Mathware & Soft Computing 11 (2004) 163-177

Improving Surface Defect Detection for Quality Assessment of Car Body Panels

C. Döring+, A. Eichhorn*, D. Girimonte× and R. Kruse+

+ University of Magdeburg,
School of Computer Science Magdeburg, Germany
e-mail: {doering,rudolf.kruse}@cs.uni-magdeburg.de
* BMW Group Munich, Germany
e-mail: Andreas.Eichhorn@bmw.de
× Polytechnic of Bari,
Department of Electrotechnics and Electronics, Italy
e-mail: girimonte@deemail.poliba.it

Abstract

Surface quality analysis of exterior car body panels was still characterized by manual detection of local form deviations and subjective evaluation by experts. The approach presented in this paper is based on 3-D image processing. A major step towards automated quality control of produced panels is the classification of the different kinds of surface form deviations. In previous studies we compared the performance of different soft computing techniques for the detection of surface defect types. Although the dataset was rather small, high dimensional and unbalanced, we achieved promising results with regard to classification accuracies and interpretability of rule bases. In this paper we reconsider the collection of training examples and their assignment to defect types by the quality experts. For improving the reliability of the defect classification we try to minimize the uncertainty of the quality experts' subjective and error-prone labelling. We build refined and more accurate classification models on the basis of a preprocessed training set that is more consistent. Improvements in classification accuracy using a partially supervised learning strategy were achieved.

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

Class	Linguistic Description
bulge	*rounded* damage outward, distinctive feature, *relatively small* radius
sink mark	*slight* flat based depression inward
press mark	local *smoothing* of (micro-)surface, heavier sink mark, deep depression preceeded by a low peak
dent	*rounded* damage inward, distinctive feature
flat area	*flat* plane on curved cumber surface
uneven surface	*several* sink marks in series or adjoined
waviness	*several heavier* wrinklings in series
uneven radius	visible distortion of *radius* geometry

Table 1: Surface form deviations.

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.

Surface quality control in the press shop is currently a manual procedure. 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 grindstone. 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 manual procedure has several disadvantages. It is cumbersome, subjective, and thus error-prone. Furthermore it is time consuming, especially when analyzing the surface of large parts totally. The assessed parts are often lost

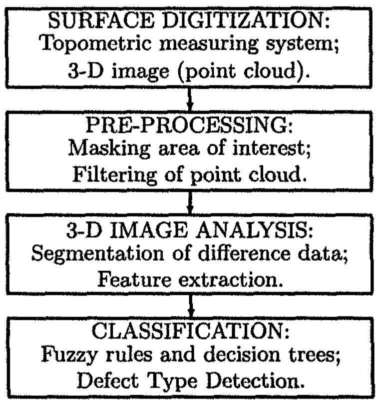

Figure 1: Automatic quality assessment based on 3-D image processing.

for the manufacturing process. Therefore it is 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 (Figure 1). 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. Our studies 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 studied the performance of different soft-computing techniques to determine the type of a defect from the extracted features. We achieved promising results with regard to classification accuracies and interpretability of rule bases [12]. In this paper we reconsider the collection of training data. Since results of previous experiments point out inconsistencies in the training data, we try to minimize the influence of the experts' subjective and error-prone labelling. In a partially supervised learning approach we try to resolve problems due to uncertain class assignments for further improving the reliability of the defect classification. We believe that refined and more accurate classification models can be built on the basis of more consistent training data.

2 Data Acquisition and Processing

Following the well known digital image processing chain (e.g. [1]), we try to implement a continuous 3-D image processing. Figure 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 $5\mu m$ and the noise in z-direction has a value of $\pm 10\mu m$. The raw data is filtered in order to delete outliers and to reduce 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. Condensed visualizations of 4 typical form deviations are shown in Figures 2-5.

