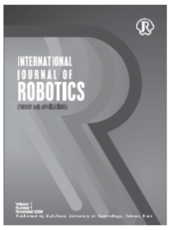

INTERNATIONAL JOURNAL OF ROBOTICS
(THEORY AND APPLICATIONS)
E-mail: IJR@kntu.ac.ir

Evaluation of Classifiers for a Vision-based Automated Mold Protection System Using Modified LBP Features

M. R. Asadi-Gandomani[a], S. S. Hashemi[b] and M. T. Sadeghi[c,*]

[a] Department of Electrical Engineering, Yazd University, Yazd, Iran
[b] Department of Electrical Engineering, Yazd University, Yazd, Iran
[c] Department of Electrical Engineering, Yazd University, Yazd, Iran

ARTICLE INFO

Article history:
Submit: 2023-01-02
Revise: 2023-02-04
Accept: 2023-02-06

Keywords:
Classification Methods
Injection Molding
Inspection Systems
Local Binary Pattern
Machine Learning
Machine Vision

ABSTRACT

For decades, plastic components have been the main parts of products in industries such as food, pharmaceutical, automotive, etc. Generally, these components are created by injection molding machines. Using these machines, raw materials are converted to plastic parts, e.g., bottle caps, dosing spoons, and bumpers. The part of the machine that provisionally holds plastic products is called "Mold" which has a unique form for each product. Since molds are sensitive components with high prices, appropriate care is required. When mold is used as the dynamic part of the machine, it's a high potential for damages due to incomplete product ejection. Utilizing an automated inspection system is a modern solution to prevent possible problems. In this paper, we propose an intelligent system based on machine vision that consists of image capturing, processing, and classification sections. In the processing section, we have used a novel modified Local Binary Pattern algorithm which leads to the suitable features for classifying images into two categories. To achieve the best classifier, four potent machine learning-based methods are evaluated: KNN, SVM, Random Forest, and Gradient Boosting. This evaluation is based on criteria like F1-score, training and processing time, and the experimental results claim that the SVM method is the best classifier with 11.87ms training time, 9.04us processing time, and F1-Score of 0.96.

* Corresponding address: Yazd University, Yazd, Iran,
Tel.: +983531232395; fax: +983538200144.
E-mail address: m.sadeghi@yazd.ac.ir.

1. Introduction

Nowadays there are several techniques in the plastic processing domain and one of the most important is injection molding. An Injection mold is a mechanical part of an injection molding machine that is used in almost all industries and converts thermoplastic into various products [1]. Through the Injection molding process, plastic pallets take a similar shape to cavities placed on the injection mold surface and create products. An Injection molding machine consists of two main parts: a clamping unit and an injection unit. In the clamping unit, products are ejected frequently by opening/ closing a die. In the injection unit, plastic is melted using heat and injected into a specific mold [2]. Although injection molding is basically an automated process, some imperfection happens during the execution most of the time. Because of the lack of workers in injection molding industries, flawed parts are not detected correctly, which leads to reduced productivity [3]. Among all damages, the presence of a foreign body before mold clamping is one of the biggest factors impeding injection mold stability and durability [4].

In the past decade, a lot of research has been done in order to mold injection defect inspection. In [5] detection methods for mold damages have been described. Through the comparison between methods, it can be found that machine vision is applied for the vast majority of molds and is suitable for detecting sudden problems and shutting down the system immediately, and providing high accuracy and speed. Retractor Retaining Bush is a vision-based inspection system that finds defects automatically and has a sorting unit for plastic components. This system has been presented in [6]. In this system addition to sensors and actuators, a vibratory feeder and linear feeder were integrated with the vision system for part presentation. [3] have used a deep learning model based on RNN architecture to study the possibility of defect prediction in plastic injection molding machines. The results show a high false rate due to the dataset skewness. In [7], researchers have proposed a surface-inspection system based on deep learning and a data-centric method for data acquisition and uses state-of-the-art techniques for object detection and segmentation in the injection molding industry. The results show that this approach improves the detection rate.

