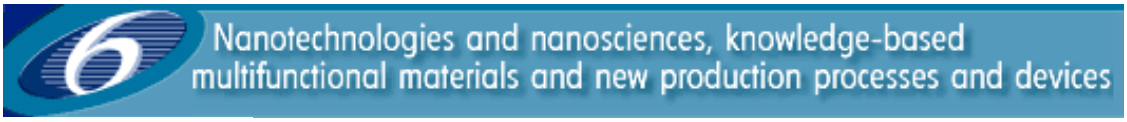

STRP 016429
DynaVis
Dynamically
Reconfigurable Quality Control
for Manufacturing and
Production Processes Using
Learning Machine Vision

SIXTH FRAMEWORK PROGRAMME
Priorities 2 & 3
NMP & IST - STREP

Publishable Final Activity Report

Period covered from: 1st of October 2005 to 30th of Sept. 2008 Date of preparation: 01.10.2008

Start date of project: 1st October 2005 Duration: 36 months

Coordinator:
Dr. Christian Eitzinger
Profactor GmbH

Revision 1

Project Execution

Inspection of products by machine vision often has to solve the problem of how to implement a human decision-making process in software. Currently, this requires a step-by-step reprogramming or parameterization of the software, which may take very long. The results of this project will enable us to use human-machine cooperation to learn complicated inspection tasks instead of step-by-step improvements and adaptations of software. The system will automatically adapt to specific (or changing) requirements.

Figure 1: Visual inspection of die-cast parts.

The project is focused on the development of "trainable" machine vision algorithms and of appropriate machine learning techniques. In order to create such methods we will focus on the following scientific objectives:

- machine learning methods for processing the complicated data produced by the vision system
- methods to deal with multiple, possibly contradictory input by the operators
- methods for predicting success or failure of the learning process in early stages of the training process

DynaVis will enable the machine vision system to **directly learn from the human operator**. Based on this input the machine vision system will gradually build a hypothesis about which parts are good and which are bad.

The structure that was implemented in the project is shown in *Figure 2*. Problem-specific (non-adaptive) preprocessing will be used to extract relevant regions in the image. Each of these regions, which correspond to potential faults, will be converted to a feature vector. All of the feature vectors will then be input to a learning classifier structure that returns a gradual good/bad decision. Human operators have the possibility to confirm or reject the system's output. Based on this input, the system learns.

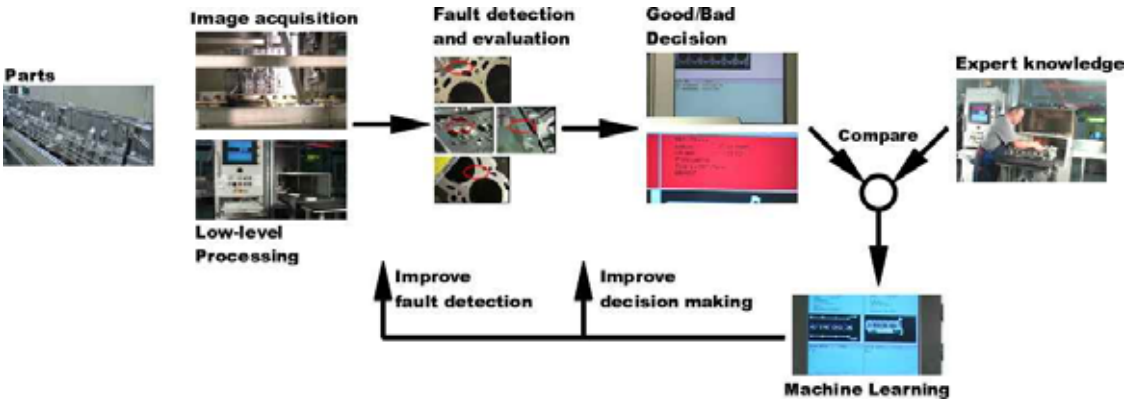

Figure 2: The learning process

Methods of how best to present the systems output to the operators so as to maximize the quality and nature of their feedback have also been investigated

.

The approach differs from existing methods by the fact that we consider a much larger degree of flexibility and adaptivity. Standard approaches usually only train the final classifier, while we are planning to go at least one step further backwards in the processing chain and also consider fault detection and features as part of the trainable structure.

