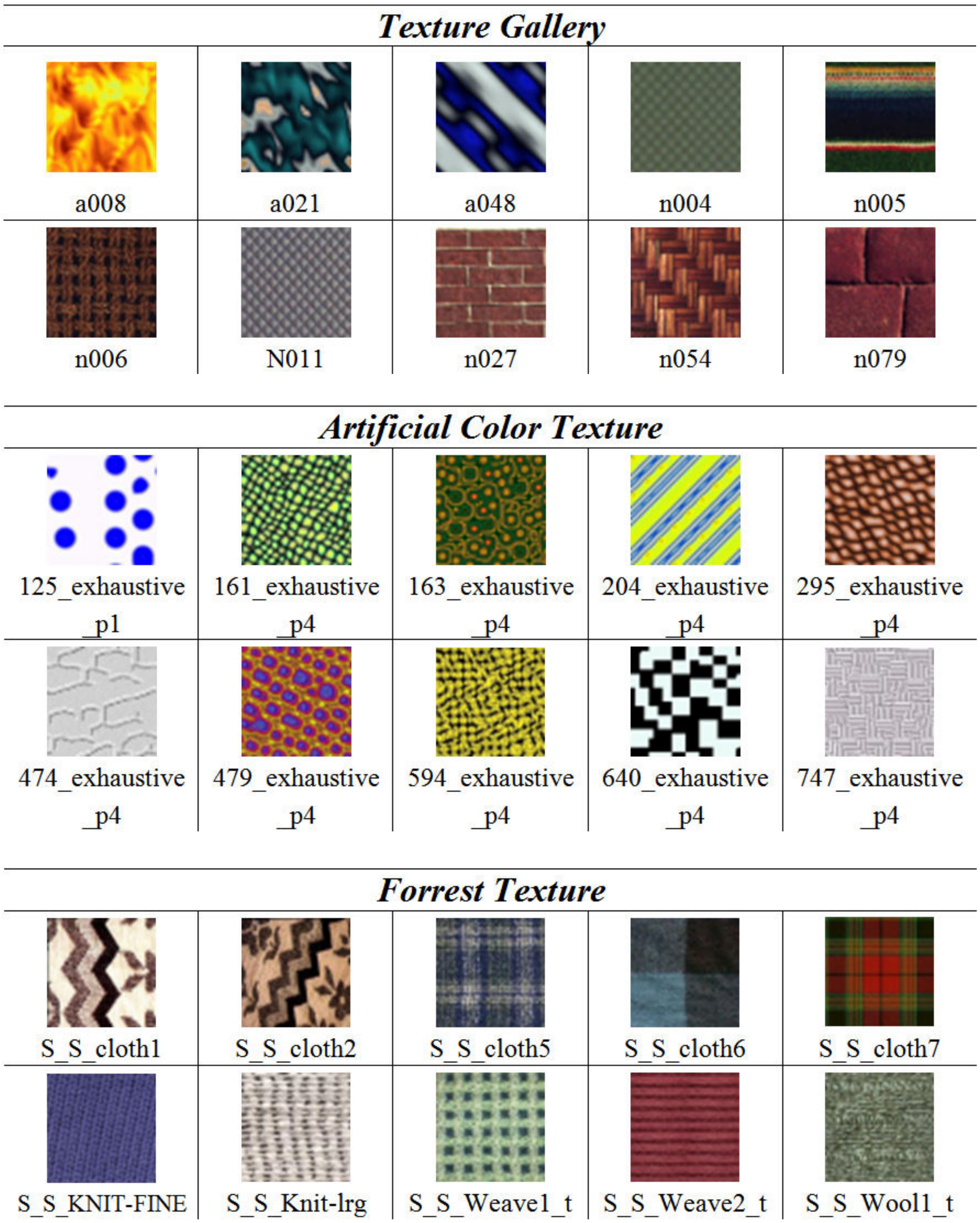

Fig. 10 Examples of the textures used.

Seven different kinds of tests were carried out under factors as: texture images rotated 90 and 180 degrees, scale, texture images halved and doubled, scale and rotation, texture images scaled and rotated, texture images with salt and pepper noise added, and finally, texture images with uniform noise added.

Each texture has five features: {*x1(average in red), x2(average in green), x3(average in blue), x4 (variance in green), x5(average in illumination)*}, obtained through the *typical testors* during the feature selection stage [10].

The *Alvot technique* was used to perform the recognition stage. Meanwhile the NSHS methodology use the numeric textures base in order to perform the learning and test with the neural network compiled with the symbolic knowledge of human expert. The compilation was done with *Neucomp* and the integration with *Neusim* [7]. The rule used for the knowledge compilation was:

If (*x1* has a range of (20, 246) and *x2* has a range of (20, 217) and *x3* has a range of (4, 253) and *x4* has a range of (72.2796, 10792.5) and *x5* has a range of (35, 225)) ***then*** the texture is of good quality.

The ranges were obtained by the expert through an analysis of the data. The rule was compiled obtaining an ANN with five inputs, one output, two layers and eleven units. The knowledge refinement was made using the Cascade Correlation (CasCor) algorithm in order to obtain the incremental learning [7].