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 features was developed. The goal was to define features that are in a close connection with the linguistic descriptions. The system is structured into four main categories:

- features derived from orthogonal projections,
- gradient analysis,
- features calculated directly on the numerical basis of the extracted difference data (e.g. volume or maximum depth of a defect), and
- ratios between other features.

In all, 58 features have been defined that are the basis of the further analysis.

3 Data Characteristics and Previous Results

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 19 master pieces with a total number of 273 defects recorded by the experts in the corresponding quality protocols. From those protocols, we know the position and type of the defects as they were determined by a quality expert. For each of these defects the complete set of 58 features was calculated. Figure 6 shows the frequencies of the defect types in the dataset. Obviously, the types are rather unbalanced, and the less frequent types occur very rarely. Defect type *uneven radius* and *flat area* were

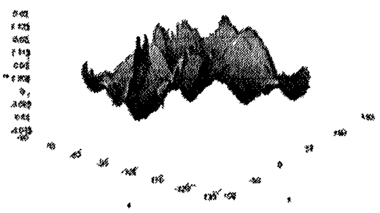

Figure 2: Uneven surface

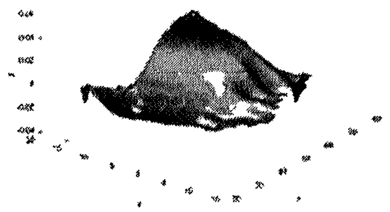

Figure 3: Press mark

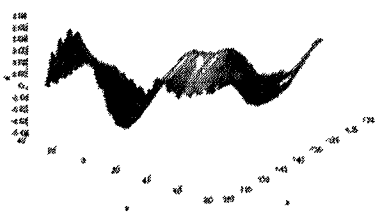

Figure 4: Waviness

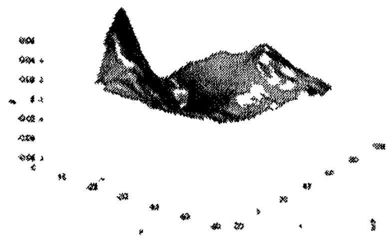

Figure 5: Sink mark

observed only less than ten times. Some examples were labelled to be of defect class *uneven surface/pressmark* in the dataset. This unexpected class name was introduced by quality experts and indicates a mix of two defect types. 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 our studies were equally suited to learn from high dimensional data.

We expected the classification to be rather difficult: we have a low number of examples, with many dimensions and highly unbalanced class frequencies. Regarding the classification accuracy previous work sounds promising: in stratified 4-fold cross validation decision trees correctly classified 82.42% of the test patterns [12]. The fuzzy rule bases which were induced from the described dataset were approximately 7% less accurate. The classifiers performed fairly, but not equally well in discriminating between the majority classes. However, reliable descriptions of the minority classes could not be obtained. Decision trees and the fuzzy rule based models misclassified many examples of class *flat area* and *uneven radius*. Both types could not be separated from the other classes. The pruned rule bases often did not contain rules for the small classes. The common approaches for handling unbalanced class frequencies also did not seem promising in our previous studies:

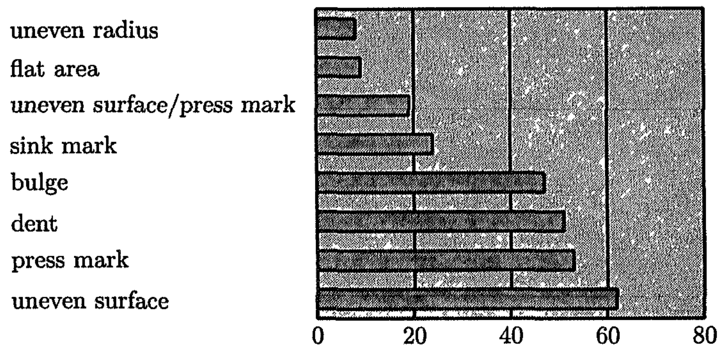

Figure 6: Occurrences of defect types

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. The small numbers of patterns of the types *flat area* and *uneven radius* are not representative enough to reliably detect those types.

