Carpet Wear Classification Using Coocurrence Matrices and Support Vector Machines

S. A. Orjuela Vargas[1], C. Copot[2], S. Syafiie[2], E. Vansteenkiste[1], F. Rooms[1], W.Philips [1], R. de Keyser[2], and L.Van Langenhove[3]
[1]Department of Telecommunications and Information Processing (TELIN), Ghent University, Sint-Pietersnieuwstraat 41, B-9000 Gent, Belgium.
[2]Department of Electrical energy, Systems and Automation, Ghent University, Belgium
[3]Department of Textile, Ghent University, Belgium

***Abstract*—Nowadays, carpet industries have to comply with standard accreditations, requiring labels to certify both the quality and application of their products. Therefore, carpet companies are demanding a reliable, accurate and objective evaluation of the carpet wear level. This paper[1] presents a new approach to analyzing textures on carpet surface data taken using a 3D laser scanner which provides both structural and intensity information. Our aim is to develop and validate a strategy to classify changes in the surface texture on worn carpets. 3D data are resampled on different grid sizes in order to be able to apply 2D image techniques. The features we used are based on Haralick descriptors from the coocurrence matrix. This paper explains how an optimal experimental design is applied to identify the best combination of sizes and descriptors. Subsequently, the results of a support vector machine classification of the chosen descriptors are shown. With the methodology proposed, we achieve an average of over 90% correct (label) classification.**

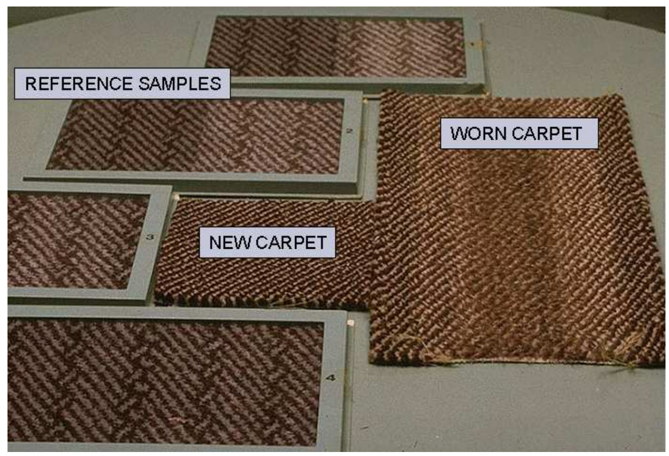


Fig. 1. Test environment used by human expert to classify wear carpets. Source: Textile Department, Ghent University

I. Introduction

In carpet quality evaluation facilities, the conservation of carpet aspects after common wear is one of the most important parameters. To make the appropriate carpet selections, depending of the application, usually labels are assigned using the Carpet and Rug Institute (CRI)'s Performance Standards [1]. This rating only takes into account how the carpet texture changed due to matting and crushing that might occur from walking on it. Years of wear are not associated with the performance ratings. To simulate these wear processes in an accelerated, controlled environment, several laboratory instruments are used, such as ”Tetrapod Walker Tester”, ”Hexapod Drum Tester” and ”Vettermann Drum Tester”[1].

There are five standard carpet quality classes, which range from class 1: maximum wear to 5: no wear. The changes in appearance in the carpet due to wear have been classified visually and subjectively by human experts by comparing them to a set of reference standard samples. Usually, at least 3 human experts are involved to classify the carpet level. Each of them may give his/her own grading level depending on a visual inspection and comparison of the examined carpet with the standard one. Each human expert could give different levels, so the class of the carpet can be in between standard levels. Therefore, the level scales are extended from integer levels to half levels as well. Consequently, human experts frequently disagree in their conclusion as their judgments can be affected by their mood, expertise and other external factors. Fig 1 shows a common line-up used by human experts to classify through visual inspection.

Consequently, carpet companies demand a more reliable, accurate and objective evaluation of the carpet wears level. In the last decades, researches to replace the human expert by automated evaluation methods have been an active topic in academic institutions and industry [2], [3], [4]. However, until now, human judgement is still the ”best” choice because no qualified automated systems for this task are available yet.