Contractors

Participant name	**Short name**	**Country**
Profactor GmbH	Profactor	*AUT*
Katholieke Universiteit Leuven	K.U.Leuven	*BE*
University of the West of England, Bristol	UWE	*UK*
Johannes Kepler University of Linz	JKU	*AUT*
Sony DADC Austria AG	Sony DADC	*AUT*
Asentics GmbH & Co KG	Asentics	*GER*
The European Association of Innovating SMEs	Eurexcel	*UK*
Atlas Copco	ATC	*BE*

Coordinator:
Dr. Christian Eitzinger
Profactor GmbH
E-mail: christian.eitzinger@profactor.at
Tel.: +43 7252 885 250
Fax: +43 7252 885 101

Project objectives

In order to make sure that the training process results in usable decision structures of high quality, the following goals were defined

⇒ (1) the inspection system is able to learn a set of 5 non-trivial rules (such as: part is bad, if there are at least 5 faults with diameter>1.5mm) purely from a good/bad input provided by the operator. The system reaches an error rate of 0.5% after 10.000 presentations of input data.

⇒ (2) the vision system is able to learn a set of 5 non-trivial rules (such as: part is bad, if there are at least 5 faults with diameter>1.5mm). The operator will provide more detailed input, but 10% of his/her input will be random. The system reaches an error rate of 1% after 10.000 presentations of input data.

⇒ (3) the vision system will be trained on purely random input. It has to detect its failure to make progress after less than 1000 presentations.

These goals are selected to cover the most relevant criteria that enable later use of the results: The accuracy of the classifier structure has to be very high (99.5%) and it does not degrade substantially if there is random training input (99%). Also we need to detect early (after 1000 parts) if the training process will not be successful, i.e. it does not reach a certain classification accuracy.

Two demonstrators were planned in the project:

⇒ (4) an existing **inspection system for prints on CDs** and DVDs located at Sony DADC. This system will be extended with machine learning capabilities. Goal of the demonstrator is to reduce the number of falsely detected faults by learning.

⇒ (5) an **inspection system for surface inspection of compressor parts** of Atlas Copco. A demonstrator for this inspection system will be built during the project and will be able to learn an inspection task just by cooperating with human operator(s) (error rate < 1%).

During the initial phase of the project a large set of test images was collected from a number of different applications such as print inspection, surface inspection and inspection of food. These data were created by the industrial partners in the project (Atlas Copco, Sony DADC, Asentics) with the help of the RTD performers. In addition to these real world data sets another set of 100.000 artificial test images was generated for testing.

These data were then used in 3 parallel activities that investigated different steps of the processing chain.

The first activity was dedicated to the lower-level processing steps. For each of the faults in the image a set of features needs to be calculated that contains information about the faults. These features include quantities such as the size of the fault, its position, shape descriptors and histogram information about the fault. The main goal of the activities was to make the feature calculation adaptive. The solution was to define a set of parameters

that have influence on the feature calculation and to adapt these parameters so as to maximize the separation between the good and bad classes. The optimization criterion was chosen different for each classifier so as to increase the separation perpendicular to the decision boundary of the classifier.
A second topic was how to group the faults to form a larger single fault. The basic methods such as connected components consider each single, isolated pixel as a fault, which – in some applications – results in an extremely large number of faults. To solve this problem clustering methods were investigated that merge single pixels to approximate human perception, thus grouping those pixels that would also be perceived as belonging together by a human.

The second activity was focused on the downstream classifier structure. The consortium used a number of existing classifiers (such as nearest neighbor, decision trees, …) but also developed new methods based on evolving vector quantization and fuzzy inference systems.
The key question that had to be addressed was the fact that the number of potential defects is different for each image. Thus a varying number of feature vectors is generated for each image, which – due to the varying dimension of the input – prohibits the direct application of existing classifier methods. To solve this problem a range of pre-processing methods for the feature vectors were investigated. The most successful (in terms of classification accuracy) was a semi-supervised approach, where the single feature vectors were grouped in a fixed number of clusters and statistical quantities of these clusters (such as number of objects in the cluster) were used as input to the downstream classifiers. Other methods that were based on aggregating all the features vectors also achieved reasonable results, although not as good as the semi-supervised approach.

