Journal of Ambient Intelligence and Humanized Computing (2021) 12:9269–9285
https://doi.org/10.1007/s12652-020-02631-6

ORIGINAL RESEARCH

Automated leather defect inspection using statistical approach on image intensity

Y. S. Gan[1] · Sue-Sien Chee[2] · Yen-Chang Huang[3] · Sze-Teng Liong[4] · Wei-Chuen Yau[2]

Received: 15 April 2020 / Accepted: 24 October 2020 / Published online: 20 November 2020

Abstract
Leather is a very important raw material in many manufacturing industries. For example to produce footwear, garments, bags and accessories. Prior to the mass production of certain product, a professional leather visual inspection process for defection spotting is essential as the quality control step. However, to date, there is a lack of fully-automated leather inspection systems in the industry, whereby most manufacturers rely on experienced and trained experts to mark out the defects in the leather. This kind of human assessment work is inefficient and inconsistent. Therefore, this paper proposes a method that based on image processing techniques, namely, gray level histogram analysis, to detect defects of the leather. Specifically, the histogram characteristics such as the mean and standard deviation are extracted and treated as the features. Then, the statistical Kolmogorov–Smirnov's two-sample test is utilized to perform feature selection. Followed by a thresholding method to reduce the dimensionality of the features. Finally, the features are categorized by several well-known classifiers. The best classification accuracy obtained are 99.16% and 77.13% on two different datasets respectively.

Keywords Leather · Defect · Statistical · Classification · Feature selection

1 Introduction

Leather is a material made from animal hide that has been treated with chemicals to preserve them and make them suitable for use as clothing, handbags, sports equipment, furniture, footwear, and tools. A recent 2018 study reveals that the world's leather goods market is at 95.4 billion USD, and it is forecasted to reach 128.61 billion USD by 2022, with a rate of growth of 4.36% (APLF 2018). Because of the importance of leather in the manufacturing industry, it is vital to ensure the good quality of the leather to improve customer satisfaction.

Normally, a natural leather piece may contain imperfections like insect bites, cuts, stains and wrinkles, as illustrated in Fig. 1. To produce leather products of high quality, these defects must be identified and removed during the defect determination process. In brief, the quality of leather can be judged based on established quality standards such as the standard listed by SATRA (Technology 2020). SATRA advocates a five-point leather grading system that assigns a specific band to a leather piece which expresses the percentage of usable surface (i.e., not marked by defects) of the leather (Table 1).

The defect identification process is generally acknowledged as one of the most time-consuming aspects in production process that involve leather material. At present, it is still a common practice to perform this manual defect spotting process by trained inspectors. During the defect inspection process, the inspectors are required to use various

✉ Sze-Teng Liong
stliong@fcu.edu.tw

✉ Wei-Chuen Yau
wcyau@xmu.edu.my

Y. S. Gan
ysgan@fcu.edu.tw

Sue-Sien Chee
swe1609507@xmu.edu.my

Yen-Chang Huang
ychuang@mail.nutn.edu.tw

[1] School of Architecture, Feng Chia University, Taichung, Taiwan

[2] School of Electrical and Computer Engineering, Xiamen University Malaysia, Sepang, Malaysia

[3] Department of Applied Mathematics, National University of Tainan, Tainan, Taiwan

[4] Department of Electronic Engineering, Feng Chia University, Taichung, Taiwan

(a) Scratch

(b) Wrinkle

(c) Stretch marks

Fig. 1 Typical flaws seen on leather skins

Table 1 SATRA's five-point grading system for leather (Technology 2020)

Quality coefficient	Grade	Representative coefficient
100 to 95.1	A	97%
95 to 90.1	B	93%
90 to 85.1	C	88%
85 to 80.1	D	83%
80 to 75.1	E	78%
75 to 70	F	73%

angles and distances to examine the same section of leather multiple times (Liong et al. 2019a). Qualified inspectors that have shown consistency and competency in the standard and procedures will be awarded individual certificate.

Current solutions are to mark the defect areas manually and then grade the defects based on the level of damage (i.e., minor, major, severe, etc.) and the defects forms (i.e., bruises, tick bites, wrinkles, scabies, etc.) (Liong et al. 2019a). Typical defects marking systems work by first scanning and automatically detecting the leather boundary. Trained professionals will then use an electronic pen to mark the defects on the leather and grade the defects accordingly. The positions and gradings of these markings are recorded by the defects marking system and will be used in the leather cutting process in the later stages.

Since inspection process relies on a human expert, it is costly, time consuming, and subjective. Moreover, it is always prone to human error because it requires a high degree of concentration which may lead to fatigue. Therefore, to eliminate manual intervention in this specific process, there is a need to establish an automatic vision-based solution (Aslam et al. 2019). Thus far, there are plenty of important applications of image processing in the quality inspection field (Xie et al. 2018; Aslam et al. 2020). Image processing is currently being used in the grading and sorting of products such as fish and fruits and even recycled plastics and timber products (Blum 2018). It is an important step in automating the entire process of inspection, thus increases the productivity (San-Payo et al. 2019).

This paper attempts to propose an automatic leather inspection system that utilizes a series of simple yet powerful image processing techniques. Succinctly, the feature extraction process is based on the analysis of the histogram of grey levels of the leather images. In brief, histogram provides rich information about the surface of the leather, especially those details that contain irregularities on the surface. It is important to know which information is pertinent or useful in detecting the presence of certain kinds of defects. Therefore, this paper explores how the most relevant histogram features can be chosen, which are then used for classification of the leather images. With the above definitions and observations, the main objectives in this paper are summarized as follows:

1. Proposal of a feature selection and extraction method based on gray level histogram analysis.
2. Evaluation of the proposed algorithm on multiple machine learning classifiers in order to verify the adaptability of the algorithm.
3. Detailed quantitative analysis of the results is provided by exploiting multiple metrics.
4. Demonstration of the versatility of the algorithm by applying it to two distinct datasets that have different nature and characteristics.

2 Literature review

Existing literature suggested many vast and varied methods in classifying defects on different surface types. Most of them apply some variation of statistical, spectral, or edge-detection approaches (Sundari 2017). Approaches such as Gray Level Co-occurence Matrix (GLCM) (Mohanaiah et al. 2013), edge-detection (Kasi et al. 2014), morphological operations (Kwak et al. 2001), Gabor filters (Hu 2015), etc. have been proposed and used in detecting leather defects. Nevertheless, the approaches proposed each have their own merits and weaknesses. For example, edge detection methods are excellent at differentiating defects that have a clear edges from the background area, but is less effective when the defects are characterised by a gradual change in texture. On the other hand, owing to the advances of deep learning architecture, research in computer vision has been booming in recent years (Rajagopalan et al. 2020; Zavala-Mondragon et al. 2019). Deep learning networks such as convolutional neural network (CNN) have recently been utilised to solve different type of problems. Many variants of CNN were also proposed, such as AlexNet, VGG, NIN, Inceptions, Inception-Resnet and DenseNet (Krizhevsky et al. 2012; Simonyan and Zisserman 2014; Lin et al. 2013; Szegedy et al. 2015, 2017; Huang et al. 2017). Following subsections review both the traditional image processing and deep learning methods that had been employed in automatic leather inspection research works.

2.1 Traditional image processing method

One of the early works is proposed by Georgieva et al. (2003) that employed χ^2 criteria to analyze the leather surfaces. The author compares defective areas on the leather surface to an averaged histogram of non-defect leather

samples. The application of the χ^2 criteria is then used for distance computing in relation to the averaged histogram. However, the author did not provide any concrete evidence of the effectiveness of this approach.

Pereira et al. (2018) compares the performance of combinations of different algorithms for feature extraction and classification of goat leather. The feature extractors such as Gray Level Co-occurence Matrix (GLCM) (Haralick et al. 1973), Local Binary Pattern (LBP) (Ahonen et al. 2006) and (Pixel Intensity Analyzer) PIA are employed and evaluated on many different classifiers such as e *k*-nearest neighbors (kNN), support vector machines (SVM) and minimal learning machine (MLM) (Cover and Hart 1967; de Souza Júnior et al. 2015). The results of this paper (Pereira et al. 2018) show that the combination of PIA feature descriptor and ELM classifier emerges the most cost-effective method to the classification problem. The drawback is that the author did not describe the setup in configuring the experiments.

He et al. (2006) outline a method to detect leather defects using wavelet band selection procedure. The authors aim to eliminate all regular and repetitive texture patterns by decreasing the resolution levels of the leather images until the texture patterns are undetectable. The selection of decomposition levels can be realized by adopting the wavelet band selection procedure. This is conducted by using the energy function and lowering the resolution until the energy ratio (ratio of detail between resolution levels) is minimum. Besides, this paper claims that this method can achieve the same detection percentage as a trained professional and is capable to implement in real-time inspection. However, no quantitative results were given and reported in the paper.