Another difficulty for classification has been posed by the defect type *uneven surface/ press mark*. Most patterns of this type have been classified incorrectly as being either of class *uneven surface* or of class *press mark*. Apparently, it cannot be separated clearly as its own class. In the trained and pruned fuzzy rule bases there were no rules for this defect type. This indicates inconsistencies in the dataset and let us to question the way the data were collected and labelled.

4 Data Quality

In our application classification models are built based on training examples that were labelled by experts in a manual, subjective procedure. Thus, the expert decision on the defect type at hand is as error-prone as the manual surface quality control. In the most severe situation our classifiers are trained on basis of too many wrongly labelled or noisy examples. Then automated classification would jeopardize any quality control efforts. In the other case, fairly good classifiers can be generated with quite accurate training examples. Then the classifier decision can doubt an expert opinion in case of misclassified test examples. These viewpoints highlight the importance of consistent training data and the benefit of expert involvement in classifier design for arriving at reliable defect type detection.

The examples which were labelled as defect type *uneven surface/ press mark* by the expert could not be separated well by all classifiers in our previous experiments. Instead, they were classified as *press marks* or *uneven surfaces*. We can assume that the number of extracted features is sufficient to discriminate all classes. Thus, we suspect that these examples are not of there own separate class. Possibly due to the geometric characteristics of these defect examples which showed similarities to the shape properties of press marks or uneven surfaces (whiteout being clearly of either type), experts invented this mixture class to have a new 'bucket' to collect these defects. Consequently, we view these examples as noisy training data points with highly uncertain class labels in our following study. We do this in attempt to minimize the uncertainty introduced by the experts' labelling of examples.

The low number of minority cases which were not representative enough to allow detection of their types cases. This poses the second problem we face regarding the quality of our training dataset. For ensuring comparability with previous results, however, we do not omit these examples in the following study that focuses on eliminating inconsistencies among majority classes.

5 Classifying Defect Types

The simplest way to reduce the inconsistencies introduced by the class of mixed examples *uneven surface/ press mark* is to remove these defects from the dataset. This, however, would reduce the size of our already small set of training examples. Instead we only delete the class-label of these examples and use a partially supervised learning scheme. This way, we eliminate the uncertainty of this training data label without throwing away data, that has been collected in a time-consuming and expensive manner. The induction of classifiers in our following study is then supported by these unlabelled cases in the hope not to compromise the classification performance. We use a two step post-labelling approach [13]. In the first step we build a data model of the relevant data. Having obtained a cluster model, the labelled data is used for labelling the whole clusters by applying the majority principle. That is, the label of the cluster is assigned to the class, which is represented in cluster with the largest number of data points. All unlabelled samples are relabelled with the majority class name of the cluster they belong to. Given the entire newly labelled dataset we construct the final classifiers in the second, supervised learning phase. The first step of the approach should result in new class assignments for unlabelled examples based on other data points in the training set which are more representative of their true classes. In the second step we hope that better decision boundaries can be constructed than those of the previous classifiers, which were trained on examples with highly uncertain or wrong class labels.

For the clustering step and one of the final classifiers in this study we had to perform an explicit feature selection, due to the high dimensionality of the data [4]. First of all, we found that some of the features were almost identical, i.e. their linear correlation 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 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.

When selecting suitable classification methods for a system that detects surface faults we have to consider, that confidence into the system that predicts expert

cluster	1	2	3	4	Σ
pressmark	0	36	17	0	53
unevensurface	44	2	1	15	62
no label	14	5	0	0	19
Σ	58	43	18	15	134

Table 2: Result of the clustering step.

decisions is extremely important in responsible fields like quality control. The involved experts are more confident in a defect detection system, if its decisions are transparently and understandably given by rules or trees. Therefore we compared the following approaches for the final classification: decision trees [9], neuro-fuzzy classification with the well-known NEFCLASS, and a rather new fuzzy rule induction algorithm. For the experiments with the final classifiers we used 4-fold cross validation [4]. That is, 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.