The textile department at Ghent University has been working on a project named COMPAS (Computer-based Assessment of Aspect Change in Carpets due to Wear), in order to develop an automated system that allows an objective [5], flexible, accurate, low cost and fast assessment of carpet wear. The goal is to deliver a prototype system that can be developed further for commercial use. Previous research has been focused on

[1]S. A. Orjuela Vargas is supported by a grant of the “LASPAU Academic and Professional programs for the Americas” in agreement with COLCIENCIAS and Universidad Antonio Nariño, Colombia.

either the CCD camera or the 3D laser scanner [5]. Some of the results have been applied successfully to a limited set of carpet samples.

A 3D laser scanner provides both structural (carpet surface) as well as intensity information. Previous research involving this scanner concentrated on depth information only and focused on finding a regression function between the depth and the wear The validation was performed using nested cross-validation, whereas in our paper the validation is done using using the input carpet labels [6].

Moreover, our study presents techniques of automated carpet wear classification of changes in carpet texture using the 3D laser scanner, pattern recognition approaches and SVMs. SVMs are a new type of pattern classifier based on a statistical learning technique that has been proposed by Vapnik and his co-workers [7], [8], [9]. The paper is organized as follows; a resampling technique from 3D laser data to a regular 2D image grid needed to perform the texture analysis is given in section II. Subsequently, texture feature extraction and carpet rating performances are discussed in section III and IV, respectively. Section V discusses the feature selection of our descriptor set. Finally, a SVM classification approach is presented in Section VI and final remarks are discussed in section VII.

II. 2D IMAGE CONSTRUCTION

During scanning, pieces of wood are used to keep the carpet flat. These pieces are also scanned and need to be removed from the dataset (see Fig. 4 - and comment on the white holes in the carpet there in the caption of the figure). Therefore, 21 random carpets were analysed to define two depth thresholds -112 m, -99 m to separate the pieces of wood from the carpet: scanned carpets are typically between these thresholds, scanned wood is not. The 3D laser scanner produces measurements on a random grid, so these 3D data need to be resampled into a regular grid to construct 2D images of them. The way we perform this directly influences the image resolution we will work on.

We empirically found that a sampling of 10×10 cells per square centimetre contains at most two points. In this case, the number of cells containing sample points is on average 30% of the total number of cells. In order to increase this percentage, we also tested different sample schemes (5×5 cells and 2×2 cells per cm2). Examples of the three schemes are shown in Fig 2.

Next, two resampling methods were used: the first estimates the data in each cell by using the samples in the cells and its 8 neighbours are averaged. In the second, only the sample points in the cell itself are averaged, see Fig 3.

Per data cell,the scanner provides 4 types of measurements: the first two are the mean depth ”Z-mean” and median depth ”Z-median”. The third one is the median intensity value (named ”Q”), and the last one is the density ”D”, which represents the number of points either per cell. As a result there are 24 images per carpet to analyse (4 measurement values; each time for both resample techniques and that for 3 different resample sizes: 2×2, 5×5 and 10×10 cells per

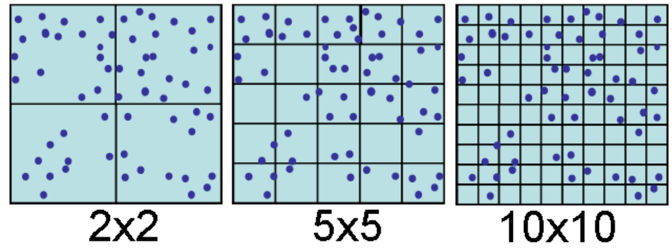


Fig. 2. Example of schemes for resampling 3D laser data

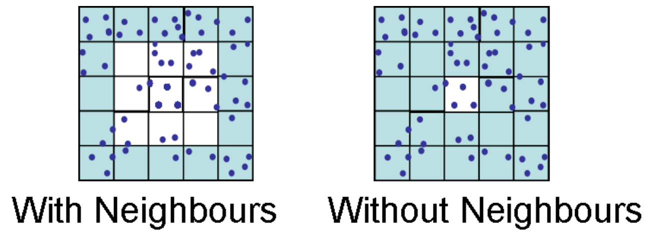


Fig. 3. Filling type methods used for estimate data in cells

cm2). A set of 10 different carpets types was used in this experiment A8-501, A8-701, BIG4, 20 KL 803 Beige, LA7, BIG8 Beige, 517, BIG8 Dark, LA9, 20 KL 803 Dark, one particular example is shown in Fig 4.

