SURFACE QUALITY ANALYSIS WITH SOFT COMPUTING

Andreas Eichhorn
BMW Group
Munich, Germany
Andreas.Eichhorn@bmw.de
Daniela Girimonte
Polytechnic of Bari, Electrotechnology and Electronic Department
Bari, Italy
girimonte@deemail.poliba.it
Aljoscha Klose, Rudolf Kruse
University of Magdeburg, School of Computer Science
Magdeburg, Germany
aljoscha.klose@cs.uni-magdeburg.de, rudolf.kruse@cs.uni-magdeburg.de,

Abstract:
Today the method for surface quality analysis of exterior car body panels is still characterized by manual detection of local form deviations and evaluation by experts. The new approach presented in this paper is based on 3-D image processing. A major step in this process is the classification of the different kinds of surface form deviations. For this purpose, we compared the performance of different soft computing techniques. Although the dataset was rather small, high dimensional and unbalanced, we achieved promising results with regard to classification accuracies and interpretability of rule bases.

1 Introduction

The quality standard of today's automotive industry products is very high. Especially car manufacturers of the upper-class and premium market segments differentiate their products from their competitors among other things by a perfect appearance of the painted car body. This is an important quality demand, as the outer panels are rather exposed and directly visible to the customer. In general, the impression of a car is determined by an appealing design of its body, the color and gloss of its paint, and the manufacturing and assembly accuracy of the exterior body panels.

The geometric complexity of these panels makes them difficult to produce with metal forming technologies. Small surface form deviations like sink marks always exist. Typical imperfections that are considered as distortions deviate in normal direction by tens of microns. The surface paint does not cover such imperfections. They result in inhomogeneous runs of light fringes on the highly reflective paint, which visibly disturb the perfect appearance of the car body.

The manufacturing process is optimized in order to eliminate or at least to minimize such surface defects at the end of the product development process. The position and the kind of the remaining surface form deviations on each outer panel are documented in a surface quality protocol and physically in a so called master piece. By definition the master piece represents the just acceptable geometric shape of each local form deviation. This high quality level has to be kept after the start of the series production. Therefore it is imperative to control the quality of the parts directly in the first steps of the manufacturing process in the press shop.

The current surface quality control procedure in the press shop is still done manually. Former studies about quantitative detection methods have not resulted in satisfying quality control systems. Today, during series production an experienced worker checks the produced parts at the end of the press line in constant intervals by treating their exterior surfaces with a grind-

Class	Linguistic Description
buldge	***rounded*** damage outward, ***distinctive*** feature, ***relatively small*** radius
sink mark	***slight*** flat based depression inward
press mark/ impact mark	***local*** smoothing of (micro-)surface, ***heavier*** sink mark, ***deep*** depression proceeded by a ***low*** peak
dent	***rounded*** damage inward, ***distinctive*** feature
flat area	***flat*** plane on curved camber surface
uneven surface	***several*** sink marks in series or adjoined
waviness	***several heavier*** wrinklings in series
line	***distinctive*** visible line
draw line	visible line caused by contact with tool
uneven radius	***visible*** distortion of radius geometry

Table 1: Surface form deviations

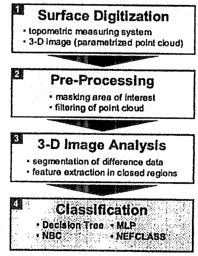

Fig. 1: 3-D image processing

stone. From the resulting specific patterns of local grinding marks he is able to detect form deviations, and derive their type and acceptance.

The experts introduced a list of surface defects and characterizations, to that they conform more or less in their daily quality work. The surface form deviations are characterized by linguistic descriptions of their specific appearance, as shown in Table 1 for some common defects. The geometry of the defects is specified by vague terms and attributes.

However, the current method has several disadvantages. It is cumbersome, subjective, error-prone and time consuming, especially when analyzing the surface of large parts totally. Moreover the assessed parts are often lost for the manufacturing process. Therefore it would be desirable to have a more objective, non-contact, faster and automatic estimation method.