A machine vision-based monitoring system has been developed in [8] to monitor anomalies during injection molding machine execution. This system could analyze and detect details of products in real time with 100% with good accuracy and high speed. In [9] researchers have designed a radiator tank mold using machine learning. Abnormalities present in injection molding production lines are discovered by machine vision algorithms in [10]. In this paper multi-scale feature Gaussian weighting analysis is proposed for analyzing problems such as dynamic scenes. A machine vision method for mold protection is presented in [11] which is a stable and high-speed approach for mold monitoring and product integration. This method works based on the presence of a foreign body in the mold and the product quality. This solution applies different image algorithms and background updating methods which are in one of the image processing techniques. Such automated mold injection systems have been implemented in industry and some companies like [12-14].

In this paper, we propose a method to inspect the injection molding process. This research develops a fully automated inspection system for checking the presence or absence of plastic products on the mold surface based on machine vision. In the proposed method, appropriate images will be provided by the image acquisition system. Then using a computer vision-based algorithm, images will be processed and the results of the algorithm determine inspection system decisions. Fig. 1 represents the injection molding machine and the proposed inspection system. This research is the developed version of our previous work, which aimed to implement a hardware system in an industrial environment [15]. In our study there is a binary classification task, so four types of classifiers are implemented as the binary classifier. Next, the obtained results from the four classifiers are compared to each other, and the best model will be selected. In addition, since collecting sufficient images for the training of deep learning-based models seems to take considerable time, so we prefer using machine learning-based classifiers. In summary, the novelty of the proposed method is dominant in the feature extraction step as follows: a) solving illumination variations problem by using a modified LBP algorithm as a feature extraction method; b) smart use of histograms of images; i.e., selecting special parts of histograms representing the most differences between two categories. This idea leads to reduce feature space dimension and high-speed operation, especially in real-time applications.

The paper is arranged as follows: In Section 2, we describe the basic knowledge of hardware and software system parts. In Section 3, the implementation of the proposed methods is discussed. Next, the system novelty is explained in Section 4. The obtained experimental results and a comparison between the results of various implemented classifiers are reported in Section 5. Finally, a conclusion in summary is presented.

2. Basic knowledge

2.1. Injection Molding Process

To manufacture plastic products with unique and complicated shapes, an injection molding process is a good choice that is implemented in all industries. In summary, the injection molding process consists of six major stages: Clamping, Injection, Dwelling, Cooling, Mold opening, and Removal of products. The injection molding machine includes two main parts, the core side (movable) and the cavity side (fixed). First, the plastic materials are transferred into the fixed unit, and then the movable unit closes in on the fixed part for the plastic molding. Then the movable part opens and is molded and warm plastics are ejected [16]. In Fig. 2 an example of the mold core side image with some products is shown which is our case study.

During the process, If the injection and removal of products steps are not done well and the movable part is closed again, costly damages will occur to the mold. For this reason, a human operator usually monitors that the mentioned steps are performed correctly and stops the machine in case of any problem. But the use of human monitoring is cumbersome, costly, and error-prone. Using an intelligent system, reliability increases in addition to the elimination of the mentioned disadvantages. In this paper, we present a vision-based inspection system that intelligently monitors the injection molding process and stops the machine when a problem is detected, and prevents any possible damages.

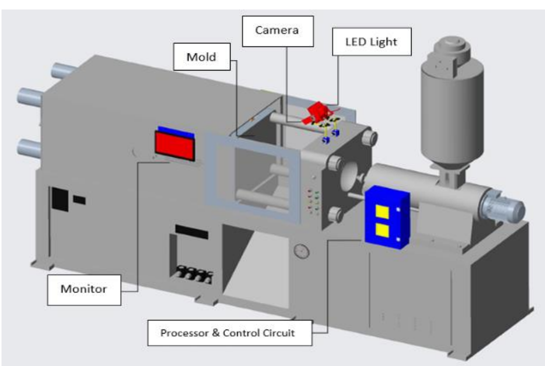

Figure 1. Injection molding machine (gray parts) and our inspection system (colorful parts) [15]