The third activity investigated the user's role in the training process.
A major topic was how to deal with the fact that training input is provided by a number of different experts who all provide systematically different input. Instead of averaging over all these inputs we solved the problem by creating personalized classifiers and merging these classifiers using ensemble methods. A range of existing methods were tested and new ones were developed. The most successful was an enhanced Grading method that resulted in significant improvements of the classification accuracy, in particular in the presence of substantial contradictions.
A second topic was the prediction of the success or failure of the training process. A bias/variance decomposition approach was developed that could accurately predict the achievable classification accuracy. It was tested not only on the classifiers and test data considered in DynaVis but on a much larger range and achieved remarkable precision in predicting the final accuracy. Also a hypothetical "Oracle" ensemble method was implemented that provided a good upper bound on the classification accuracy. Furthermore, a prediction method was implemented which predicts the achievable classification accuracy by applying a regression method to the observed classification accuracies and provides an estimate of *how* the accuracy will progress.
Another topic in this activity was the user interface that should be developed to maximize the quality of the training input. Different phases during the creation of an inspection

system (installation, cooperation mode, automatic mode, maintenance) were considered. In the last year of the project these interfaces were implemented for a specific application in the food industry.

Project results

Classifiers	Standard						Novel Developments			
Data sets	C4.5	CART	k-NN	SVMs*	NN*	1NN /TS*	NB/TS*	Ens. 1NN/TS*	eVQ-Class	Flexfis Class*
ArtifData01	99,69	99,68	99,33	99,66	99,40	99,36	99,58	99,49	99,29	99,60
ArtifData02	99,84	99,83	99,64	99,80	99,57	99,64	99,75	99,76	99,68	99,78
ArtifData03	99,78	99,79	99,52	99,74	99,54	99,52	99,68	99,68	99,55	99,70
ArtifData04	99,65	99,62	99,32	99,61	99,31	99,34	99,52	99,44	99,41	99,56
ArtifData05	99,69	99,68	99,45	99,65	99,32	99,46	99,54	99,59	99,46	99,58
BearingEdge	99,34	98,81	99,42	99,50	96,00	99,62	98,48	99,70	99,21	99,07
BearingSurface	98,84	98,91	98,80	99,81	96,00	99,02	98,54	99,11	99,31	99,08
Eggs	99,82	99,84	99,52	99,85	99,46	99,56	99,70	99,86	99,41	99,80
Sony Op05	99,72	99,78	99,71	99,80	99,48	99,79	99,71	99,82	99,74	99,64
Sony Op06	99,82	99,91	99,83	99,93	99,70	99,90	99,83	99,92	99,87	99,81
Sony Op08	99,76	99,92	99,74	99,86	99,56	99,79	99,74	99,84	99,76	99,76
Sony Op09	99,80	99,88	99,79	99,88	99,61	99,86	99,75	99,88	99,82	99,74
Sony SuperOp	99,81	99,81	99,77	99,83	96,00	96,00	96,00	96,00	99,67	99,58
Rotor	99,51	99,42	99,47	99,56	99,35	99,72	99,80	99,82	99,65	99,38

Table: Classification accuracies for different classifiers on the artificial and real-world data sets. The accuracies are based on the assumption of a 96%/4% split between good and bad parts.

The table above shows the classification results for all the classifiers on all the test data that were used in DynaVis. It is obvious that for almost all of the test data at least one classifier achieves an accuracy above 99.5% (object (1) of the project), with C4.5, CART and SVMs being the most successful ones. The ensemble methods that were implemented achieved a further increase in accuracy over the single classifiers.
A similar test was also conducted with training data with 10% random labels and again for most of the classifiers an accuracy of >99% was achieved, which corresponds to objective (2).

Regarding the prediction of the final accuracy, tests were conducted on a wide range of test data and classifiers. The results are summarized in the table below (see next page). The red bar in the center shows the predicted error, while the blue one shows the actual error (determined on a much larger data set using 10-fold cross-validation). It is obvious that the method developed in DynaVis can quite accurately predict the final accuracy after just 1000 presentations of input images. This completes objective (3).