On the other hand, Hu (2015) introduces an elliptical Gabor filters (EGF) to analyze the patterns of leather surface with very specific orientation properties. During the training process, the optimized elliptical Gabor Filter is trained using genetic algorithm (Holland 1992). In the inspection process, the selected filter is convoluted with each sample under inspection, followed by a gray level thresholding process to generate a binary segmented result. The results show that the EGF outperforms ordinary Gabor filters in terms of detection accuracy.

Bong et al. (2018) proposes a 6-step inspection method for leather defect detection and classification. Several image processing algorithms are employed as the feature extraction step and to identify the position of the defect. The extracted defect features include color moments, color, correlograms, Zernike moments, and texture features. The accuracy obtained when evaluating the proposed method using SVM classifier is 98.8% for detecting one type of defect. An accuracy of 92.4% is attained in the case of two different types of defects. However, the results in the case of three or more defects were not shown.

2.2 Deep learning method

A supervised neural network is adopted in (Villar et al. 2011) to distinguish defective and non-defective leather images. Prior to that, the feature extraction, a Sequential Forward Selection method is performed to identify the meaningful features. Particularly, the type of defects considered in the experiment is open cut, closed cut, and fly bite. The average accuracy achieved is remarkable, viz, ~ 96.5%. However, the defective images used in the experiment is relatively obvious and hence leading to high classification result.

A similar approach is proposed by Jian et al. (2010) to identify the leather defects such as knife hole, nicks, imprint injury, and aberration. The overall accuracy attained is more than 92% when tested on a total amount of 200 images. However, the sample image used in the experiment did not show on the paper and did not release to the public. Therefore, it is ambiguous about the difficulty in classifying the defects.

A recent work presented by Liong et al. (2019b) is to utilize Mask-RCNN to perform defect segmentation to spot small tick-bite defects on leather surface. Besides, the same research group (Liong et al. 2019c) proposes two neural network approaches to extract the rich image features of the leather images: Artificial Neural Network (ANN) and Convolutional Neural Network (CNN). For CNN, a pre-trained neural network (i.e., AlexNet) is used. The network's architecture is comprised of five types of operations: convolution, ReLU, pooling, fully connected and dropout. The best performance exhibited by using ANN is 80%, whereas CNN achieves classification accuracy of 76.2%. It should be noted that this paper mentions that the dataset used suffers from a severe imbalanced class distribution problem.

The same group of authors Liong et al. (2020) extends the leather classification work by applying the AlexNet network architecture on other type of leather that consists of black line and wrinkle defects. The accuracy result obtained for the three-category classification is 94.67% when employing a train/test split of 80/20. In addition, a segmentation task is performed for both of the defects. As a result, the overall mean pixel accuracy (mPA) achieved is 88.34%. The performance of wrinkle defect is relatively lower than that of black line defect. This is because the appearance of wrinkle defect is not so obvious compared to that of black line defect.

In short, deep learning can greatly improve the accuracy of classification (up to 95% (Liong et al. 2020)). However, training a deep learning network requires higher computing power and a longer processing time. In contrast, histogram analysis, the heuristical conventional image processing method is useful because it is intuitive, easy to visualise and understand. Therefore, this paper will adopt this histogram analysis approach in the feature extraction process to classify the leather defects.

3 Dataset

There are two datasets involved in this experiment. Both of the datasets were elicited using a six-axis robotic arm, DRV70L from Delta, and a Canon EOS 77D camera. The leather is placed on a table and the robot arm - with the camera mounted on it - travels the length of the leather capturing images of the leather from the top-down viewpoint. A professional lighting source was used to provide consistent and continuous source of illumination. The light source was placed and fixed at 45 degrees from the leather, aiming downward (Liong et al. 2019b).

3.1 Dataset I

The first dataset contains several types of defects, but we will mainly focusing on the dark lines defects. These defects are more obvious and can be easily spotted with the naked eye. This dataset contains 199 defective images and 199 non-defective images. The analysis of the dataset in Table 2 reveals that the width of the dark line defects range from ~ 7 pixel up to ~ 80 pixel with an average of ~ 20 pixel. This information is useful during the image segmentation to determine the best segment size. The example of the images collected are shown in Fig. 2.

3.2 Dataset II

The second dataset used is the same as (Liong et al. 2019c). In brief, it contains leather samples that have circular tick bite-like defects. As seen in Table 3, these defects vary greatly in area, from 30 pixel^2 to 3195 pixel^2. The average area of the defects is 480 pixel^2. These defects are quite difficult to be detected even through human inspection. This dataset contains 503 defective images and 1102 non-defective images. The example of the images collected are illustrated in Fig. 3.

Table 2 Statistical analysis for the black line defects on the leather patches in Dataset I

	Width (pixel)	Area (pixel^2)	Area (mm^2)
Minimum	7.32	53.58	0.0754
First quartile	13.41	179.83	0.2529
Median	19.06	363.28	0.5109
Third quartile	25.16	633.03	0.8902
Maximum	80.63	6501.2	9.1423
Mean	20.9	436.81	0.6143
Standard deviation	9.79	95.84	0.1348

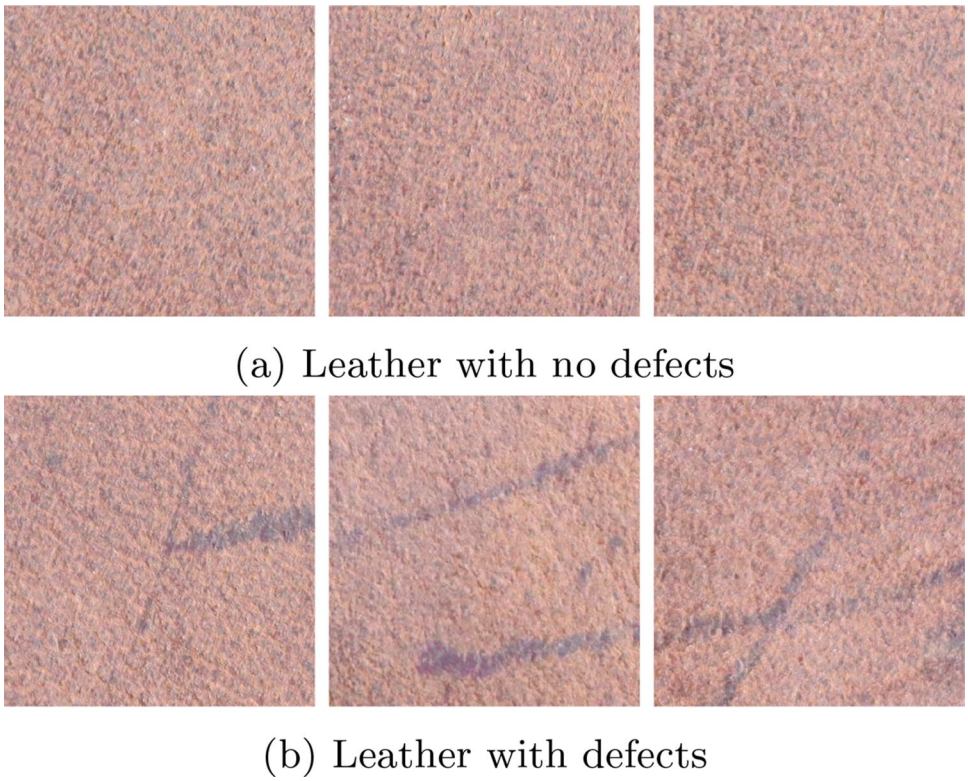


Fig. 2 Example of defective and non-defective leather images in Dataset I

Table 3 Statistical analysis for the tick bite defects on the leather patches in Dataset II

	x-axis (pixel)	y-axis (pixel)	Area (pixel^2)	Area (mm^2)
Minimum	6	5	30	0.0422
First quartile	16	16	272	0.3825
Median	20	20	396	0.5569
Third quartile	26	25	575	0.8086
Maximum	65	71	3195	4.4930
Mean	21.56	20.86	480.44	0.6756
Standard deviation	8.22	7.58	347.29	0.4884