After the description of the clustering step in this partially supervised learning approach, we give a brief outline of NEFCLASS and the constructive training algorithm for the induction of mixed fuzzy rules. Results of the final classification are presented in section 6.

5.1 Unsupervised Classification Step

The questionable class label *unevensurface/ pressmark* has been removed from the respective examples with the objective of finding more appropriate and reliable class re-assignments than the expert grouping into a mixture class. The relabelling with a cluster model is objective and purely based on similarity to data points which are hopefully more representative for the classes. Assuming that the true class of the unlabelled examples could not be judged between *unevensurface* and *pressmark* by the experts, we comprised a dataset with all examples of these two classes and those previously assigned to the mixture class. We clustered this set of data with the EM algorithm as implemented in the WEKA package [5]. We therefore used the subset of the 10 best features that resulted from forward-sequential feature selection. The cluster model with the best data likelihood we obtained has 4 clusters. The distribution of the samples over the clusters is shown in Table 2.

Labelling according to the majority principle yielded the following class assignments to the clusters: clusters 1 and 4 were labelled *unevensurface* while cluster 2 and 3 were assigned to class *pressmark*. 14 of the examples previously in the mixture class were relabelled with class *unevensurface*, since they belong to cluster 1. Five of the unlabelled examples in cluster 2 consequently got the class label *pressmark*. With this relabelling of the examples of the mixture class we obtained the modified dataset, which now only contains 7 classes. It was used in the supervised classification step with decision trees and the fuzzy rule classifiers that are described in the following.

5.2 NEFCLASS: a hybrid neuro-fuzzy classifier

NEFCLASS is a well-known hybrid neuro-fuzzy classifier developed at the University of Magdeburg. It has been designed to overcome the limited interpretability of neural networks. 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 [7]. 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:

$$R_r\text{: if } x_1 \text{ is } A_r^1 \text{ and ... and } x_n \text{ is } A_r^n \text{ then } \vec{x} \text{ is } c_r,$$

where A_r^1, ..., A_r^n are linguistic terms (like small, medium or large). They are represented by fuzzy sets μ_1^i, ..., μ_m^i , that build a fuzzy partition of the $i-th$ dimension. The patterns are vectors $\vec{x} = (x_1, ..., x_n)$ 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 [10]. 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 [7]. 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. An important 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 only 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 are effective in both reducing the number of rules and the number of features in the antecedences for improving generalization ability. Pruning is of great importance for practical applications with higher numbers of dimensions. Details can be found in [8][6].

5.3 Constructive Induction of Fuzzy Rule Bases

The creation of fuzzy rules by globally partitioning the data space into a multidimensional grid can be problematic. Then the extracted rules are constrained on all features in the dataset. This leads to the limitations of NEFCLASS and numerous other fuzzy rule induction algorithms: they often scale badly with higher dimensions of the feature space. Extracted rule bases are hard to interpret although they are meant to yield understandable descriptions. Thus, feature selection beforehand and extensive pruning of initial rule bases are required. An alternative approach presented in [11], called mixed fuzzy rule formation, tries to avoid these problems with a sequential, constructive algorithm. The fuzzy rules are called mixed, since they can handle different types of features: continuous, granulated, and nominal features. A mixed rule on the feature space is defined through a fuzzy set which assigns a degree of fulfilment for a data point $\vec{x} = (x_1, \ldots, x_i, \ldots, x_n)$:

$$\mu(R, \vec{x}) = \min_{i=1,\ldots,n} \{\mu_i\{c_i^{supp}, c_i^{core}, x_i\}\} \tag{1}$$