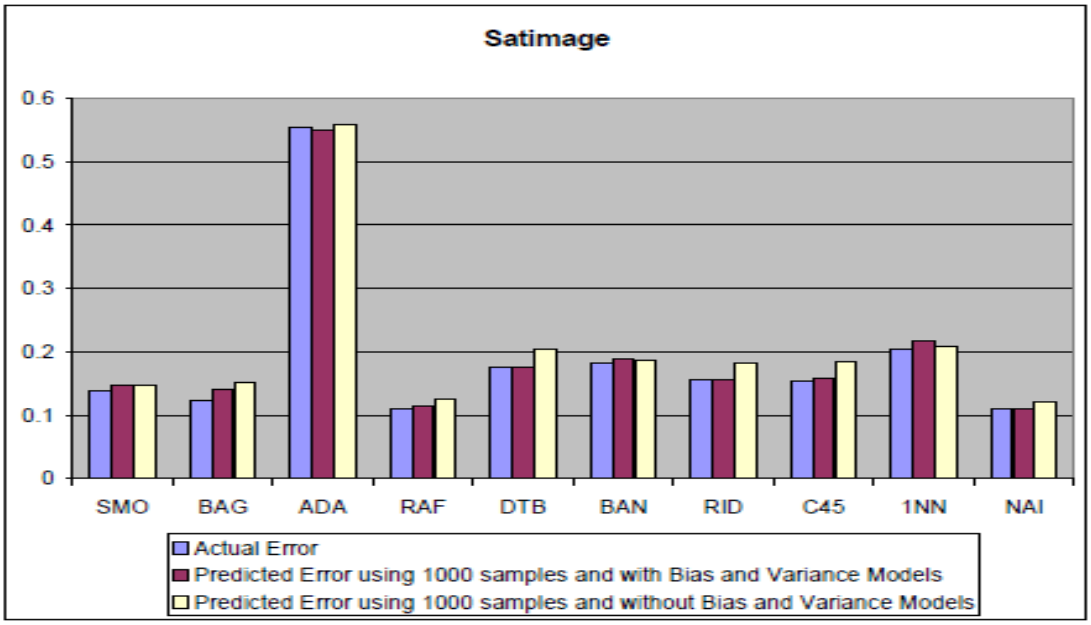

Table: Comparison of predicted and actual accuracy achieved by different classifiers.

Demonstrators

Two hardware demonstrators were set up and also shown at the final meeting. The first one was the "rotor scanner". This demonstrator was built early in the project and used to scan images of rotors at Atlas Copco. In the last year of the project it was converted to a demonstrator that also processes the images, calculates features and classifies the images with a set of trained classifiers and ensembles.

The second demonstrator was based on a CD print inspection system that checks the printed side of the CD for any kind of printing defects. A hardware setup consisting of a light source and camera was built and integrated with the DynaVis software. The classifiers were trained on existing data sets and could be shown to improve the decisions made by the original inspection software, in particular by recognizing so called pseudo-defects.

Dries Verduyn of Atlas Copco presenting the Rotor inspection demonstrator (left). Thomas Radauer of Sony presenting the CD print inspection (right).

Deliverables

Del. no.	Deliverable title	WP no.	Nature	Diss. level	Deliv. Month plan/ actual
1	Inspection tasks for a dynamically reconfigurable quality control system.	1	R	CO	6/6
2	Project presentation	7	R	PU	6/6
3	Applicable cooperation concepts for machine vision applications	1	R	RE	9/9
4	Common concepts for trainable machine vision	1	R	RE	12/12
5	Machine learning methods applicable for industrial quality control systems	3	R	RE	12/12
6	Influence and processing of contradictory input, Operator acceptance	4	R	RE	12/12
7	Concept for trainable machine vision algorithms	2	R	RE	18/18
8	Report on formation of Enterprise Grouping and Industrial Referee Group	5	R	PU	18/18
9	Implementation of machine learning for quality control systems	3	P	RE	24/24
10	Report on dissemination activities including workshops and seminars	5	R	PU	24/24
11	Method for predicting the success of learning	4	P	RE	27/27
12	Implementation of trainable machine vision algorithms	2	P	RE	30/30
13	Concepts of providing feedback to the user	4	R	RE	30/30
14	Human-machine interaction interface implementation	4	P	CO	33/34
15	Integration of machine vision methods and machine learning	3	P	RE	36/36
16	Demonstration of machine learning for industrial quality control	6	D	PU	36/36
17	Report on raising public participation and awareness	5	R	PU	36/36
18	Final Plan for Using and Disseminating Knowledge	5	R	CO	36/36

Dissemination and Use

3 Publishable Results

This section provides a publishable summary of exploitable results the DynaVis project has generated. Especially with respect to commercial available products the consortium should be ready to publicise and have taken appropriate measures to protect their IPR. A number of DynaVis results, which were listed in section 1 are company-specific and therefore restricted it in use. For this reason, these results are excluded from publication.

3.1 Planned products

This section summarises results (partly comprised from single results in sections 1.1 and 1.2) which may become products in the near future. The publication in CORDIS as well as the usage through the European Commission in its own promotional material will support the dissemination and exploitation of DynaVis results, also under commercial aspects. The following tables characterising publishable results are sorted according to the exploiting project partner.