Our approach, which is currently in development, is based on the digitization of the exterior body panel surface with an optical measuring system. From the resulting point cloud we try to characterize the form deviations by mathematical properties that are close to the subjective properties that the experts used in their linguistic descriptions. The approach has two major aspects: the quality specialists need information about the type of defect detected, and additionally they are interested in its severeness. In this paper we focus on the first aspect.

The characteristics of the described problem – its uncertainty, fuzziness and the use of expert knowledge – point to possible solutions in the field of soft-computing. Therefore, we compare the performance of different soft-computing techniques to determine the type of a defect from the extracted features.

2 Data acquisition and processing

Following the well known digital image processing chain (e.g. [4]), we try to implement a continuous 3-D image processing. Fig. 1 provides a simplified overview of the process, including digitization, image pre-processing and image analysis, and the application of soft-computing techniques for the classification of surface defects.

The digitization of the exterior body panels surface with a topometric 3-D measuring system is the basic step of our approach. The optical metrology offers high accuracy and resolution in a large sized measurement volume as well as fast and non-contact data acquisition. The operating principle of the sensor is called Miniaturized Projection Technique (MPT) and is based on a combined Gray code/phase shift technique [2]. Therefore, the MPT sensor projects a sequence of gratings onto the surface of the object to be measured. Each grating is digitized with a high resolution CCD camera under a defined angle. The superposition of the single images of one sequence enables a unique correlation between every pixel on the CCD chip and the position of each fringe in the projection plane, so that the depth information can be obtained by triangulation. The resolution limit in z-direction is about 5μm and the noise in z-direction has a value of ±10μm. The raw data is filtered in order to delete outliers and to re-

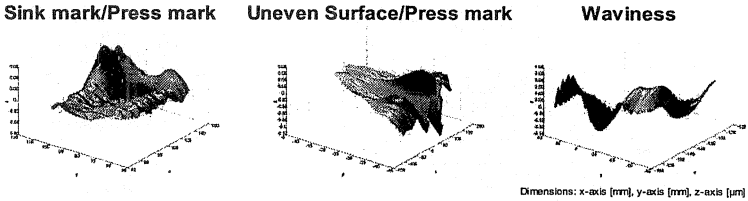

Fig. 2: 3-D visualization of local surface defect examples

duce the noise to a minimum. The outcome of this operation is an accurate 3-D point cloud, which contains the required geometric information of the surface defects.

From this point cloud, the ideal geometric shape of the part is approximated by a rather inertial surface of low polynomial degree. The local form deviations can then be determined as the differences between the 3-D point cloud and the approximated surface. 3-D plots of typical examples for local surface form deviations are illustrated in Fig. 2.

With respect to the linguistic description of the different defect classes it is not obvious, which mathematical characteristics permit an efficient classification process. For this reason a system of geometric attributes was developed. The goal was to define features that are in a close connection with the linguistic descriptions. In all, 42 attributes have been defined that are the basis of the further analysis.

3 Data characteristics

Currently, the handling of the 3-D measurement system and the data processing itself requires a considerable amount of manual interaction due to its prototypical stage. We were thus forced to restrict our analyses to a small, but hopefully representative set of selected master pieces. Concretely, the basis of our analyses are 9 master pieces with a total number of 99 defects recorded by the experts in the corresponding quality protocols. From those protocols, we know the position and type of the defects, however their severeness is not clear. For each of these defects the complete set of 42 features was calculated.

Fig. 3 shows the distribution of defect types in our database. Obviously, the types are rather unbalanced, and the less frequent types occur very rarely. Defect type *uneven radius* was even observed only once – one can hardly expect to learn a somehow generalizing classifier for a single pattern. The common approaches for handling unbalanced class frequencies do not seem promising in our case: reducing the more frequent types would decrease our already small database, and duplicating the rare cases would not increase their variance needed to learn a well generalizing classifier. Therefore, we discarded defect types with less than four occurrences. Thus, 94 examples of classes *draw line, flat area, sink mark, press mark and uneven surface* are left.