Each one-dimensional constraint c_i defines a subset of the corresponding domain D_i. In case of numerical attributes intervals are used for the core region c_i^{core} and the support region c_i^{supp}. The trapezoidal membership functions μ_i are then defined as follows: membership values of 1 are assigned one to patterns that fall inside the core region (area of evidence). The membership degrees linearly decline until they reach 0 for patterns with attribute values outside the support region (area of support). The sets c_i^{supp} can contain the entire domain of values $(\infty, -\infty)$, i.e. the corresponding domain is not constrained at all. During a learning epoch each training pattern is analyzed and the existing rules are modified or new rules are inserted into the rule base. If a presented pattern is already covered by a rule of the correct class, the core region of the rule is eventually widened in order to increase the membership degree of the pattern to 1. If there is no rule that covers the presented pattern, a new rule for the class of the pattern is created such that the core region of the new rule only covers the pattern itself. The support area of the new rule covers the entire feature space, i.e. $c_i^{supp} = (\infty, -\infty)$, $\forall i$.

In any of the two cases above the algorithm ensures that no existing rule of conflicting class covers the presented pattern. This is achieved by modifying the constraints in the conflicting rules such that the pattern is not covered anymore. The support region of the conflicting rule is reduced if the presented pattern $\vec{x}$ lies in the support region of that rule. Then among all available attributes one dimension i is chosen for which the constraint c_i^{supp} can be reduced such that the specialization results in minimal loss of volume covered by the rule. This is a heuristic for deciding on the most important attribute for class separation in the rule's local part of the feature space. If the presented pattern $\vec{x}$ lies in the core region of a rule of a different class, both core and support region in one dimension are modified accordingly. In this case the coverage of previously presented patterns gets lost. However, new rules for these patterns are created in later epochs of this pattern-by-pattern approach. Usually the algorithm terminates after only a few presentations of the entire training set. Finally, all patterns are covered by fuzzy rules which only constrain a small individual subset of features that are relevant for discriminating between classes in the particular parts of the feature space.

An extension of the algorithm for dealing with outliers and rare phenomena in the data has been proposed [11]. It aims at building models on different levels of detail. Rules that cover a small number of uninformative training patterns are used as a filter. Training patterns covered by those irrelevant rules are removed from the training set. Afterwards, the algorithm is executed again on the remaining data only. This yields a coarser as well as more compact model. The resulting rule bases are easier to interpret and describe the more relevant concepts in the data. Repetitive filtering can be used to build hierarchical rule bases with the most detailed model at the top and the more general rule bases on lower levels. A very simple approach for dealing with potential outliers or irrelevant patterns is to include all rules into the filter model that cover a number of patterns lower than a given threshold. The construction of the hierarchy of rule bases stops if no more rules fall under the threshold.

6 Application and Results

This section describes the application of the selected classifiers to our preprocessed 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, classifier peculiarities, and steps to improve the classification. To measure the performance of the classifiers we present classification accuracies on training and test data (Table 7) and the confusion matrices on the test data (Table 3, Table 5, Table 6). The accuracies, i.e. the averaged relative and absolute number of misclassifications 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 show 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 a large number of different parameter settings, here we only report the results that we consider to be optimal.

6.1 Decision trees

For the induction of the decision trees we tried several attribute selection measures, as described in [3]. Most of the measures yield reasonable results. However, the Symmetric Gini Index maximized the tree accuracy over the training data set so we employed it as split criteria. For the pruning we use confidence level pruning [9] with a confidence of 50%. The classification accuracy is 95.6% on the training and 87.20% on the test data.