3.1.1 JKU - Publishable Results

Product Name	**Exploiting Partner**	**Contact Details**
Image Classifier Package including: SVMClassifier.DLL ClusterBasedClassifier.DLL CARTClassifier.DLL	**Johannes Kepler University Linz** Department of Knowledge-Based Mathematical Systems	Dr. Edwin Lughofer Edwin.lughofer@jku.at
Product Description The image classifier package contains three software libraries (in form of DLLs), namely SVMClassifier.DLL, ClusterBasedClassifier.DLL and CARTClassifier.DLL, that allow the user to create classifiers for images. In principal, they can be used with arbitrary feature sets, but were specifically designed for surface inspection tasks under the scope of DynaVis, i.e. they have been designed for interoperability within the DynaVis framework for image inspection tasks. All libraries support the following functionality: – Batch mode training from a training data set with supplied class labels – Predicting the class labels for previously unseen data – Loading and saving classifiers from and to disc – Returning a feature importance list in decreasing order – Sample-wise incremental adaptation of classifiers (for on-line mode, only for ClusterBasedClassifier.DLL applicable) Note that the cluster-based classifier is a novel methodological development which is based on an evolving clustering technique. The interfaces are designed in a way that they can be simply incorporated externally as required, they are documented in ‘classifiers.chm’.		
Target Markets	**Product Status**	**IPR Status**
Machine Vision and general Machine Learning applications	Laboratory prototypes	Copyrighted software
Collaboration sought and/or offered Collaboration sought with machine vision and machine learning companies that want to use/license to		

software and/or participate in projects for joint further development.

Product Name	**Exploiting Partner**	**Contact Details**
TimViewer	**Johannes Kepler University Linz** Department of Knowledge-Based Mathematical Systems	Dr. Edwin Lughofer Edwin.lughofer@jku.at DI Roland Richter Roland.richter@jku.at
Product Description The TimViewer software tool is a stand-alone executable with an extensive GUI whose basic functionality provides a support for extracting and labelling objects in contrast images from surface inspection processes (after removing the application dependent elements). A built in feature extraction library contains an enhanced clustering approach for object extraction and a computation of 57 object and 17 aggregated features after the DynaVis feature definition document. In the Annotation Wizard labels for single objects as well as for whole images are suggested by built in classifiers and may be corrected by the operator with simple mouse clicks. According to these suggestions, the workload of operators for labelling a new set of images is significantly decreased. He also may provide certainty levels in his labels. The TimViewer tool is documented in 'TimViewer.chm'.		
Target Markets	**Product Status**	**IPR Status**
Machine Vision, Machine Learning and Image Processing applications	Laboratory prototype	Copyrighted software
Collaboration sought and/or offered Collaboration sought with machine vision, machine learning and image processing companies that want to use/license to software and/or participate in projects for joint further development. The software is copyright-protected.		

Product Name	**Exploiting Partner**	**Contact Details**
TIFFwriter.dll	**Johannes Kepler University Linz** Department of Knowledge-Based Mathematical Systems	Dr. Edwin Lughofer Edwin.lughofer@jku.at DI Roland Richter Roland.richter@jku.at
Product Description The TIFFwriter.dll is a software library for writing and reading TIFF-images in the specified DynaVis format.		
Target Markets	**Product Status**	**IPR Status**
Image Processing applications	Laboratory prototype	Copyrighted software
Collaboration sought and/or offered Collaboration sought with image processing companies that want to use/license to software and/or participate in projects for joint further development.		

Product Name	**Exploiting Partner**	**Contact Details**
FLLLAME	**Johannes Kepler University Linz** Department of Knowledge-Based Mathematical Systems	Dr. Edwin Lughofer Edwin.lughofer@jku.at DI Roland Richter Roland.richter@jku.at
Product Description FLLLAME is a software tool with a GUI, which can be used for training of classifiers from a loaded feature set. Built in functionality includes, feature selection, best parameter grid search and cross-validation for obtaining an estimation of the expected prediction error as well as a final training of classifiers. Every classifier.dll meeting the DynaVis C++ interface definitions can be simply plugged into the GUI.		
Target Markets Machine Learning applications	**Product Status** Laboratory prototype	**IPR Status** Copyrighted software
Collaboration sought and/or offered Collaboration sought with image processing companies that want to use/license to software and/or participate in projects for joint further development. The software is copyright-protected.		