In contrast to the low number of examples, the number of features is extremely high. High dimensionality is a general problem in data analysis, and not all of the classifiers used in this study are equally suited to learn from high dimensional data. Therefore we performed an explicit feature selection [3]. First of all, we found that some of the features were almost identical, i.e. their linear correla-

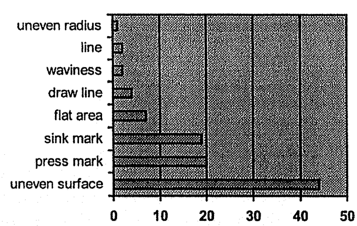

Fig. 3: Occurrences of defect types

tion is very close to 1. As the vertical extension of defects is orders of magnitude smaller than their size, features calculated on the 3-D shape of the defects are very similar to those calculated on their 2-D projections. We therefore discarded some of the extremely high correlated features. We then ranked the remaining 31 features by importance using forward-sequential feature selection. We estimated the error by 1-nearest-neighbor-classification on the normalized features with 1-leave-out.

For the experiments we used 4-fold cross validation [3]. Therefore, the database was split into four parts using stratified sampling to ensure that every split contains a similar distribution of defect types. Especially, this procedure ensures that each part contains at least one instance of each class.

4 Classifying Defect Types

We compared four different approaches to classification. The first three methods are commonly used classification techniques, namely Naïve Bayes classification, decision trees, and multi-layer perceptrons. The fourth is NEFCLASS, the well-known hybrid neuro-fuzzy classifier developed at the University of Magdeburg. The following paragraphs will briefly outline the concepts of NEFCLASS.

4.1 NEFCLASS: A Hybrid Neuro-Fuzzy Classifier

Although neural networks are popular data mining methods, the "learnt" knowledge is stored in the numeric network connections, and thus they do not provide human understandable information about the data. A remedy lies in the combination of neural networks with fuzzy systems: we use a fuzzy system to represent knowledge in an interpretable manner, and use the learning ability of neural networks to determine membership values. The drawbacks of both of the individual approaches – the black box behavior common to neural networks, and the problem of finding suitable membership values for fuzzy systems – can thus be avoided.

NEFCLASS is such a hybrid approach [8]. Its structure is a three layer feed-forward network with coupled fuzzy weights. The network can be interpreted as fuzzy if-then rules of the form

$$R_r : \text{if } x_1 \text{ is } A_r^{(1)} \text{ and } \ldots \text{ and } x_n \text{ is } A_r^{(n)} \text{ then } x \text{ is } c_r, \qquad (1)$$

where $A_1^{(i)}, \ldots, A_m^{(i)}$ are linguistic terms (like *small, medium* or *large).* They are represented by fuzzy sets $\mu_1^{(i)}, \ldots, \mu_m^{(i)}$, that build a fuzzy partition of the *i*-th dimension. The patterns are vectors $\boldsymbol{x} = (x_1, \ldots, x_n)$ that belong to k disjunct classes c_i.

The network structure – i.e. the set of rules – is created by the procedure suggested by Wang and Mendel [13]. The initial fuzzy partitions structure the data space as a multidimensional fuzzy grid. The rule base is created by selecting those grid cells that contain data. This can be efficiently done in a single pass through the training data.

After a rule base has been generated from an initial fuzzy partitioning, the membership functions must usually be fine-tuned in order to improve the performance. In the NEFCLASS model, the fuzzy sets are modified by simple backpropagation-like heuristics, motivated by neural network learning. In the learning phase, constraints are used to ensure that the fuzzy sets still fit their associated linguistic terms after learning. For example, membership functions of adjacent linguistic terms must not change position, and must overlap to a certain degree [8].