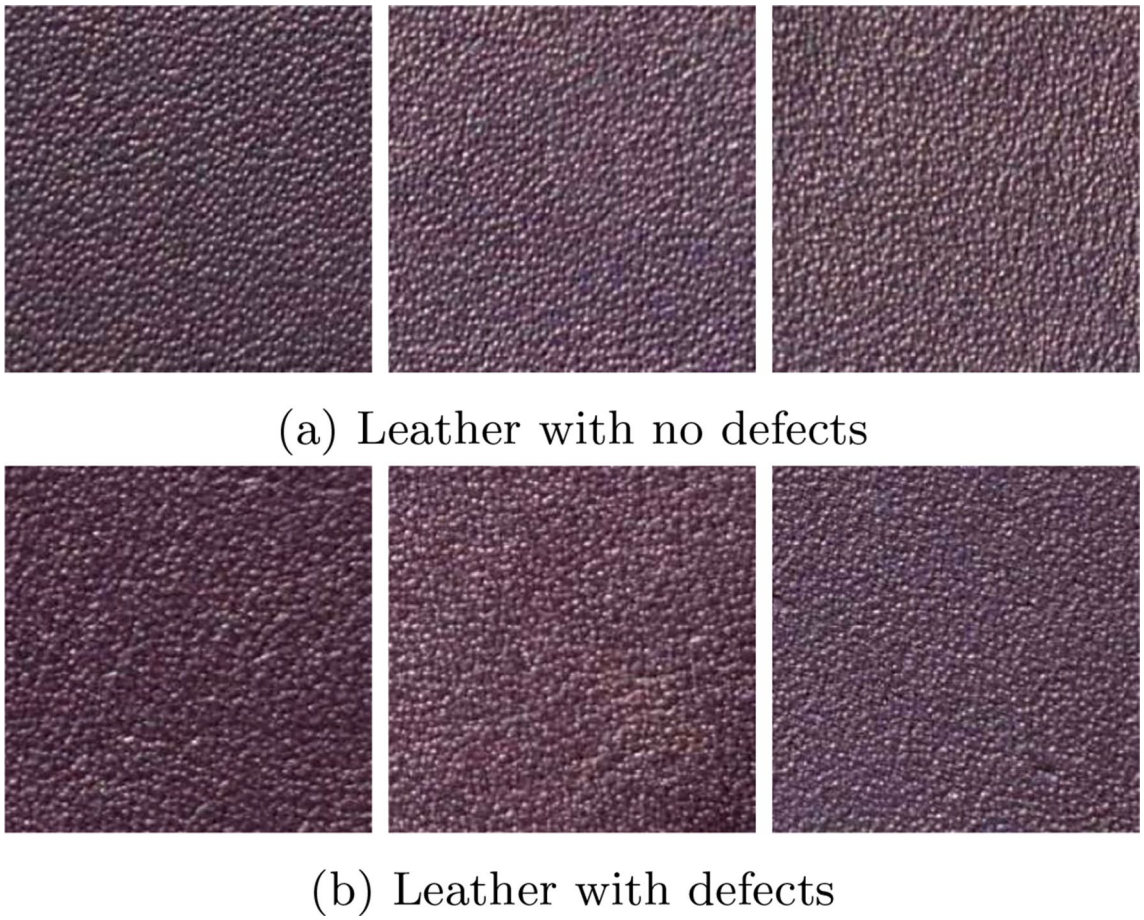
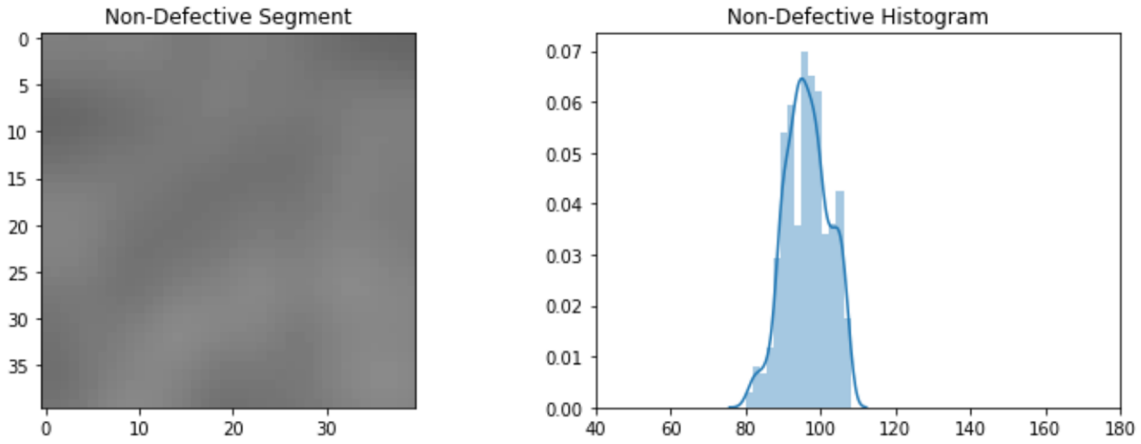


Fig. 3 Example of defective and non-defective leather images in Dataset II

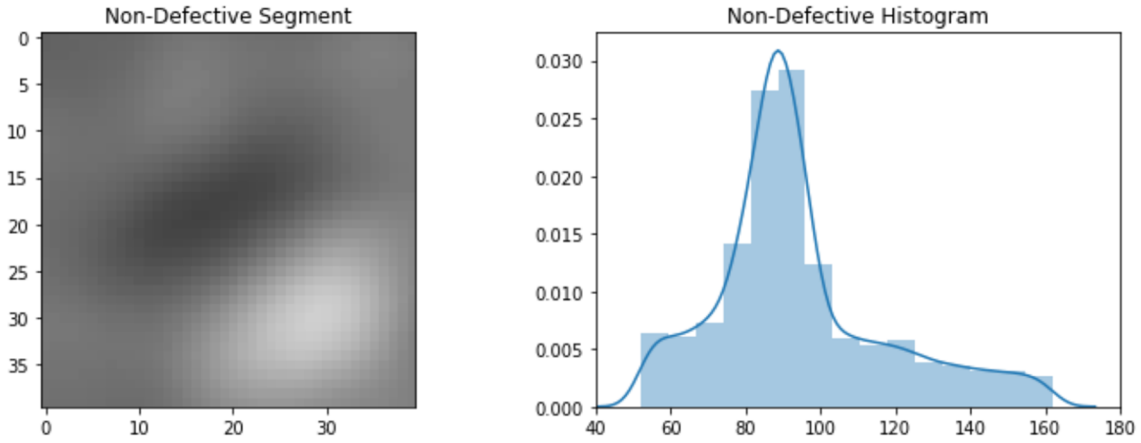
4 Proposed method

Figure 4 shows the histogram of gray levels of the non-defective segment (i.e., Fig. 4a) and defective segment (i.e., Fig. 4b) of a piece of leather. The two histograms are visibly different in many ways, viz, the defective histogram is having less magnitude, shifted more to the left and has a larger spread in width. Hence, from the graphical presentation shown, it is possible to determine if an image has defects simply by analysing certain characteristics of its distribution. Our proposed method therefore revolves around this concept of histogram analysis.

The proposed algorithm composes six main steps: (1) image pre-processing; (2) image segmentation; (3) statistical features computation; (4) feature selection; (5) feature dimensionality reduction, and; (6) defect classification. The flowchart of the algorithm is illustrated in Fig. 5, and the detailed descriptions of each step are elaborated the following subsections.



(a) Non-defect segment and corresponding histogram



(b) Defect segment and corresponding histogram

Fig. 4 Histogram of defect segment vs non-defect segment using Dataset II leather images

4.1 Image pre-processing

The image is converted to grayscale and a 2D Gaussian filter is applied to the image. The purpose of this step is to emphasize the defect as much as possible. The image after converting to grayscale and applying the Gaussian filter is shown in Fig. 6. It is observed that the noise in the background is reduced, thereby making the black line defect more noticeable.

4.2 Image segmentation

The original image has the spatial solution of 400×400 pixels. Each image is equally partitioned into 100 small patches, such that each patch is having the size of 40×40 pixels. The purpose of this partitioning is to allow the extraction of local features in later processes. Since the area of the defects are much smaller compared to the surface area of the leather, segmenting the image reduces this ratio considerably. Ideally, the segments should be around the size of the defects in order to maximise the difference between defective segments and non-defective segments.