Table 6 shows the results obtained with the *NSHS methodology* and a comparison with the results obtained in [10].

From the results obtained we can give several comments. First, using the methodology proposed we can improve the results about 20 %, compared with the original results. Second, the combination offers the possibility of understanding the problem solution in a better way. Third, by the analysis of column three and four of Table 6 we can see that the results obtained are better than those obtained with the computer vision system and the *Alvot technique*.

5 Conclusions

In this paper we have presented a *Neuro-Symbolic Hybrid System (NSHS) methodology* in order to find an alternative to solve several problems where the integration of two approaches as ANN and Production rules are necessary.

We implement the NSHS methodology in the texture quality inspection. The results show the capacity of the Neuro-symbolic approach to solve this problem in a better way.

The methodology proposed can be used to solve quality inspection problems, where the expert knowledge can contribute and complement the numeric knowledge of an artificial vision system.

Table 6. NSHS methodology results.

Class	*Number of images*	*NSHS methodology % recognition*	*CVS [10] %recognition*
Rotated 90°	30 30 30	96.66% 100% 100%	76.66% 56.66% 86.66%
Rotated 180°	30 30 30	96.66% 100% 100%	76.66% 56.66% 86.66%
Halved	30 30 30	96.66% 93.33% 100%	73.33% 56.66% 86.66%
Doubled	30 30 30	86.66% 96.66% 90%	70% 56.66% 86.66%
Scaled and rotated	30 rotated 90° and halved 30 rotated 90° and doubled 30 rotated 180° and halved 30 rotated 180° and doubled	100% 90% 100% 90%	76.66% 76.66% 80% 76.66%
Salt and pepper noise	30 30	100% 100%	90% 100%
Uniform noise	30 30	100% 96.67%	96.66% 100%

References:

[1] Wiriyacoonkasem S. and Esterline A. C., Adaptive Learning Expert Systems, *Proceedings of the IEEE SoutheastCon*, pp. 445 - 448, April 2000.

[2] Marcos N. and Azcarraga A., Successive Rule Refinement in a Hybrid Intelligent System, *WSEAS Transactions on Systems*, vol. 3, no. 2, pp. 396-401, April 2004.

[3] Wermter S. and Sun R., The Present and the Future of Hybrid Neural Symbolic Systems: Some Reflections from the NIPS Workshop, *AI Magazine*, vol. 22 no. 1, pp.123 – 126, 2001.

[4] Pacheco R. C., A hybrid Intelligent System for Diagnosing and Solving Financial Problems, *Ph.D. dissertation*, Santa Catarina Federal University, Brasil, March 1996.

[5] Sun R., *Artificial intelligence: Connectionist and symbolic approaches*, International Encyclopedia of the Social and Behavioral Sciences, Pergamon/Elsevier, Oxford. pp. 783 - 789, 2001.

[6] Hesham A., A novel Framework for Hybrid Intelligent Systems, *Proceedings of the 12th international conference on Industrial and engineering applications of artificial intelligence and expert systems: multiple approaches to intelligent systems*, Springer - Verlag, pp. 761 – 770, 1999.

[7] Osorio F. S., An Hybrid System for Constructive Machine Learning, *Ph. D. dissertation*, LEIBNIZ - IMAG –INPG Laboratory, France, 1998.

[8] Amirhassan Monadjemi S. and Moallem P., DWHT: A Fast Multi-Scale/Multi-Directional Texture Feature Extraction Scheme, *WSEAS Transactions on Computers*, vol. 5, no. 9, pp. 2106-2113, September 2006.

[9] Suzuri Hitam M., Haq Muslan M., Mat Deris M. and Mohd Saman M., Image Texture Classification using Gray Level Co-occurrence Matrix and Neural Network, *WSEAS Transactions on Computers*, vol. 3, no. 4, pp. 1151-1156, October 2004.

[10] Vergara Villegas O. O., Application of Visual Inspection for Artificial Textures Recognition, *Master dissertation*, Cenidet, Mexico, April 2003.

[11] González R. C. and Woods R., *Digital Image Processing*, Addisson – Wesley Iberoamerican, 1996.

[12] Peccardi M. and Cucchiara R., Bariani M. and Mello P. Exploiting Symbolic Learning in Visual Inspection, *Lecture Notes In Computer Science*, vol. 1280, pp.223 – 234, August 1997.

[13] Cruz Sanchez V. G., Reyes Salgado G., Vergara Villegas O. O. and Pinto Elias R., A Combined Representation to Refine the Knowledge Using a Neuro-Symbolic Hybrid System Applied in a Problem of Apple Classification, *Proceedings of the 16th International Conference on Electronics, Communications and Computers (CONIELECOMP)*, February 2006.

[14] Negnevitsky M., *Artificial Intelligence*, Addison Wesley, 2002.

[15] Jeremy B., Artificial Color Textures, Massachussets Institute of Technology (MIT), [on line]: http://graphics.stanford.edu/projects/texture/demo/synthesis_debonet.html, 2008.

[16] Forrest, Forrest Textures Forrest, [on line]: http://textures.forrest.cz/, 2008.

[17] The Gallery, Texture gallery, [on line]: http://www.clone.demon.nl/web/wg_index.htm, 2008.