Product Name	**Exploiting Partner**	**Contact Details**
MATLAB's feature extraction software	**Johannes Kepler University Linz** Department of Knowledge-Based Mathematical Systems	Dr. Edwin Lughofer Email Edwin.lughofer@jku.at
Product Description MATLAB's feature extraction software (as p-code) provides the functionality for extracting single objects from contrast TIFF-images (format as specified in DynaVis) and for calculating a list of pre-defined features (57 object features, 17 aggregated features) from these objects. Applied object recognition approaches can be selected through a call parameter in the interface and include connected components, morphology plus connected components and clustering approaches such as hierarchical clustering, iterative vector quantization, reduced delaunay graph, DBSCAN, normalized cut and mean shift. The feature vectors are stored in single features matrices (one for the object features, one for the aggregated features) onto the hard disc, where they can be used for further processing (e.g. in FLLLAME or the classifier DLLs). The software is documented in 'ReadMe_feat_extract.txt', the interfaces of the single object recognition approaches are defined in 'ROI_Extraction_Interfaces.txt'.		
Target Markets Image processing applications	**Product Status** Laboratory prototype	**IPR Status** Copyrighted software
Collaboration sought and/or offered Collaboration sought with image processing companies that want to use/license to software and/or participate in projects for joint further development.		

Product Name	**Exploiting Partner**	**Contact Details**
ImageFeature.DLL	**Johannes Kepler University Linz** Department of Knowledge-Based Mathematical Systems	**Contact Details** Dr. Edwin Lughofer Edwin.lughofer@jku.at DI Roland Richter Roland.richter@jku.at
Product Description The ImageFeature.DLL is a software library which extracts objects with a modified hierarchical clustering approach and calculates all the 57 objects and 17 aggregated features defined in the DynaVis feature definition document.		
Target Markets	**Product Status**	**IPR Status**
Image processing applications	Laboratory prototype	Copyrighted software
Collaboration sought and/or offered Collaboration sought with image processing companies that want to use/license to software and/or participate in projects for joint further development. The software is copyright-protected.		

3.1.2 KUL - Publishable Results

<table>
<tr><td>Product Name
ENSEMBLE.DLL</td><td>Exploiting Partner
Katholieke Universiteit Leuven
Department of Mechanical Engineering
Celestijnenlaan 300B – bus 2420
B-3001 Heverlee (Leuven), Belgium
Web</td><td>Contact Details
Davy Sannen
davy.sannen@mech.kuleuven.be

dr. ir. Hendrik Van Brussel
hendrik.vanbrussel@mech.kuleuven.be</td></tr>
<tr><td colspan="3">Product Description
Ensemble.dll is a software library that allows the users to combine the outputs of different classifiers using ensemble techniques. The supported ensemble techniques include fixed schemes such as voting and simple algebraic connectives as well as trainable methods such as fuzzy integrals, Decision Templates and standard and discounted Dempster-Shafer combination. The ensembles predict the class labels for previously unseen data based on the outputs of the classifiers within the ensemble. The trainable ensembles are trained in batch mode beforehand using the outputs of the different classifiers in the ensemble for the training data together with the class labels. They can also be adapted incrementally in a sample-wise manner. All of the ensembles have the advantage of being designed for interoperability within the DynaVis framework for image inspection tasks. However, the interfaces are designed in such a way that they can be simply incorporated externally as required.</td></tr>
<tr><td>Target Markets
Machine Vision and general Machine Learning applications</td><td>Product Status
Laboratory prototype, demonstrator</td><td>IPR Status
Copyrighted software</td></tr>
<tr><td colspan="3">Collaboration sought and/or offered
Collaboration was sought with Machine Vision and Machine Learning companies that want to use/license software and/or participate in projects for joint further development.</td></tr>
</table>

Product Name	**Exploiting Partner**	**Contact Details**
ORACLE.DLL	**Katholieke Universiteit Leuven** Department of Mechanical Engineering Celestijnenlaan 300B – bus 2420 B-3001 Heverlee (Leuven), Belgium Web	Davy Sannen davy.sannen@mech.kuleuven.be dr. ir. Hendrik Van Brussel hendrik.vanbrussel@mech.kuleuven.be
Product Description Oracle.dll is a software library that predicts an upper bound on the accuracy of classifiers and ensembles for a given initial data set. The value of this upper bound is that it can be used as an "early warning" system in the case that the predicted upper bound on the accuracy is not going to be sufficient, indicating that a revised data collection or other classification/ensemble methods are required. Additionally, the upper bound can be used to better estimate the accuracy curves of the classifiers and ensembles after further training, indicating how the accuracy will proceed towards the estimated upper bound.		
Target Markets Machine Vision and general Machine Learning applications	**Product Status** Laboratory prototype, demonstrator	**IPR Status** Copyrighted software
Collaboration sought and/or offered Collaboration was sought with Machine Vision and Machine Learning companies that want to use/license software and/or participate in projects for joint further development.		