4.3 Statistical features computation

To reduce the data redundancy and in the meantime extract important information from the image, each leather image is transformed into a set of vector/ features. Commonly, there are six characteristics of distributions involved in the general image processing process:

1. Mean: Average of the intensities of all pixels in an image.

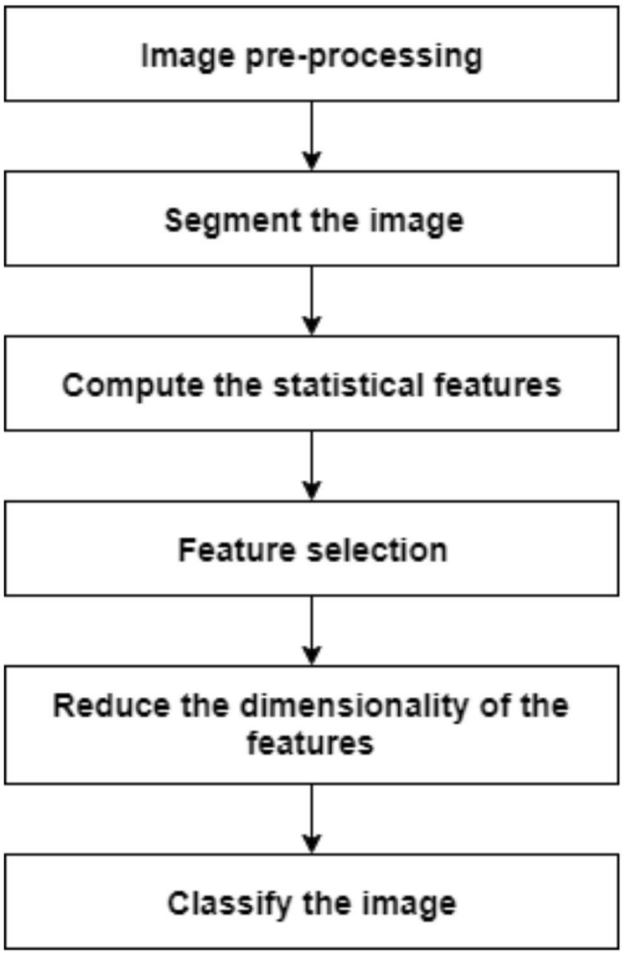


Fig. 5 Flowchart of the proposed method

$$\mu = \frac{\sum_{x=1}^{X}\sum_{y=1}^{Y} I_{x,y}}{X \times Y}, x = 1, \ldots, X; y = 1, \ldots, Y \tag{1}$$

where X×Y is the dimensions (viz., width-by-height) of the leather image.

2. Variance: The spread of intensities (light to dark) of the pixels in the image.

$$\sigma^2 = \frac{\sum_{x=1}^{X}\sum_{y=1}^{Y}(I_{x,y} - \mu)^2}{X \times Y}, \quad x = 1, ..., X; y = 1, ..., Y \tag{2}$$

3. Skewness: The degree of distortion of the distribution of the pixels in the image.

$$\gamma = \frac{X \times Y}{(X \times Y - 1)(X \times Y - 2)} \sum_{x=1}^{X}\sum_{y=1}^{Y} \frac{(I_{x,y} - \mu)^3}{\sigma^3}, \quad x = 1, ..., X; y = 1, ..., Y \tag{3}$$

4. Kurtosis: The measure of the thickness or heaviness of the tails of the distribution of the pixels in the image.

$$\kappa = \frac{n(n+1)}{(n-1)(X \times Y - 2)(n-3)} \sum_{x=1}^{X}\sum_{y=1}^{Y} \frac{(I_{x,y} - \mu)^4}{\sigma^4} - \frac{3(n-1)^2}{(n-2)(n-3)}, \quad x = 1, ..., X; y = 1, ..., Y; n = X \times Y \tag{4}$$

5. Lower quartile value: the first quartile of the range of intensities of the pixels in the image.

$$Q_1 = (Max - Min) \times 0.25 \tag{5}$$

where *Max* is the maximum value of the pixel intensities of the leather image, and *Min* is the minimum value.

6. Upper quartile value: the third quartile of the range of intensities of the pixels in the image.

$$Q_3 = (Max - Min) \times 0.75 \tag{6}$$

As such, there will be six values to represent a patch of the leather image. Instead of the original the raw data of 160000 (400×400), one image can be now represented using 600 values (i.e., 6 values ×100 patches). However, it should be noted that not all types of the statistical features are adequate for the classifier to learn, in both the quantity and quality perspectives. Therefore, selecting suitable features is carried out before the classification stage. Particularly, those features that have the greatest influence on the distribution are regarded as meaningful properties.

4.4 Feature selection

As stated previously, there are 6 distribution features to be considered for selection: mean, variance, upper quartile value, lower quartile value, skew, and kurtosis. In order to choose which feature is best suited for the given dataset, a series of feature selection procedure is proposed in Algorithm 1.

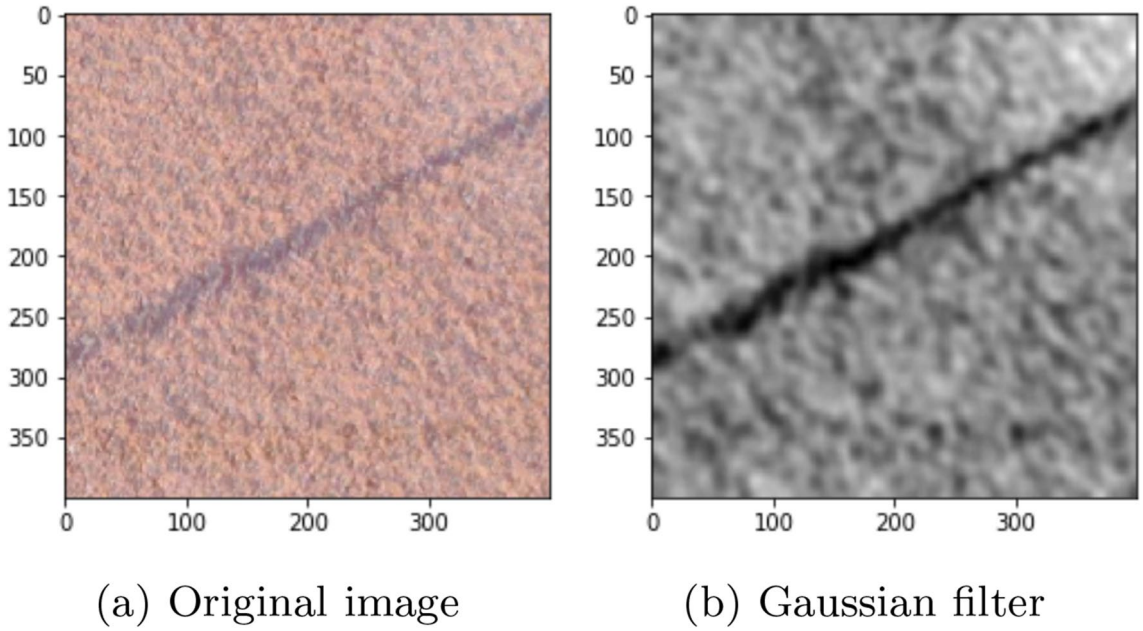


Fig. 6 Leather sample that is (**a**) before and; (**b**) after performing image processing

Algorithm 1 Feature Selection Process

```
for non-defect image, then defect image do
    for each section i in the image do
        calculate all 6 features of the pixel intensities in section i.
    end for
end for
for each feature f do
    perform the two-sample K-S test on the array of calculated feature f for the non-defect image, array of calculated feature f for the defect image.
    get pvalue of the test
end for
Select 3 features with the minimum pvalue
```

Basically, the algorithm uses the 2-sample Kolmogorov-Smirnov test to determine which features have the highest influence. A sample of non-defect and defect images are chosen, then the feature to be compared is computed from the segments in the images, forming two separate distributions: one distribution containing the aggregation of the computed feature from the non-defect images, the other containing the aggregation of the computed feature from the defect images. The two different distribution is put through the 2-sample Kolmogorov test to determine the degree of dissimilarity of the two distributions. A p-value is returned from this test. The algorithm is repeated for each of the six features, then whichever three features that return the lowest p-values are chosen as the selected features.

The Kolmogorov-Smirnov test provides a powerful, non-parametric tool for the objective statistical analysis of histogram data (Conover 1965). The test may be used to test whether two data samples come from the same distribution. In this two-sample case, the Kolmogorov–Smirnov statistic is defined as:

$$D_{n,m} = \sup_x |F_{1,n}(x) - F_{2,m}(x)|, \tag{7}$$

where $F_{1,n}$ and $F_{2,m}$ are the empirical distribution functions of the first and the second sample respectively, and sup is the supremum function.