III. FEATURE DESCRIPTORS

Due to the characteristics of the scanner, the concentration of sample points is higher in the middle of a scan strip than at the borders. Due to the mechanical acceleration wear setup, the wear is concentrated on the middle of the X-axis of the carpet, so the amount of wear typically is only a function of the Y-direction. Therefore, each scan line can be used as a replica for classification with a support vector machine, as will be explained in section VI. How we represent the scan lines, is explained next.

First, we extract the regions around the scanned lines (holes are removed with morphological operators). Next, each scan strip is divided into 48 equal regions and coocurrence matrices [10] are calculated for each (Resample, Filling, Measurement) combination. Subsequently, a window of 5 coocurrence matrices is defined and moved along the X-axis. Then, 14 Haralick

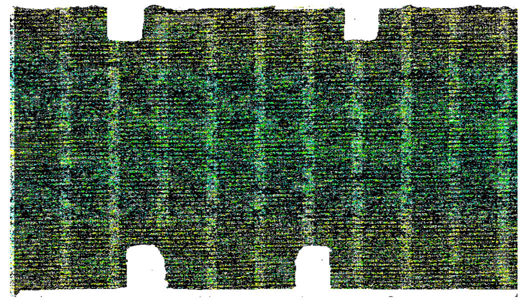


Fig. 4. Representative image of subset Carpet Type, Measurement, Resample Size, Filling type = BIG4 , Z-Median, 5x5, without.

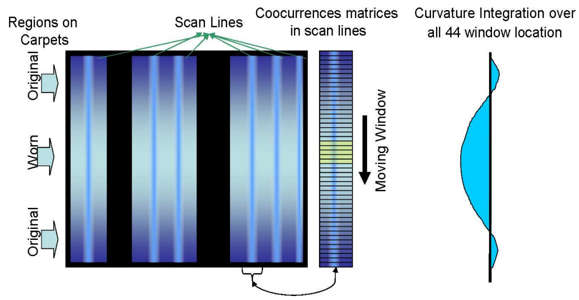


Fig. 5. Computation of Haralick descriptors on scan lines.

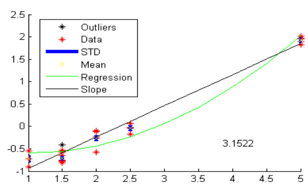


Fig. 6. Carpet Rating Performance (CRP) for subset Carpet, Measurement, Resample Size, Filling type = 701, Z-Median, 5x5, without.

descriptors [11], [12] (per combination) are computed for each window location resulting in a set of 24×14×44 descriptors per scan line. This technique is illustrated in Fig 5.

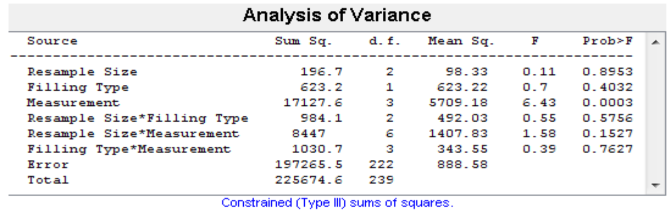
Finally, the curvature obtained from plotting each of the 24×14 descriptors is computed integrating over all 44 window location. This technique is illustrated in Fig 5.

IV. CARPET RATINGS PERFORMANCE (CRP) CRITERION

To unambiguously characterize the status of a given carpet, the estimated labels must be a monotonic function (either increasing or decreasing) of the descriptors. To verify this, the following error criterion is computed.