3.1.3 UWE - Publishable Results

Product Name	Exploiting Partner	Contact Details
kNN.dll naivebayes.dll decision_tree.dll	**University of the West of England** Bristol Institute of Technology Bristol, BS16 1QY, UK Web	Dr Jim Smith Email james.smith@uwe.ac.uk

Product Description

kNN.dll, naivebayes.dll and decision_tree.dll are software libraries that allow the user to create classifiers of the given types that can (i) learn from a set of training data with supplied class labels, and then (ii) be used to predict the class labels of previously unseen data. All three libraries support off-line or "batch-mode" training, naivebayes and kNN also support incremental or "on-line" learning. All three have the advantage that they have been designed for interoperability within the DynaVis framework for image inspection tasks. However the interfaces are so designed that they can be simply incorporated externally as required. All libraries support the following functionality:

- Batch mode training from a training data set with supplied class labels
- Predicting the class labels for previously unseen data
- Loading and saving classifiers from and to disc
- Returning a feature importance list in decreasing order
- Sample-wise incremental adaptation of classifiers (for on-line mode, only kNN.dll and naivebayes.dll)

Target Markets	Product Status	IPR Status
Machine Vision and general Machine Learning applications	Laboratory prototype	Copyrighted software

Collaboration sought and/or offered

Collaboration sought with machine vision and machine learning companies that want to use/license to software and/or participate in projects for joint further development.

Product Name	Exploiting Partner	Contact Details
prediction_of_learning.dll	**University of the West of England** Bristol Institute of Technology Bristol, BS16 1QY, UK Web	Dr Jim Smith Email james.smith@uwe.ac.uk

Product Description

prediction_of_learning.dll is a software library, that given an initial data set, and a choice of classifier to use, is able to calculate reliable estimates of the current error, and how it can be decomposed into terms relating to bias (resulting from the choice of classification method) and variance (resulting from stochastic forces). The principal novelty and value is that it then uses these calculations to make high accurate predictions of the final achievable error values of the system after further training. This has the advantage that it can act as an "early warning" system in the case that the final achievable accuracy is not going to be sufficient, triggering revised data collection or classification methods before further user effort in labelling training items is incurred.

Target Markets	Product Status	IPR Status

Machine Vision and general Machine Learning applications	Laboratory prototype	Copyrighted software
Collaboration sought and/or offered Collaboration sought with machine vision and machine learning companies that want to use/license to software and/or participate in projects for joint further development.		

3.1.4 Profactor – Publishable Results

Product Name	**Exploiting Partner**	**Contact Details**
FEATURE.DLL	Profactor GmbH Im Stadtgut A2 4407 Steyr-Gleink, Austria Web www.profactor.at	Dr. Christian Eitzinger christian.eitzinger@profactor.at
Product Description Feature.dll is a software library containing a set of features typically used as blob descriptors in machine vision applications. The library allows an automatic optimization of feature calculation to facilitate classification of the blobs (e.g. different types of defects).		
Target Markets	**Product Status**	**IPR Status**
Machine Vision	Laboratory prototype	Copyrighted software
Collaboration sought and/or offered Collaboration sought with machine vision companies that want to use the software and participate in projects for joint further development.		

3.2 Other publishable results

The deliverables [10], [11] and [12] can be used as input for a revised technical documentation of DynaVis related products. It is proposed to publicise the before mentioned product-related documentation instead of the deliverables.

Additionally, all the subsequently listed scientific publications that have been prepared within DynaVis project can be regarded as publishable results:

Tahir, M.A., and Smith, J.E. (2006) **"Improving Nearest Neighbor Classifier Using Tabu Search and Ensemble Distance Metrics"** pp 1086-1090 in "ICDM'06, Proceedings of the Sixth International Conference on Data mining, IEEE Press
J. E. Smith and M. A. Tahir (2007), **"Stop Wasting Time: On Predicting the Success or Failure of Learning for Industrial Applications"**, 8th International Conference on Intelligent Data Engineering and Automated Learning (IDEAL'08), Lecture Notes in Computer Science, LNCS 4881, pp 673-683, Springer Verlag, December 2007.
Muhammad Atif Tahir, Jim E. Smith: **Feature Selection for Heterogeneous Ensembles of Nearest-neighbour Classifiers Using Hybrid Tabu Search**. Advances in Metaheuristics for Hard Optimization 2008: 69-85
M. A. Tahir, J. Smith and Caleb-Solly, P (2008); **"A Novel Feature Selection based Semi-Supervised method for Image Classification"**. Pp 484-493, proceedings of 6th International Conference on Computer Vision Systems, Greece, 2008.
Pauplin, O., Caleb-Solly, P. and Smith, J.E. (2008): **"Interactive Parameter Adaptation Tool for Image Segmentation"**. Proceedings of IADIS Computer Graphics and Visualization 2008 (CGV 2008) Conference. Amsterdam, July 2008.
C. Eitzinger, M. Gmainer, W. Heidl, **A Framework for Adaptive Image Classification in Visual Inspection Tasks**, 32nd OAGM/AAPR Workshop, ISBN 978-3-85403-232-8, Linz, 26-27 May, 2008, pp. 159-168, 2008
D. Sannen, M. Nuttin, J. Smith, M. A. Tahir, P. Caleb-Solly, E. Lughofer, and C. Eitzinger, **An On-Line Interactive Self-Adaptive Image Classification Framework**, 6th International Conference on Computer

Vision Systems, Santorini, 12-15 May, 2008, A. Gasteratos, M. Vincze, and J.K. Tsotsos (Eds.): ICVS 2008, LNCS 5008, pp. 171–180, 2008.
C.Eitzinger, M.Gmainer, W.Heidl, E.Lughofer, **Increasing Classification Robustness with Adaptive Features**, 6th International Conference on Computer Vision Systems, Santorini, 12-15 May, 2008, A. Gasteratos, M. Vincze, and J.K. Tsotsos (Eds.): ICVS 2008, LNCS 5008, pp. 445–453, 2008
W. Reisner, C. Eitzinger, **"Automatische Gußteilprüfung mittels Bildverarbeitung"**, Giesserei Rundschau, vol. 54, no. 5/6, 2007, pp. 87-91
S. D. Seichter, C. Eitzinger, F. Breitenecker, **Model-Based Learning Classifiers for Surface Inspection Problems. Frontiers in simulation.** Simulationstechnique 19th Symposium in Hannover. Eds. Matthias Becker, Helena Szczerbicka. Society for Modeling and Simulation International. SCS Publishing House e.V. San Diego, Erlangen. 2006. ISBN 3-936150-49-4. Hannover, September 2006
E. Lughofer, **Process Safety Enhancements of Data-Driven Evolving Fuzzy Models**, In *Proceedings of the International Symposium on Evolving Fuzzy Systems (EFS)*, pp. 42-48, Lake District, UK, September 2006 **(awarded as best paper)**
E. Lughofer, P. Angelov and X. Zhou. **Evolving single- and multi-model fuzzy classifiers with FLEXFIS-CLASS**, *in Proceedings of FUZZ-IEEE 2007,* pp. 363-368, London, UK, 2007
E. Lughofer. **Extensions of Vector Quantization for Incremental Clustering**. *Pattern Recognition*, vol. 41(3), pp. 995-1011, 2008
E. Lughofer and S. Kindermann. **Improving the Robustness of Data-Driven Fuzzy Systems with Regularization**, in Proc. of the IEEE World Congress on Computational Intelligence (WCCI) 2008, Hongkong, pp. 703-709, 2008
E. Lughofer. **Towards Robust Evolving Fuzzy Systems**, book chapter in Evolving Intelligent Systems - Methodologies and Applications, editors: Plamen Angelov, Dimitar Filev and Nik Kasabov, John Wiley and Sons, to appear, 2008-09
E. Lughofer, S. Raiser, R. Richter, **Object Recognition in Deviation Images - A Comparison of Methods**, Technical Report FLLL-TR-06-1, 2006
S. Raiser, **Object Extraction with Clustering Methods in Industrial Machine Vision Applications,** Master Thesis at University Linz
D. Sannen, H. Van Brussel, M. Nuttin, **"Classifier fusion using discounted Dempster-Shafer combination"**, in: P. Perner (Ed.), Machine Learning and Data Mining in Pattern Recognition, 5th International Conference, MLDM 2007, Poster Proceedings, IBaI publishing, 2007, pp. 216–230.
D. Sannen, H. Van Brussel, M. Nuttin, **"Learning visual quality inspection from multiple humans using ensembles of classifiers"**, in: A. Gasteratos, M. Vincze, J. K. Tsotsos (Eds.), Computer Vision Systems, Vol. 5008 of LNCS, Springer, 2008, pp. 454-463.