For large samples, the null hypothesis is rejected at level α if

$$D_{n,m} > c(\alpha)\sqrt{\frac{n+m}{nm}}, \tag{8}$$

where n and m are the sizes of first and second sample respectively. The value of $c(\alpha)$ is given by:

$$c(\alpha) = \sqrt{-\frac{1}{2}\ln\alpha} \tag{9}$$

4.5 Feature dimensionality reduction

When the features have been extracted from all the segments, a representative value of each feature is chosen to represent the feature of the image as a whole. This representative value should be able to reveal whether the image contains defects or not. Three representative methods were examined: percentile thresholding, Gaussian mixture model (GMM) and K-means clustering:

1. Percentile thresholding is simply calculating a threshold value following the formula given:

   $$T = (Max - Min) \times 75\% + Min \tag{10}$$

   where Max and Min are the maximum and minimum of the calculated values of a particular feature for each segment of an image, respectively.
2. The k-means clustering algorithm is a simple learning algorithm and is suitable for identifying spherical clusters in data. The advantages of the k-means clustering is that it is fast and efficient while being reasonably accurate in identifying clusters in the data. In this method, the segment points were divided into 2 clusters (normal cluster and outlier cluster), then the representative value was obtained by taking the difference between the maximum value and the main cluster centroid.
3. The GMM is a probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters. Concretely, GMM can be used to cluster unlabeled data in much the same way as k-means. The advantage of using GMM over k-means clustering is that it can handle oblong clusters, whereas k-means works on circular clusters only. The Gaussian mixture model method is similar to the k-means method; In this method, the segment points are divided into 2 clusters (normal cluster and outlier cluster), then the representative value is obtained by taking the difference between the maximum value and the main cluster centroid.

4.6 Classification

Now that the features have been extracted (3 features per image), these features are put through a classifier. Several widely known classifiers are selected, which are: SVM, kNN, decision tree, ensemble classifier, logistic regression, and Naive Bayes. The evaluation metrics of the model is then determined using a 70/30 for the train/test split (Liong et al. 2019a):

1. SVM (Burges 1998): It categorizes new data points by outputting the optimal hyperplane that distinctly classifies the data points.
2. Decision tree (Fürnkranz 2010): It outlines all the possible outcomes in a tree structure with roots, nodes and leaves. Input data is classified by utilizing an if-then set of rules.
3. NN (Cover and Hart 1967): This is a lazy learning algorithm that determines the class of the input data based on the majority vote from its nearest neighbours. The most common NN technique is kNN.
4. Ensemble classifier (Dietterich 2000): It generates several base classifiers and integrates their discrimination capabilities by combining them into a new and better performing classifier.
5. Naive Bayes (Rish et al. 5001): It is a probabilistic classifier inspired by the Bayes theorem. The probabilities are computed for every factor and selects the outcome with highest probability. It operates under the assumption that the attributes are conditionally independent.
6. Logistic regression (Hosmer et al. 2013): This classifier uses the linear regression equation to construct discrete binary outputs. It is a simple and powerful algorithm for many two-class-only classification problems.
7. Discriminant analysis (Klecka et al. 1980): This classifier models the distribution of the predictors separately in each of the response classes, and uses the Bayes' theorem to convert these into estimates of the probability of the response classes. It is useful for small dataset with normal predictors.

5 Performance metrics

Six performance metrics are utilized to evaluate the performance of the method proposed, viz, specificity, sensitivity, precision, F1-score, error rate and accuracy. These metrics are defined as:

$$\text{Sensitivity} := \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{11}$$

$$\text{Specificity} := \frac{\text{TN}}{\text{TN} + \text{FP}}, \tag{12}$$

$$\text{Precision} := \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{13}$$

$$\text{F1-score} := 2 \times \frac{\text{Precision} \times \text{Specificity}}{\text{Precision} + \text{Specificity}}, \tag{14}$$

$$\text{Error rate} := \frac{\text{FP} + \text{FN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \tag{15}$$

$$\text{Accuracy} := \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{16}$$

where TP refers to true positive, where a defect image is correctly identified as containing defect. FP is the false positive, where a non-defect image is wrongly identified as containing defects. FN represents the false negative, where a defect image is wrongly identified as containing no defects. TN is the true negative, where a non-defect image is correctly identified as containing no defects.

6 Results and discussion

Table 4 shows the results of different classifiers on both datasets. For Dataset I, the features selected were mean, variance and lower quartile range. The Medium kNN and Cubic kNN classifier exhibit the best results for accuracy (99.16%), sensitivity (100%) and error rate (0.84%). On the other hand, the linear discriminant analysis classifier and the subspace discriminant ensemble classifier attains the best F1-score (100%). In short, the experimental procedure performed on Dataset I resulted in an average accuracy of 97.11% and F1-score of 97.39%.

Table 4 Classification performance metrics (%) evaluated on different types of classifiers for Dataset I

Classifier		Sensitivity	Specificity	Precision	F1-score	Error rate	Accuracy
Discrim. Analysis	Linear Discriminant	95	**100**	**100**	**100**	2.52	97.48
	Quadratic Discriminant	95	98.31	98.28	98.29	3.36	96.64
Ensemble	Bagged Trees	98.33	94.92	95.16	95.04	3.36	96.64
	Subspace Discriminant	95	**100**	**100**	**100**	2.52	97.48
	Subspace KNN	96.67	98.31	98.31	98.31	2.52	97.48
KNN	Coarse KNN	93.33	96.61	96.55	96.58	5.04	94.96
	Medium KNN	**100**	98.31	98.36	98.33	**0.84**	**99.16**
	Fine KNN	95	88.14	89.06	88.6	8.4	91.6
	Cosine KNN	93.33	98.31	98.25	98.28	4.2	95.8
	Cubic KNN	**100**	98.31	98.36	98.33	**0.84**	**99.16**
	Weighted KNN	96.67	94.92	95.08	95	4.2	95.8
	Logistic Regression	96.67	98.31	98.31	98.31	2.52	97.48
Naive Bayes	Gaussian Naive Bayes	96.67	98.31	98.31	98.31	2.52	97.48
	Kernel Naive Bayes	96.67	96.61	96.67	96.64	3.36	96.64
SVM	Linear SVM	98.33	98.31	98.33	98.32	1.68	98.32
	Quadratic SVM	98.33	96.61	96.72	96.66	2.52	97.48
	Cubic SVM	96.67	98.31	98.31	98.31	2.52	97.48
	Fine Gaussian SVM	98.33	96.61	96.72	96.66	2.52	97.48
	Medium Gaussian SVM	98.33	98.31	98.33	98.32	1.68	98.32
	Coarse Gaussian SVM	96.67	98.31	98.31	98.31	2.52	97.48
Decision Tree	Fine Tree	98.33	96.61	96.72	96.66	2.52	97.48
	Medium Tree	98.33	96.61	96.72	96.66	2.52	97.48
	Coarse Tree	96.67	**100**	**100**	**100**	1.68	98.32
Average		96.88	97.35	97.43	97.39	2.89	97.11

Table 5 Classification performance metrics (%) evaluated on different types of classifiers for Dataset II

Classifier		Sensitivity	Specificity	Precision	F1-score	Error rate	Accuracy
Discrim. Analysis	Linear Discriminant	32.91	96.9	83.87	89.92	24.12	75.88
	Quadratic Discriminant	36.71	95.98	81.69	88.26	23.49	76.51
Ensemble	Bagged Trees	39.87	89.16	64.29	74.71	27.03	72.97
	Subspace Discriminant	25.95	98.76	91.11	94.78	25.16	74.84
	Subspace KNN	37.97	81.73	50.42	62.37	32.64	67.36
KNN	Coarse KNN	23.42	98.14	86.05	91.7	26.4	73.6
	Medium KNN	34.18	94.43	75	83.6	25.36	74.64
	Fine KNN	**51.27**	81.73	57.86	67.75	28.27	71.73
	Cosine KNN	32.91	93.19	70.27	80.12	26.61	73.39
	Cubic KNN	34.81	95.36	78.57	86.15	24.53	75.47
	Weighted KNN	44.94	89.47	67.62	77.03	25.16	74.84
	Logistic Regression	41.14	93.19	74.71	82.93	23.91	76.09
Naive Bayes	Gaussian Naive Bayes	39.24	95.67	81.58	88.06	**22.87**	**77.13**
	Kernel Naive Bayes	43.67	91.64	71.88	80.57	24.12	75.88
SVM	Linear SVM	26.58	97.21	82.35	89.17	25.99	74.01
	Quadratic SVM	29.11	97.21	83.64	89.92	25.16	74.84
	Cubic SVM	32.91	92.57	68.42	78.68	27.03	72.97
	Fine Gaussian SVM	26.58	92.88	64.62	76.21	28.9	71.1
	Medium Gaussian SVM	29.11	97.83	86.79	91.98	24.74	75.26
	Coarse Gaussian SVM	22.15	**99.38**	**94.59**	**96.93**	25.99	74.01
Decision Tree	Fine Tree	43.67	84.83	58.47	69.23	28.69	71.31
	Medium Tree	39.24	91.95	70.45	79.78	25.36	74.64
	Coarse Tree	46.84	87.31	64.35	74.09	25.99	74.01
Average		35.44	92.89	74.29	82.35	25.98	74.02

Table 6 Confusion matrix of the proposed classification system on the test portion of dataset I, which consists of 119 images