The NEFCLASS model has been continuously improved and extended over the last few years. Most of these extensions address the specific characteristics and problems of real world data and their analysis, like for example symbolic attributes [9], missing values [10], or learning with asymmetric error semantics [5]. Another import extension is the integration of pruning techniques. When a rule base is induced from data it often has too many rules to be easily readable, and thus gives little insight into the structure of the data. Therefore, to reduce the rule base, several pruning techniques have been presented for NEFCLASS. These methods

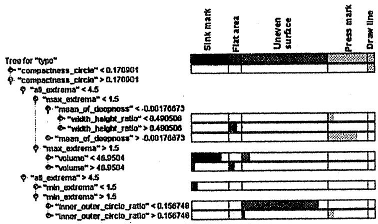

Fig. 4: A learnt decision tree

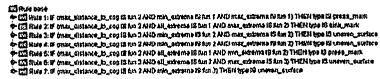

Fig. 5: The results for NEFCLASS

are effective in both reducing the number of rules and increasing generalization ability, and are of great importance for practical applications with higher numbers of dimensions. Details can be found in [10][6].

5 Application and Results

This section describes the application of the selected classifiers to our database. For each of them we tried to find a set of parameters that perform well on specific training data and are still general enough that they can be applied to other data. We therefore trained the classifiers with fixed settings to all four training datasets and applied the results to the corresponding test datasets. We will describe settings, classifiers peculiarities, and steps to improve the classification.

To measure the performance of the classifiers we present classification accuracies on training and test data (Table 3) and the confusion matrices on the test data (Table 2). The accuracies, i.e. the averaged relative and the absolute number of misclassification over the four datasets, give us an idea how well the classifier performed in general. The differences of accuracy on learning data and validation data shows how well the classifier generalizes on unseen data. The confusion matrices allow a detailed view into the classification. The entries on the main diagonal are the correctly classified patterns. The remaining entries show, how many patterns of a class have been wrongly classified as some other class.

Although we tried an abundance of different parameter settings in countless classifier runs, we only report the results that we consider to be optimal.

5.1 Naïve Bayes

The Naïve Bayes classifier in its basic form has no learning parameters. However, one can often improve its performance by selecting an optimal subset of features [7]. This selection is carried out as follows. We start with no attributes at all. Then we add attributes one by one. In each step we select the attribute which, if added, leads to the smallest number of misclassification on the training data. We stop adding attributes when adding any of the remaining attributes does not reduce the number of errors.

The final classification accuracy is 89.0% on the training and 75.6% on the test data.

5.2 Decision Trees

For the induction of the decision trees we tried several attribute selection measures, as described in [1]. Most of the measures yield reasonable results. However, the *Symmetric Specificity Gain Ratio* maximized the tree accuracy over the training data set so we employed it as split criteria. For the pruning we use confidence level pruning [11] with a confidence of 50%. The classification accuracy is 94.7% on the training and 75.6% on the test data. An exemplary tree is shown in Fig. 4.

5.3 Neural Networks

We found that standard back-propagation learning did not perform well on our datasets. We got the best results with *resilient back-propagation* (RPROP) learning, that performs a local adaptation of the weight-updates according to the behavior of the error function [12].
Because of the extremely high number of dimensions the network is extremely prone to overfitting, and some form of pruning is imperative. For the pruning we created a network that was initially larger than necessary, and then stepwise pruned out the weakest neurons. As we pruned not only neurons in the hidden, but also in the input layer, we were able to reduce the number of inputs the network uses. Furthermore to improve classification we use weight decay [14], which is a regularization method that penalizes large weights by adding a penalty term to the error function.
The best network had one hidden layer of 7 units, 5 units in the output layer, and 5 units in the input layer (after learning). The classification accuracy is 90.0% on the training and 85.5% on the test data.

5.4 NEFCLASS

When we tried to train a classifier with NEFCLASS we encountered some problems due to the high dimensionality of the dataset. In such cases, the structure-oriented approach by Wang and Mendel tends to produce too many, too specialized rules. Fuzzy set optimization gets unstable on such neuro-fuzzy networks, and as the pruning methods rely on an initial rule base, they might fail too.
We therefore used a subset of 7 attributes as chosen by the forward-sequential feature selection (s. Sect. 3). This made it easier to find good and general parameter settings for NEFCLASS.
The best classification accuracy was 81.6% in average on the training sets and 79.9% on the test sets. The fuzzy sets after learning and pruning, and the rules are shown in Fig. 5.