Decision Trees	1)	2)	3)	4)	5)	6)	7)
1) *bulge*	46	0	1	0	0	0	0
2) *dent*	0	50	0	0	1	0	0
3) *flatarea*	0	0	3	2	3	1	0
4) *pressmark*	2	0	0	53	1	2	0
5) *sinkmark*	0	0	3	0	12	9	0
6) *unevensurface*	0	0	0	1	4	70	1
7) *unevenradius*	0	0	0	0	0	4	4

Table 3: Confusion matrix for Decision Trees

R0: IF sum_maxima >≈ 10,05 AND sum_minima >≈ 10,74 AND area_inner_circle >≈ 9274,46 AND volume_3_volume_all_ratio <≈ 0 THEN **Class** = *unevenradius*
R1: IF sum_maxima >≈ 10,05 AND sum_minima ≈ 3,62 AND area_inner_circle >≈ 9274,46 AND volume_3_volume_all_ratio <≈ 0 THEN **Class** = *flatarea*
R2: IF sum_maxima <≈ 1,36 AND area_inner_circle <≈ 86,6 AND volume_3_volume_all_ratio >≈ 0,47 THEN **Class** = *dent*
R3: IF sum_maxima <≈ 1,36 AND sum_minima ≈ 3,62 AND area_inner_circle <≈ 86,6 AND volume_3_volume_all_ratio <≈ 0 THEN **Class** = *pressmark*
R4: IF sum_maxima>≈ 10,05 AND sum_minima ≈ 1,74 AND area_inner_circle <≈ 86,6 AND volume_3_volume_all_ratio <≈ 0 THEN **Class** = *sinkmark*
R5: IF sum_minima <≈ 0,5 AND volume_3_volume_all_ratio <≈ 0 THEN **Class** = *bulge*
R6: IF sum_minima >≈ 10,74 AND area_inner_circle <≈ 86,6 AND volume_3_volume_all_ratio <≈ 0 THEN **Class** = *unevensurface*
R7: IF sum_maxima >≈ 10,05 AND sum_minima ≈ 3,62 AND area_inner_circle <≈ 86,6 AND volume_3_volume_all_ratio <≈ 0 THEN **Class** = *unevensurface*

Table 4: NEFCLASS Rules.

6.2 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 the subset of the 15 best features which were determined by explicit feature selection (see section 5). This made it easier to find good and general parameter settings for NEFCLASS. After extensive pruning the best classification accuracy was 90.11% in average on the training sets and 81.32% on the test sets. The rules of the obtained neuro-fuzzy classifier are shown in table 4.

NEFCLASS	1)	2)	3)	4)	5)	6)	7)
1) *bulge*	47	0	0	0	0	0	0
2) *dent*	0	46	0	5	0	0	0
3) *flatarea*	0	1	0	5	3	0	0
4) *pressmark*	0	1	0	55	1	1	0
5) *sinkmark*	0	0	0	6	9	9	0
6) *unevensurface*	0	0	1	7	2	65	1
7) *unevenradius*	0	0	0	1	0	7	0

Table 5: Confusion matrix for NEFCLASS

Mixed Fuzzy Rules	1)	2)	3)	4)	5)	6)	7)
1) *bulge*	47	0	0	0	0	0	0
2) *dent*	0	51	0	0	0	0	0
3) *flatarea*	0	0	0	5	4	0	0
4) *pressmark*	0	0	0	50	1	7	0
5) *sinkmark*	0	0	0	5	7	12	0
6) *unevensurface*	0	0	0	4	1	71	0
7) *unevenradius*	0	0	0	0	0	4	4

Table 6: Confusion matrix for the mixed fuzzy rules

6.3 Mixed Fuzzy Rule Formation

We applied the constructive fuzzy rule induction algorithm on the entire dataset with all features, since it is suited to learn from high dimensional data. Neither feature selection nor an initial fuzzy partitioning of the attribute scales is needed. In the first run the induction algorithm always constructs a rule base that classifies all training patterns correctly. Pruning is required in order to avoid over-fitting. Ideally, some relevance measure should be used to identify irrelevant rules. However, in our experiments we applied a very simple pruning strategy: As the weight of a rule we consider the number of covered training patterns. Outliers are likely to be covered by rules with very low weight. Thus, we removed all patterns from the training set which were covered by rules with lower weight than a threshold t. We repeated experiments with a range of thresholds from 1 to 10 to obtain more general rule bases in a second run of the algorithm on the remaining training data. Instead of constructing an entire hierarchy of rules we determined training and test accuracy on basis of the less detailed rule bases after only one filtering/pruning step. For t = 4 the average training error of 93.17% was obtained. The average classification accuracy of the rule bases for this threshold on the test data sets was 84.20%.