		Predicted	
		No defect	Has defect
Actual	No defect	60	1
	Has defect	0	58

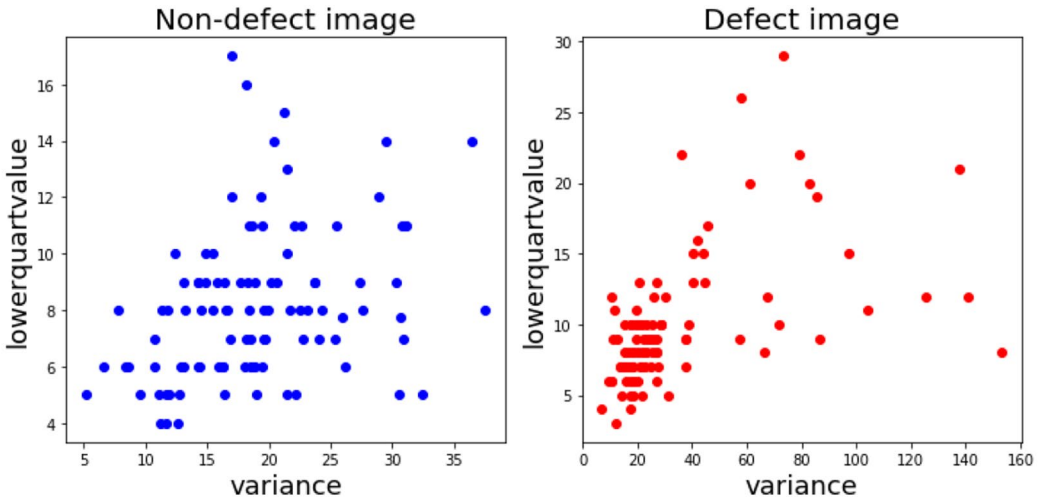
Table 7 Confusion matrix of the proposed classification system on the test portion of dataset II, which consists of 481 images

		Predicted	
		No defect	Has defect
Actual	No defect	62	14
	Has defect	96	309

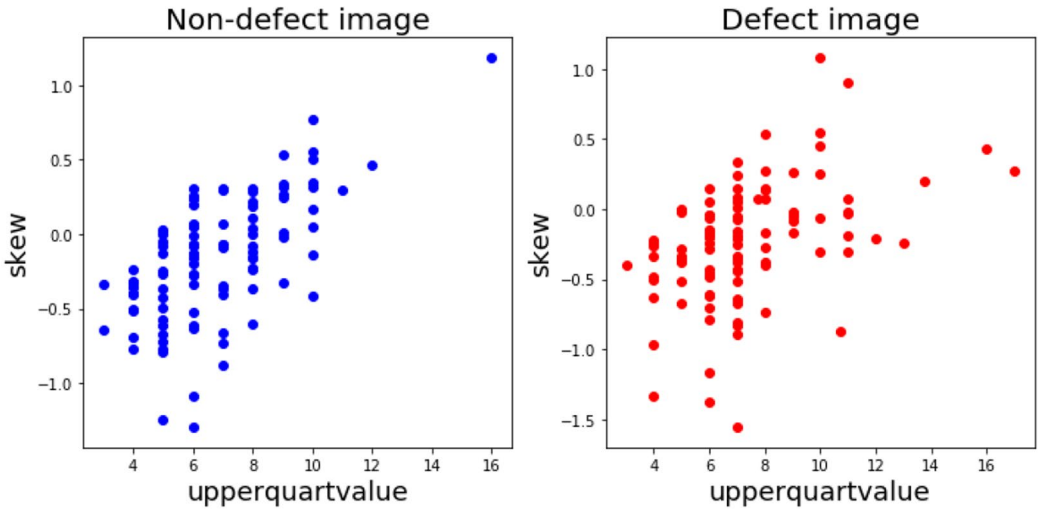
For Dataset II, the features selected were variance, mean and skew. The Gaussian Naive Bayes classifier achieves the highest accuracy of 77.13% and the lowest error rate of 22.87%. The F1-score for coarse Gaussian SVM classifier obtained is 96.93%. In general, the classifier that exhibits the best overall results is the gaussian naive bayes classifier. To conclude, the experimental procedure performed on Dataset II resulted in an average accuracy of 74.02% and an average F1-score of 82.35% (Table 5).

The confusion matrices shown in Tables 6 and 7 help to visualize the results clearer. For Dataset I, among the 119 images, there is only one image (i.e., non-defective image) that has been wrongly classified. On the other hand, the algorithm did not perform so well on Dataset II. The total number for the false negative is relatively high, resulting in a low sensitivity. Even so, the high total of true negatives helped to increase the accuracy to a reasonable value.

To investigate the effectiveness of the proposed method, we perform detail analysis on some of the steps proposed. For instance, Fig. 7 shows the scatter plots of features during the third step in the proposed method, viz, statistical features extraction (step C in Sect. 4). Specifically, Fig. 7a illustrates the scatter plots of the lower quartile value against the variance. It shows a noticeable difference in between the distribution of segments from a defect image and a non-defect image, in that the majority of the points in the defect image scatter plot are clustered towards the bottom graph with several points far away from the cluster (these are the segments with defect). On ther other hand, Fig. 7b portrays the scatter plots for skew against upper quartile value, where



(a) Comparison of scatter plots of lower quartile value against variance for defect image and non-defect image



(b) Comparison of scatter plots of upper quartile value against skew for defect image and non-defect image

Fig. 7 Scatter plots of features for defect and non-defect images

they do not show any significant difference between the non-defective and defective image sampled. Consequently, variance and lower quartile value can be regarded as useful features, whereas upper quartile value and skew can be regarded as not significant.

Next is the analysis performed on the fifth step in the proposed method, viz, feature reduction, that include the techniques of: percentile thresholding, k-means clustering and Gaussian mixture model. The advantages of the percentile thresholding method is its simplicity and effectiveness. A defect image will only have one or two sections with defects, (outliers in the distribution), hence it is important to recognise these outliers in our representative value. This percentile thresholding method effectively spreads the representative value much lower or higher if a defect is present in the image, as seen in Fig. 8. In non-defect images, the representative value is within the main cluster, whereas in defect images, the outlier segments will cause the value outside of the main cluster to become more sparse.

The k-means clustering method and the GMM method both gave similar results, as seen in Figs. 9 and 10. Both were reasonably adept at finding the cluster centroid and

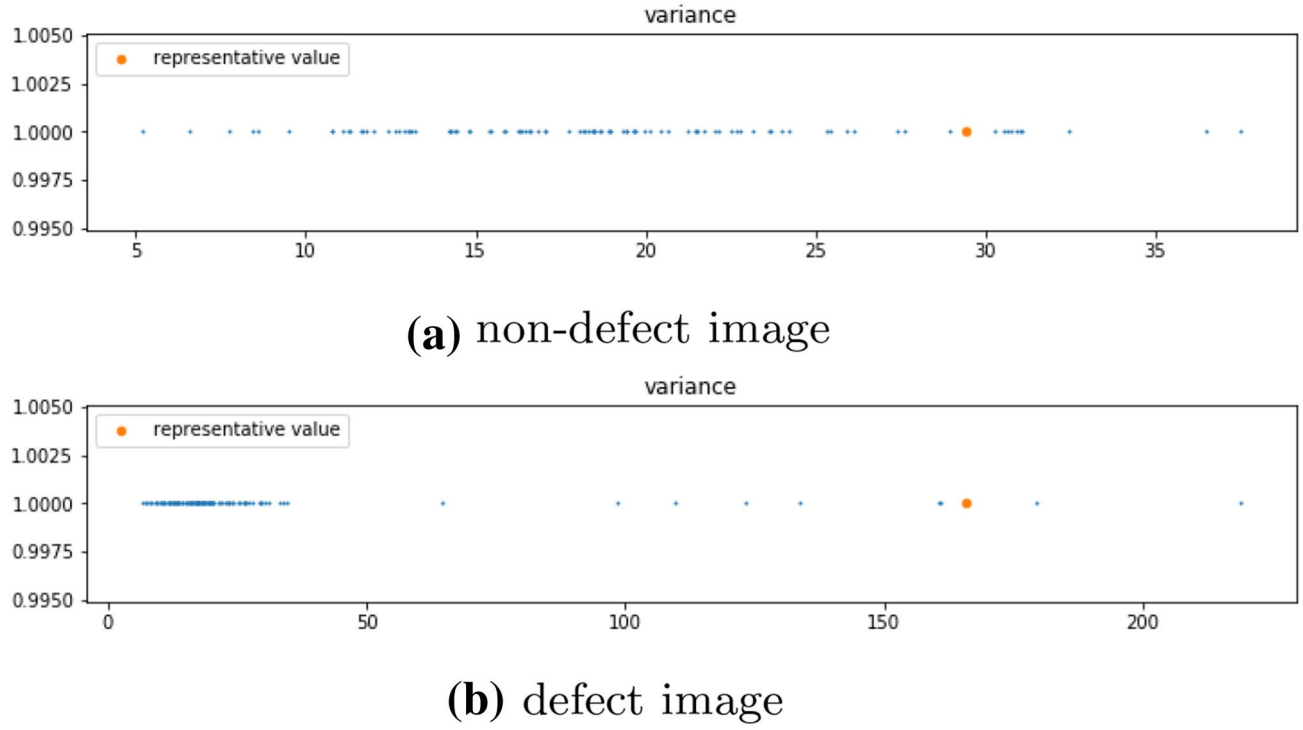


Fig. 8 Features distribution when employing percentile thresholding for dimensionality reduction