6 Discussion

Table 3 summarizes the classification accuracies for all classifiers. The last column ("DC") shows the accuracy for the default classifier, which always predicts the majority class *uneven surface*.

Table 2: The confusion matrices for the four classifiers

Naïve Bayes	1)Sink mark	2)Flat area	3)Uneven surf.	4)Press mark	5)Draw line
1) Sink mark	13	-	3	2	1
2) Flat area	3	-	2	2	-
3) Uneven surface	3	-	41	-	-
4) Press mark	1	1	-	17	1
5) Draw line	1	-	1	2	-
Neural networks					
1) Sink mark	15	-	4	-	-
2) Flat area	2	1	-	4	-
3) Uneven surface	2	-	42	-	-
4) Press mark	-	1	-	19	-
5) Draw line	-	-	1	-	3

Decision trees	1)Sink mark	2)Flat area	3)Uneven surf.	4)Press mark	5)Draw line
1) Sink mark	14	1	4	-	-
2) Flat area	1	3	1	2	-
3) Uneven surface	4	-	36	1	3
4) Press mark	1	1	1	17	-
5) Draw line	1	-	1	1	1
NEFCLASS					
1) Sink mark	18	0	1	0	0
2) Flat area	2	0	1	4	0
3) Uneven surface	6	0	38	0	0
4) Press mark	1	0	0	19	0
5) Draw line	1	0	1	2	0

Table 3: Classification Accuracy on the Training and Test Cases

	NBC	Dtrees	NN	NEFCLASS	DC
Train Set	89.0%	94.7%	90%	81.6%	46.8%
Test Set	75.6%	75.6%	85.5%	79.9%	46.8%

We expected the classification to be rather difficult: We have a low number of examples, with many dimensions and highly unbalanced class frequencies. However, although the results are not perfect, they are all significantly better than the default classifier.

In unbalanced datasets minority classes may likely be ignored by a classifier. The second and fifth diagonal entries in the confusion matrices correspond to the number of correctly classified patterns for the minority classes *flat area* and *line*. Obviously all classifiers had difficulties with those classes, as the values in the matrices are rather small. Both, Naïve Bayes classifier and NEFCLASS misclassified all of these patterns. In the trained and pruned NEFCLASS classifier there is not even a rule for those classes.

Furthermore, looking at the confusion matrices tells us about the relations of the majority classes. From the higher number of confusions we can see, that *uneven surface* and *sink mark* seem to be rather overlapping, whereas *pressure mark* seems to be better separated from the other classes. These relations can also be found in the linguistic descriptions of the defects, where *uneven surface* and *sink mark* are rather similar.

It is hard to give a general recommendation, which of the classifiers is best suited for the problem at hand. If classification accuracy was the only goal, the feed-forward neural networks would be the method of choice with 86% correctly classified test patterns, followed by NEFCLASS with 80%. However, in responsible fields like quality control, especially when we try to predict an expert decision, confidence into the system is of extreme importance. Neural networks are not of much help in that aspect, as the knowledge is hidden in the network connection weights. Experts will most probably be more confident in a system, if its decisions are transparently and understandably given by rules or trees. Both rules and trees have their advantages. However, in our case the fuzzy rule base seems to be more suited to the problem, as – in contrast to the decision trees – almost no overfitting can be observed. Even the pruned decision trees performed much better on the training data than on the unseen test data. If we look at the number of rules and paths, respectively, and the number of used attributes, we find that the NEFCLASS rule bases are in average less complex than the decision trees.

Another criterion for classifier choice could be its computational simplicity. Naïve Bayes and decision trees have considerable advantages over the neuro- or neuro-fuzzy-techniques in their computational demands and the expertise and time needed to choose appropriate parameters. However, this criterion has little weight, as our application is not time critical.

Consequently, for the given application, NEFCLASS turns out to be the best compromise between accuracy and transparency.