For each replica (Carpet Type, Measurement, Filling Type, Resample Size), we have a set of 14 descriptors which by Principal Component Analysis (PCA) can be reduced to 3 descriptors (the 3D projection of the 14D space still contains 95% of variation) [13]. Thus, per label, we obtain a 3D cluster of sample points (PC1, PC2, PC3). We then compute the Mahalanobis distance between the current cluster and the total set of all other clusters [14], [15].

We assume that in the ideal case, the relationship between the labels and the Mahalanobis distances would be linear with a slope of about 45o (i.e. resulting in a linear 1-1 relationship). We first fit a second order regression function (constrained to be either monotonically increasing or decreasing) to the mean values per label over all tested carpets. As a performance measure for our carpet rating method, we computed the ratio between the square error of the regression function of sin(2 + /4), where is the slope of the first order regression line (this function is 0 for 0 and 90 and has a maximum at 45). The lower this ratio, the better the performance of our method is. An example is shown in Fig 6.



Analysis of Variance

Source	Sum Sq.	d.f.	Mean Sq.	F	Prob>F
Resample Size	196.7	2	98.33	0.11	0.8953
Filling Type	623.2	1	623.22	0.7	0.4032
Measurement	17127.6	3	5709.18	6.43	0.0003
Resample Size*Filling Type	984.1	2	492.03	0.55	0.5756
Resample Size*Measurement	8447	6	1407.83	1.58	0.1527
Filling Type*Measurement	1030.7	3	343.55	0.39	0.7627
Error	197265.5	222	888.58		
Total	225674.6	239			

Constrained (Type III) sums of squares.

Fig. 7. Result of the analysis of variance for the experiment.

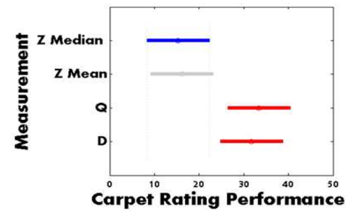


Fig. 8. Comparison of Measurement data.

V. FEATURE SELECTION

The performance measure we described in the previous section is used to compare combinations of {Carpets Type, Measurement, Filling Type, Resample Size} [15].

From Fig. 7, we noticed that for a significance level of 0.05, there are significant differences on the responses of the Measurements. So, we focused on a comparison of the measurements.

From Fig. 8, it can be seen that the Z-median and the Z-mean scores have means significantly different from Q and D values. Consequently, further comparisons are done using only Z-median.

InThe same procedure shown in Fig. 8 was used for filling type and resampling size, resulting in no significant differences between filling and no filling and neither between the three different resample sizes. Therefore, our choice is based on minimal means factor for Filling Type (Without), and resample size 55 because it has minimal changes comparing with the original optimal resample size 10×10.

TABLE I
DESIGN FOR ANALYSIS OF VARIANCE

Carpet Type	Measurement	Filling Type	Resampling Size	Replicas	CRP
1	1	1	1	1	X
1	1	1	1	2	X
1	1	1	1	3	X
1	1	1	1	4	X
1	1	1	1	5	X
⋮	⋮	⋮	⋮	⋮	⋮
1	1	1	1	5	X

VI. CLASSIFICATION

In this study Support Vector Machine (SVM) is used to classify the provided data. A SVM is a maximal margin hyperplane in feature space built by using a kernel function in gene space. When SVM is used for classification, the vectors separate a given known set of the training data via a hyperplane that is a maximum distance from the one sample class to other sample class. This hypeplane is known as optimal separating hyperplane. Then, the the test data can be plotted at the high dimensional space, to distinguish whether the data belong a class according to the hyperplane. Most of problems facing in a real-world are not linearly separable, therefore, a given data that linearly separable can be directly applied to classify. However, kernel function can be used to map a given data from input space to feature space, then the classification is done using SVM in the feature space corresponding to a non-linear decision boundary in the input space.

In this paper, two different kernel SVM functions, polynomial and Gaussian kernels, are used to analyze the performance of the classification. For both cases, the classifier is individually trained to 10 types of carpets. Each carpet has a different class distribution. The three first principal components from the chosen combination are used for the training input. Every class of the carpet has 5 replicas, of which 4 replicas are used for training and 1 replica is used for testing.