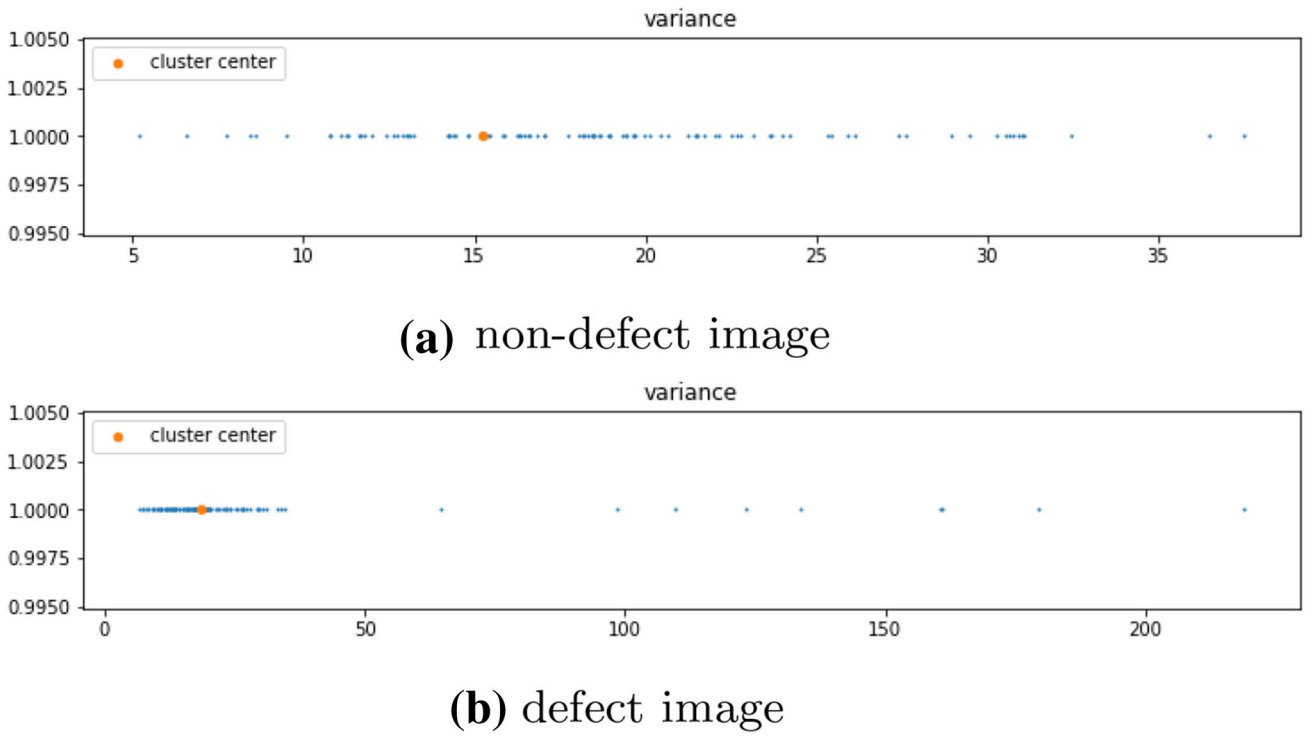


Fig. 9 Features distribution when employing k-means clustering for dimensionality reduction

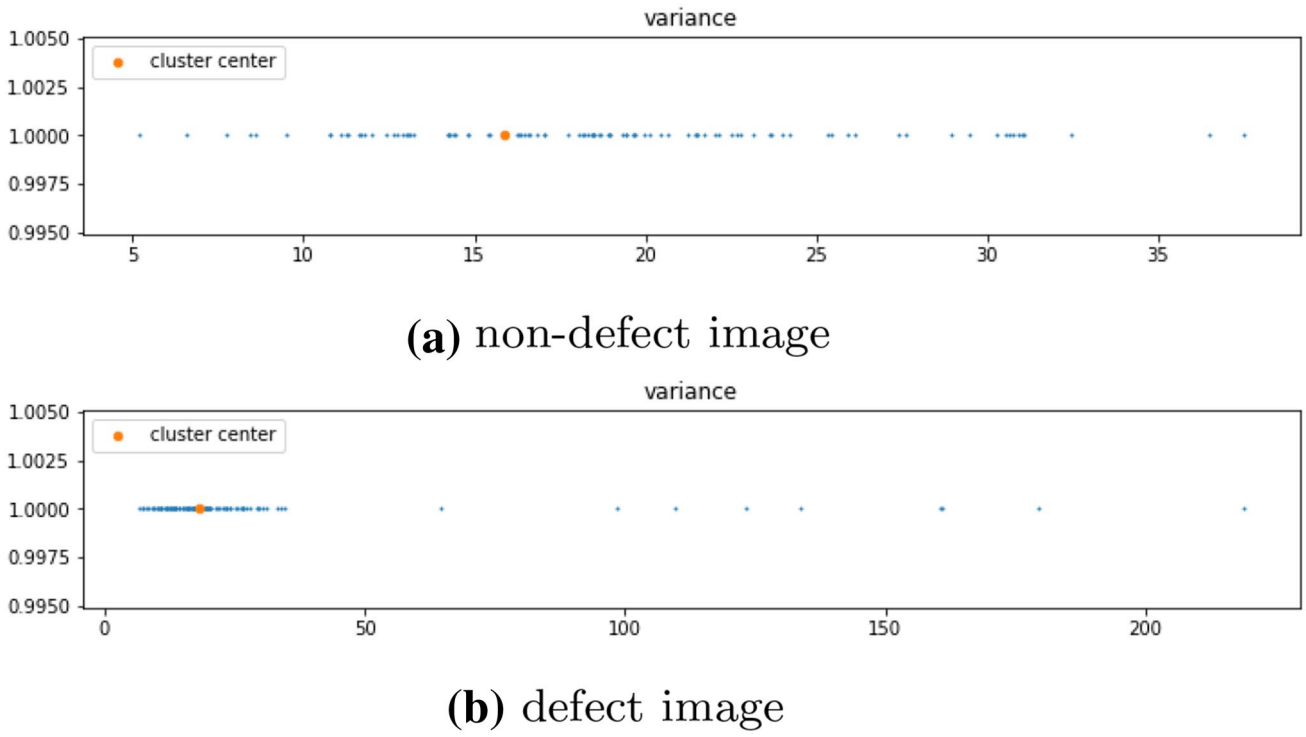


Fig. 10 Features distribution when employing Gaussian Mixture Model for dimensionality reduction

Table 8 Performance comparison to the state-of-the-art

	Method	Defective : non-defective	Train : Validation : Testing split	Method	Accuracy	F1-score
Dataset I	(Liong et al. 2020)[1]	300 :75	80 : 0 : 20	AlexNet	94.67%	94.76%
	Ours	199 : 199	70 : 0 : 30	Statistical analysis + KNN	**99.16%**	**98.33%**
Dataset II	(Liong et al. 2019c)	370 : 1527	60 : 5 :35	Artificial Neural Network	80.30%	89.00%
	(Liong et al. 2019c)	233 : 699	90 : 0 :10	AlexNet	74.18%	84.03%
	Ours	503 : 1102	70 : 0 : 30	Statistical analysis + SVM	**74.01%**	**96.93%**

[1] The defective images include both the black line and wrinkle defects

therefore obtaining a useful representative value. To conclude the feature reduction methods, all three methods showed similar accuracies, therefore the simplest method is preferable, which is the percentile thresholding.

To verify the effectiveness of the proposed approach, a performance comparison is performed for the datasets I and II individually. Specifically, the images used in the dataset I is the same as in (Liong et al. 2020), except that they carried out the task for a three-category classification, whereby distinguishing an image into either the black line defect, wrinkle defect, or non-defective. However, the images per category are fewer compared to that of our experiment, viz, 125 black line, 125 wrinkle, and 125 non-defective images. For the train/ test split of 80/ 20, the accuracy and F1-score obtained were 94.67% and 94.76% when implementing using Alexnet architecture as the feature descriptor. The performance comparison is summarized in Table 8.