It might however be interesting to train different classifiers and aggregate their information. The four presented classifiers agree in their prediction in 60 of the 94 patterns, and only 5 of those 60 classifications are wrong. For another 26 patterns, three out of four classifiers predict

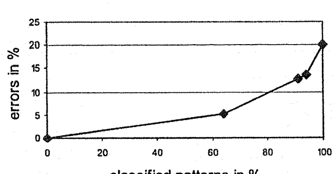

Fig. 6: The error for the ensemble classifier

the same, being wrong in 7 cases. The corresponding error curve is shown in Fig. 6. We can use such an ensemble of classifiers to get information about the reliability of the classification. On the other hand, we can take a closer look at patterns that are misclassified by all of the approaches. Looking at those outliers might help to detect inconsistencies in the dataset.

7 Conclusions and Outlook

The presented 3-D image processing approach from surface digitization to defect type classification yielded promising results. In our opinion, NEFCLASS offered the best compromise between accuracy of the results and transparency of the learnt knowledge.

Currently, we did not take into account all of the classes. In the next step of the project, we will generate a larger database, where also the rare classes occur more often. This might enable us to further improve the defect type prediction.

However, the qualitative analysis – the prediction of defect types – is only a first step. Our future work will be directed towards a more quantitative analysis, to tell how severe a form deviation is and what actions should thus be initiated.

References

[1] C. Borgelt, J. Gebhardt, and R. Kruse. *Concept for Probabilistic and Possibilistic Induction of Decision Trees on Real World Data.* Proc. 4th EUFIT'96, Vol 3:1556-1560. Verlag Mainz, Aachen, Germany, 1996.

[2] B. Breuckmann. *Topometric Sensors for Prototyping and Manufacturing.* Proc. SPIE, Vol. 2787, p. 2-11, 1996.

[3] D. J. Hand, H. Mannila, and P. Smyth. *Principles of Data Mining.* MIT Press, USA, 2001.

[4] B. Jähne. *Digital Image Processing.* 5th rev. ed., Springer Verlag, Berlin, 2001.

[5] A. Klose, D. Nauck, K. Schulz, and U. Thönnessen. *Learning a Neuro-Fuzzy Classifier from Unbalanced Data in a Machine Vision Domain.* In Proc. Fuzzy-Neuro Systems 1999 (FNS'99), pages 23-32. Leipziger Universitätsverlag, Leipzig, 1999.

[6] A. Klose, A. Nürnberger, and D. Nauck. *Some Approaches to Improve the Interpretability of Neuro-Fuzzy Classifiers.* In Proc. 6th EUFIT'98, pages 629-633. Aachen, 1998.

[7] P. Langley and S. Sage. *Induction of Selective Bayesian Classifiers.* In Proc. of 10th Conf. on Uncertainty in Artificial Intelligence UAI'94, pages 399-406. Morgan Kaufmann, San Mateo, CA, USA, 1994.

[8] D. Nauck, F. Klawonn, and R. Kruse. *Foundations of Neuro-Fuzzy Systems.* John Wiley & Sons Inc., New York, 1997.

[9] D. Nauck and R. Kruse. *Fuzzy Classification Rules Using Categorical and Metric Variables.* In Proc. Fuzzy-Neuro Systems 1999 (FNS'99), pages 133-144. Leipziger Universitätsverlag, Leipzig, 1999.

[10] D. Nauck, U. Nauck, and R. Kruse. *NEFCLASS for JAVA - New Learning Algorithms.* In Proc. 18th Intl. Conf. of the North American Fuzzy Information Processing Society (NAFIPS'99), pages 472-476. IEEE, New York, NY, 1999.

[11] J. R. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufmann, San Mateo, CA, USA, 1993.

[12] M. Riedmiller and H. Braun. *A Direct Adaptive Method for Faster Backpropagation Learning: the RPROP Algorithm.* In Proc. IEEE Intl. Conf. on Neural Networks, pages 586-591, San Francisco, CA, USA, 1993

[13] L. Wang and J. Mendel. *Generating Fuzzy Rules by Learning from Examples*, IEEE Trans. System, Man, and Cybernetics, 22(6):1414-1427, 1992.

[14] P. Werbos. *Backpropagation: Past and Future.* In Proc. IEEE Intl. Conf. on Neural Networks, pages 343-353. IEEE Press, 1988.