SVMs aim at minimizing an upper bound of the generalization error through maximizing the margin between the separating hyperplane and the data [16]. SVMs are known to generalize well even in high dimensional spaces under small training sample conditions [17]. Therefore, SVM is applied in this study.

A. Linearly Separable

let the training data of two separable classes with n samples be represented by $(\vec{x}_1, y_1), (\vec{x}_2, y_2), \cdots, (\vec{x}_n, y_n)$, where $\vec{x}_i \in R^N$ is an N dimensional space, and $y_i \in \{-1, +1\}$ is the class index. Taking a weight vector $\vec{w}$ and bias b, the two classes can be separated by two margins parallel to hyperplane, that is

$$\begin{array}{l} \vec{w} \cdot \vec{x}_i + b \geq 1, \quad y_i = +1, \\ \vec{w} \cdot \vec{x}_i + b \leq -1, \quad y_i = -1, \end{array} \tag{1}$$

where $\vec{w}$ is a vector of n elements. By multiple the left hand side, the equation can be written as

$$y_i(\vec{w}^T \cdot \vec{x}_i + b) \geq 1, \quad i = 1, 2, \cdots, n. \tag{2}$$

Here, the objective of finding the optimal hyperplane is to determine an optimal weight and bias. For any given hyperplane, the equation can be written as:

$$\vec{w}^T \cdot \vec{x}_i + b = 0, \tag{3}$$

and the distance between the two corresponding margins is

$$d(\vec{w}, b) = \min_{\{\vec{x}|y=+1\}} \frac{\vec{x}^T \cdot \vec{w}}{\|\vec{w}\|} - \max_{\{\vec{x}|y=-1\}} \frac{\vec{x}^T \cdot \vec{w}}{\|\vec{w}\|}. \tag{4}$$

The optimal separable hyperplane can be found by maximazing the above distance or by minimazing the norm of $\|\vec{w}\|$ under inequality constraints (2), and gives

$$d_{max} = d(\vec{w}_0, b_0) = \frac{2}{\|\vec{w}\|}. \tag{5}$$

The optimal solution of the problem can be solved using the saddle point of the langrangean, that is

$$L(\vec{w}, b, \alpha) = \frac{1}{2}\vec{w}^T \cdot \vec{w} - \sum_{i=1}^{n} \alpha_i \left[y_i(\vec{w}^T \cdot \vec{x}_i + b) - 1 \right], \tag{6}$$

where $\alpha_i \geq 0$ are Lagrange multipliers. By taking the derivative (6) with respect to $\vec{w}$ and b iqual to zero gives

$$\begin{array}{l} \vec{w} = \sum_{i=1}^{n} \alpha_i y_i \vec{x}_i, \\ \sum_{i=1}^{n} \alpha_i y_i = 0. \end{array} \tag{7}$$

By substituting (7) to (6), the equation becomes the maximization of the following equation

$$L(\propto) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j (\vec{x}_i^T \cdot \vec{x}_j) \tag{8}$$

subject to $\sum_{i=1}^{n} \alpha_i y_i = 0$ and $\alpha_i \geq 0, \ i = 1, 2, \ldots, n.$

The support vector is that the point situated at the two optimal margins, which have non-zero coefficients, α_i, amoung the solution to equation (8). The bias can be calculated as follows

$$b = -\frac{1}{2} \left(\min_{\vec{x}_i|y_i=+1} \vec{w}^T \cdot \vec{x}_i + \max_{\vec{x}_i|y_i=-1} \vec{w}^T \cdot \vec{x}_i \right). \tag{9}$$

After finding optimal support vector and bias, the decision function that separates the two classes can be written as

$$f(\vec{x}) = sign \left[\sum_{i=1}^{n} \alpha_i y_i \vec{x}_i^T \cdot \vec{x} + b \right] \tag{10}$$