On the other hand, for dataset II, the same type of defective image is used in (Liong et al. 2019c). Note that, the train/ validation/ test split were 60/ 5/ 35 with the image number of 1138/ 95/ 664. The best accuracy attained was 80.3% when applying a series of preprocessing techniques before passing to an artificial neural network (ANN) with 50 hidden neurons. Besides, Liong et al. (2019c) employed AlexNet architecture to extract the feature and the classification result is slightly lower. The comparison detail is shown in Table 8. It is noticed that the accuracy outperforms the proposed method but not the F1-score. This is because there is a class imbalance issue that exists in the experiment (Liong et al. 2019c), as non-defective images are about 4

times more than that of the defective image. Therefore, most of the defective images misclassified to the non-defective ones.

7 Conclusion

In a nutshell, this paper presents a statistical, histogram-based approach to perform automated detection of defects on cow leather. Concretely, the common histogram characteristics are proposed to be used as features. Then a statistical test is utilized and the Kolmogorov-Smirnov test is applied as the feature selection step. In addition, a simple, yet effective thresholding method is introduced to perform dimensionality reduction of the features extracted. As a result, the proposed algorithm has low latency and produces reasonably accurate results, viz, accuracy of 97.11% in one of the datasets.

Future works include incorporating machine learning algorithms to improve the robustness of the classification model. Moreover, different types of defects within the same surface can be distinguished. Instance segmentation of multiple defects can be implemented to precisely identify the defective areas.

Acknowledgements This work was funded by Ministry of Science and Technology (MOST) (Grant Number: 109-2221-E-035-065-MY2, 108-2218-E-009 -054 -MY2, 108-2218-E-035-007-, 108-2218-E-227-002-).

References

Ahonen T, Hadid A, Pietikainen M (2006) Face description with local binary patterns: Application to face recognition. IEEE Trans Pattern Anal Mach Intell 12:2037–2041

APLF (2018) Forecasts - global leather goods market size, demand forecasts, industry trends & updates (2018–2025), http://www.aplf.com/en-us/leather-fashion-news-and-blog/news/39758/forecasts-global-leather-goods-market-size-demand-forecasts-industry-trends-updates-2018-2025

Aslam M, Khan TM, Naqvi SS, Holmes G, Naffa R (2019) On the application of automated machine vision for leather defect inspection and grading: A survey. IEEE Access 7:176065–176086

Aslam Y, Santhi N, Ramasamy N, Ramar K (2020) Localization and segmentation of metal cracks using deep learning. J Ambient Intell Humaniz Comput. https://doi.org/10.1007/s12652-020-01803-8

Blum A (2018) Why image analysis could create a breakthrough in manufacturing automation, https://www.b2bnn.com/2018/08/why-image-analysis-could-create-a-breakthrough-in-manufacturing-automation

Bong H-Q, Truong Q-B, Nguyen H-C, Nguyen M-T (2018) Vision-based inspection system for leather surface defect detection and classification, In: 2018 5th NAFOSTED Conference on Information and Computer Science (NICS), IEEE, pp. 300–304

Burges CJ (1998) A tutorial on support vector machines for pattern recognition. Data Min Knowl Discov 2(2):121–167

Conover WJ (1965) Several k-sample kolmogorov–smirnov tests. Ann Math Stat 36(3): 1019–1026. http://www.jstor.org/stable/2238210

Cover T, Hart P (1967) Nearest neighbor pattern classification. IEEE Trans Inf Theor 13(1):21–27

de Souza Júnior AH, Corona F, Barreto GA, Miche Y, Lendasse A (2015) Minimal learning machine: a novel supervised distance-based approach for regression and classification. Neurocomputing 164:34–44

Dietterich TG (2000) Ensemble methods in machine learning, In: International workshop on multiple classifier systems, Springer, pp. 1–15

Fürnkranz J (2010) Decision Tree. Springer, Boston, pp 263–267

Georgieva L, Krastev K, Angelov N (2003) Identification of surface leather defects, In: CompSysTech, Vol. 3, Citeseer, pp. 303–307

Haralick RM, Shanmugam K, Dinstein IH (1973) Textural features for image classification. IEEE Trans Syst Man Cybern 6:610–621

He FQ, Wang W, Chen ZC (2006) Automatic visual inspection for leather manufacture, In: Key Engineering Materials, Vol. 326, Trans Tech Publ, pp. 469–472

Holland JH et al (1992) Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT press, Cambridge

Hosmer DW Jr, Lemeshow S, Sturdivant RX (2013) Applied logistic regression, vol 398. Wiley, Hoboken

Hu G-H (2015) Automated defect detection in textured surfaces using optimal elliptical gabor filters. Optik 126(14):1331–1340

Huang G, Liu Z, Van Der Maaten L, Weinberger KQ (2017) Densely connected convolutional networks, In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4700–4708

Jian L, Wei H, Bin H (2010) Research on inspection and classification of leather surface defects based on neural network and decision tree, In: 2010 International Conference On Computer Design and Applications, Vol. 2, IEEE, pp. V2–381

Kasi MK, Rao JB, Sahu VK (2014) Identification of leather defects using an autoadaptive edge detection image processing algorithm, In: 2014 International Conference on High Performance Computing and Applications (ICHPCA), IEEE, pp. 1–4

Klecka WR, Iversen GR, Klecka WR (1980) Discriminant analysis, vol 19. Sage, Thousand Oaks

Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks, In: Advances in neural information processing systems, pp. 1097–1105

Kwak C, Ventura JA, Tofang-Sazi K (2001) Automated defect inspection and classification of leather fabric. Intell Data Anal 5(4):355–370

Lin M, Chen Q, Yan S (2013) 'Network in network', arXiv preprint arXiv:1312.4400

Liong S-T, Zheng D, Huang Y-C, Gan Y (2020) Leather defect classification and segmentation using deep learning architecture, Int J Comput Integr Manuf, pp. 1–13

Liong S, Gan YS, Huang Y, Liu K, Yau W (2019) Integrated neural network and machine vision approach for leather defect classification, CoRR arXiv:1905.11731

Liong S, Gan YS, Huang Y, Yuan C, Chang H (2019) Automatic defect segmentation on leather with deep learning, CoRR arXiv:1903.12139

Liong S, Gan YS, Liu K, Binh TQ, Le CT, Wu C, Yang C, Huang Y (2019) Efficient neural network approaches for leather defect classification, CoRR arXiv:1906.06446

Mohanaiah P, Sathyanarayana P, GuruKumar L (2013) Image texture feature extraction using GLCM approach. Int J Sci Res Publ 3(5):1

Pereira R, Dias M, Medeiros C, Filho PP (2018) Classification of failures in goat leather samples using computer vision and machine learning

Rajagopalan N, Narasimhan V, Vinjimoor SK, Aiyer J (2020) Deep cnn framework for retinal disease diagnosis using optical coherence tomography images, J Ambient Intell Humaniz Comput, pp. 1–12

Rish I et al (2001) An empirical study of the naive bayes classifier, In: IJCAI 2001 workshop on empirical methods in artificial intelligence, Vol. 3, pp. 41–46

San-Payo G, Ferreira JC, Santos P, Martins AL (2019) Machine learning for quality control system. J Ambient Intell Humaniz Comput. https://doi.org/10.1007/s12652-019-01640-4

Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556

Sundari K (2017) A survey on the approaches used for detection of defects on leather surfaces using image processing. Int J Recent Trends Eng Res 3:374–379

Szegedy C, Ioffe S, Vanhoucke V, Alemi AA (2017) Inception-v4, inception-resnet and the impact of residual connections on learning, In: Thirty-First AAAI Conference on Artificial Intelligence

Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions, In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1–9

Technology S (2020) Satra leather grading accreditation, https://www.satra.com/bulletin/article.php?id=1906

Villar P, Mora M, Gonzalez P (2011) A new approach for wet blue leather defect segmentation, In: Iberoamerican Congress on Pattern Recognition, Springer, pp. 591–598

Xie X, Ge S, Xie M, Hu F, Jiang N, Cai T, Li B (2018) Image matching algorithm of defects on navel orange surface based on compressed sensing. J Ambient Intell Humaniz Comput. https://doi.org/10.1007/s12652-018-0833-0

Zavala-Mondragon LA, Lamichhane B, Zhang L, de Haan G (2019) Cnn-skelpose: a cnn-based skeleton estimation algorithm for clinical applications. J Ambient Intell Humaniz Comput, pp. 1–12