Average Accuracies	Decision Trees	NEFCLASS	Mixed Fuzzy Rules
Training Set	95.6%	90.11%	93.17%
Test Set	87.2%	81.32%	84.20%

Table 7: Classification Accuracy on the Training and Test Cases

7 Discussion

Overall classification accuracy improved compared to the results in our previous work [12]. The average test accuracy of decision trees is enhanced by 4.78%, NEFCLASS improved by 5.88%, and the mixed fuzzy rules by 8.38%. Looking at the confusion matrices tells us about the relations of the majority classes. From the lower number of errors we can see, that *bulge* and *dent* are better separated from the other classes than *sinkmark* and *unevensurface*. The unsatisfactory detection of *sinkmarks* countersigns our preliminary results, since the classification of this defect in the experiments nighter improved nor got worse. Decision trees and the fuzzy rule bases had problems to distinguish *sinkmarks* from the class *unevensurface*.

Comparing the confusion matrices with those yielded from previous experiments, we see that a higher number of examples of the types *unevensurface* and *pressmark* were correctly classified. In the defect class *unevensurface* decision trees were in 18 cases more accurate than before, NEFCLASS in 7, and mixed fuzzy rules in 11 cases. The respective numbers of additionally correct classifications for the defect type *pressmark* are 7 in decision trees, and 8 and 6 in the fuzzy rule approaches. Furthermore, the two classes are unchanging well separated by the classifiers as in the previous experiments. Thus, we see our suspicion underpinned, that the examples in the defect type *unevensurface/ pressmark* were indeed either of type *pressmark* or *unevensurface*. Otherwise the relabelling of these examples would not have resulted in improved classification accuracy. The similarity of the defects to the two distinctive classes in the clustering model resulted in reasonable class assignments such that both classes are still well separated by the classifiers.

8 Conclusions and Outlook

The presented 3-D image processing approach from surface digitization to defect type classification yields promising results. The achieved accuracy improvements in our study clearly showed that training set consistency and revising expert decisions during classifier design are of high importance when class labels are uncertain and likely to have errors. The relabelling of examples helped to minimize the uncertainty that was contained in the class label *unevensurface/ pressmark*. It results in a more consistent training set with which we obtained more accurate classifiers than in previous experiments.

Currently, we still have not obtained reliable descriptions of all classes, yet. In the next step of the project, we will generate a larger database to solve another data quality issue. That is, more representative examples of the rare classes are needed. 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] B. Jähne. Digital Image Processing. 5th rev. ed., Springer Verlag, Berlin, 2001.

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

[3] 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.

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

[5] Ian H. Witten, and Eibe Frank. Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations. Morgan Kaufmann, San Francisco, 2000.

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

[8] 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.

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

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

[11] Michael R. Berthold, Mixed Fuzzy Rule Formation, International Journal of Approximate Reasoning (IJAR), Volume 32, Issue 2-3, pages 67-84, 2003.

[12] Andreas Eichhorn, Christian Döring, Aljoscha Klose, and Rudolf Kruse. Classification of surface form deviations for quality analysis. In Proc. of the European Symposium on Intelligent Technologies, Hybrid Systems and their Implementation on Smart Adaptive Systems (EUNITE 2003), pages 121 - 129, Oulu, Finnland, July 10 - July 11 2003.

[13] Bogdan Gabrys und Lina Petrakieva. Combining labelled and unlabelled data in the design of pattern classifcation systems. Proc. of the European Symposium on Intelligent Technologies, Hybid Systems and Their Implementation on Smart Adaptive Systems (EUNITE 2002), Albufeira, Portugal, 2002