B. Non-linear separable

In nonlinear separable data, the original training data $\vec{x}$ in the input space X is mapping into a high dimensional feature space F via a Mercer kernel operator K, and the optimal hyperplane is constracted in this feature space. The equation (10) can be rewritted as

$$f(\vec{x}) = sign \left[\sum_{i=1}^{n} \alpha_i y_i K(\vec{x}_i, \vec{x}_i) + b \right] \tag{11}$$

where K is a symetric positive definite function, which satisfies Mercer's conditions,

$$\begin{array}{l} K(\vec{x}, \vec{y}) = \sum_{m=1}^{\infty} \alpha_m \phi(\vec{x})^T \cdot \phi(\vec{y}), \alpha_m \geq 0 \\ \int\int K(\vec{x}, \vec{y}) g(\vec{x}) g(\vec{y}) d\vec{x} d\vec{y} > 0, \int g^2(\vec{x}) d\vec{x} < \infty. \end{array} \tag{12}$$

The kernel represents a legitimate inner product in input space

$$K(\vec{x}, \vec{y}) = \phi(\vec{x})^T \cdot \phi(\vec{y}). \tag{13}$$

In Feature space, the dual Langrangian of (6) can be written as

$$L(\propto) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(\vec{x}_i, \vec{x}_j) - \lambda \sum_{i=1}^{n} \alpha_i y_i \tag{14}$$

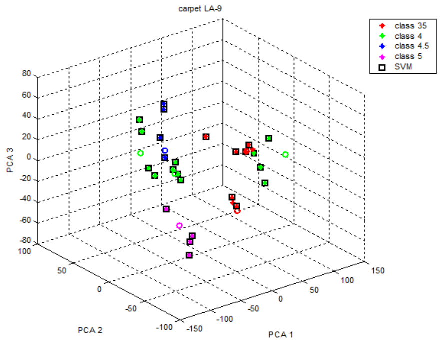


Fig. 9. Training SVM with Gaussian kernel for carpet LA-9.

subject to $\sum_{i=1}^{n} \alpha_i y_i = 0$ and $\alpha_i \geq 0,\ i = 1, 2, \ldots, n$. The bias in F-space can be calculated as

$$b = -\frac{1}{2}\left(\min_{\vec{x}_i|y_i=+1}\left(\sum_{j\in\{SV\}} \alpha_j y_j K(\vec{x}_i, \vec{x}_j)\right) + \max_{\vec{x}_i|y_i=-1}\left(\sum_{j\in\{SV\}} \alpha_j y_j K(\vec{x}_i, \vec{x}_j)\right)\right), \quad (15)$$

so that, the decision function is

$$f(\vec{x}) = sign\left[\sum_{i=1}^{n} \alpha_i y_i K(\vec{x}_i, \vec{x}) + b\right]. \quad (16)$$

The polynomial and gaussian kernels used in this study are

$$K(\vec{x}, \vec{y}) = (1 + \vec{x} \cdot \vec{y})^d, \quad (17)$$

$$K(\vec{x}, \vec{y}) = exp(-\frac{\|\vec{x} - \vec{y}\|^2}{2\sigma^2}), \quad (18)$$

respectively.

C. SVM classification performance

An example-training SVM is plotted in Fig. 9 for carpet LA-9, This LA-9 carpet has 4 classes (classes 3.5, 4, 4.5 and 5) plotted in different colors. The SVM classifier is built with these 4 classes of data for training. The trainging data are the first 3 PC. Class 3.5 has 8 training data and 2 for testing points, class 4 has 12 training data and leave 3 data for testing. For classes 4.5 and 5 each has 4 training data and 1 for testing. Similar available training data for other types of carpet are used in finding the support vector and optimal hyperplane.

The classifier is built by using both kernel functions. For this specific example of the carpet LA-9, classifier by using a polynomial kernel, gives the machine to classify 72% correctly of the given testing data. However, by using a Gaussian kernel, the machine classified 100% of testing data correctly. For several cases (see Table II), the kernel function can not help to increase the performance of the machine. For example carpet A8 - 501, 20 KL 803 Beige and Big 8 Dark, both polynomial and Gaussian kernels give the same classification. Overall, the machine classifies 89.56% correctly using a polynomial kernel and 95.48% using a Gaussian kernel. As the previus approach with 3D laser data was studied on a regresssion fitting between depth and labels. Therefore, it is not possible to compare our results with that one.

TABLE II
OVERALL CLASSIFICATION USING 2 KERNELS FUNCTIONS

Type of carpet	Polynomial kernel (%)	Gaussian kernel (%)
A8 - 501	84.4	84.4
A8 - 701	84.4	100
Big 4	100	100
20 KL 803 Beige	84.4	84.4
LA 7	84.4	100
Big 8 Beige	100	100
20 KL 517	100	100
Big 8 Dark	86.0	86.0
LA 9	72.0	100
20 KL 803 Dark	100	100
Over all	89.56	95.48

VII. Conclusion

In this paper we presented a new methodology for classifying appearance changes on carpet texture from 3D laser scanner data. By using the CRP criterion we proposed in Section IV, we have shown that this methodology is well-suited for dealing with multiple types of carpets. In this paper, we only applied our methodology to texture descriptors based on the coocurrence matrix, but we believe that our methodology can be used in a more general context to investigate relations between a mathematical description of carpet observation (descriptors based on depth and texture) and carpet quality to achieve a generic automated carpet wear evaluation system.

References

[1] CRI Test Method - 101, Technical Bulletin, *The Carpet and Rug Institute*, Revision July 2003.
[2] Wood, E., Hofgson, R., ”Carpet texture measurement using image analysis”, Textile Research Journal 59, 1-12, 1989.
[3] Pouderhimi, B., Xu, B., Nayernouri. A., ”Evaluation carpet appearance loss: pile lay orientation”, Textile Research Journal 64, 130-135, 1994.
[4] Van Steenlandt, W., Collet, D., Sette,S., Bernarn,P., Luning, R., Teze, L., Bohland, H., Schulz, H., ”Automatic assesment of carpet wear using image analysis and neural networks”, Textile Research Journal 66, 55-561,1996.
[5] Department of Textiles, Faculty of engineering, Ghent University. Annual Report 2007.
[6] Waegeman, W., Cottyn, J., Wyns B., Boullart, L., De Baets, B., Van Langenhove, L., Detand, J. ”Classifying Carpets based on Laser Scanner data”. Engineering Applications of Artificial Intelligence, Volume 21, Issue 6, September 2008, Pages 907-918.
[7] Boser, B., Guyon, I., Vapnik, V. ”A training algorithm for optimal margin classifiers”, In Proceedings of Fifth Annual Workshop on Computational Learning Theory, New York, (1992).
[8] Cortes, C., Vapnik, V. ”Support vector networks, In Proceedings of Machine Learning”, vol. 20, pp. 273-297, (1995).
[9] Vapnik,V. ”The nature of statistical learning theory”, Springer, (1995).
[10] Haralick, R,. Shanmugam, K., Dinstein, Its’Hak. ”Textural Features for Image Classification.” IEEE Transactions on Systems, Man, and Cybernetics. Vol 3 nr 6, Nov. 1973 pp 610–621.
[11] Haralick R., (May 1979). Statistical and structural approaches to texture, Proceedings of the IEEE, vol. 67, No.5, pp. 786-804.
[12] Miyamoto E., T. Merryman,” Fast Calculation of Haralick Texture Features”, Technical Report, Carnegie Mellon University, www.ece.cmu.edu/ pueschel/teaching/18-799B-CMU-spring05/ material/ eizan-tad.pdf, (accessed 2008).
[13] Jolliffe I. T., Principal Component analysis, Second Edition, Springer, Chapters 2,4 , 2002.
[14] Bar-Hillel Aharon, Tomer Hertz, Noam Shental, Daphna Weinshall ”Learning a Mahalanobis Metric from Equivalence Constraints”, Journal of Machine Learning Research 6, Published June 2005
[15] Montgomery, Douglas C. Design and analysis of Experiments, 5th Edition, John Wiley & Sons, INS, chapters, 1-3,5, 2001.

[16] Amari, S., Wu, S. ”Improving support vector machine classifiers by modifying kernel functions”, In Proceedings of International Conference on Neural Net- works, 12, pp. 783-789, (1999).
[17] Jonsson, K., Kittler, J., Matas, Y.P. ”Support vector machines for face authentication”, Journal of Image and Vision Computing, vol. 20. pp. 369-375, (2002).