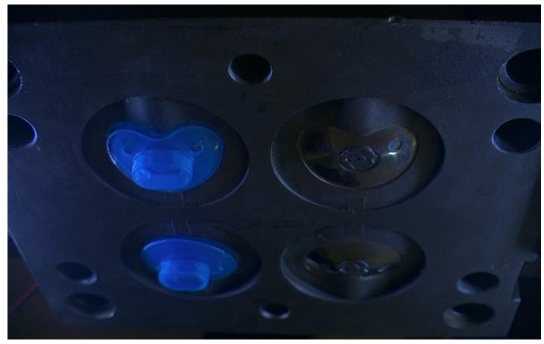

Figure 2. Mold surface with some injected plastic objects [15]

2.2. Machine Vision

Machine vision is an interdisciplinary technology, which involves computer vision, machine learning, sensors technology, optical and illumination systems and etc. The advantages of machine vision in industrial applications are reducing failure costs and errors, increasing productivity, Improvement of the data collection process, and less demand for human experts [17].

2.3. Local Binary Pattern

Local Binary Patterns (LBP) is one of the best visual descriptor models and plays a significant role as a feature in texture classification applications [18]. Due to the discriminative power of the LBP feature extraction method in texture analysis, it is used as a good approach to the traditionally divergent models. In our study, the LBP algorithm is selected because of its robustness to environmental changes (such as illumination variations), and also high-speed analysis caused by simple computations [19].

In this method, “N” pixels of form 3x3 are considered in the neighborhood of each pixel in an image within a radius of “R”. Each neighbor pixel is converted into a binary value due to a comparison with the central pixel value [18]. The mathematical expression of LBP is given as [20]:

$$\boldsymbol{LBP} = \sum_{i=0}^{N-1} s(G_i - G_c)2^i \,, \tag{1}$$

and

$$s(x) = \begin{cases} 1, & if\ x > 0 \\ 0, & otherwise \end{cases} \,, \tag{2}$$

where N is the number of neighborhood pixels and G_c and G_i denote the intensity of the center and neighboring pixel, respectively. The histogram features are extracted from the obtained LBP code and describe the texture. In Fig. 3 the basic LBP operator is shown [18]. The feature vector can now be processed using machine learning algorithms like SVM to classify images.

There are three important methods to determine a pattern with the below specifications [18]:

- Uniform: uniform patterns with grayscale and rotation invariant.
- Rotation Invariant (RI): grayscale and rotation invariant.
- Non-Rotation Invariant (NRI): non-rotation invariant uniform pattern grayscale invariant.

To reduce the feature vector length, a uniform pattern is used which is rotation invariant [21]. In uniform pattern, transitions 1-0 and 0-1 occurs more than other transitions, especially in edges and corners. Therefore, uniform patterns can be considered for feature detection. In the 'RI' pattern, image rotation has no influence on gray values and does not change the LBP value [20]. Although all these three methods are used in our case, the 'uniform' method has the best performance.

2.4. Machine Learning Algorithms

Nowadays, with the development of industry and information technology, various techniques of machine learning and artificial intelligence are widely used in various fields, including automated systems. Although the advent of machine learning returns back to past decades, the improvement of its various algorithms has made it possible to analyze more complex problems in recent years. Therefore, one of the purposes of this paper is to evaluate the performance of various machine learning methods, including K-Nearest-Neighbor, Support Vector Machine, Random Forest, and Gradient boosting in the mold injection process.

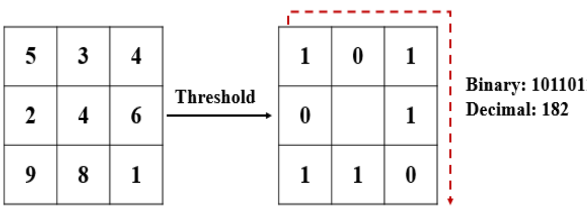

Figure 3. The basic LBP operator [15]

A) K-Nearest-Neighbor

In applications where the number of data is small or there is not enough information about the distribution of experimental data, the K-nearest-neighbor (KNN) method should be one of the first choices to solve the problem. As a matter of fact, this method is simple to implement and there is no need to set a large number of parameters based on assumptions. The KNN algorithm is based on the proximity of similar data. In other words, the algorithm states that data with similar properties are close to each other (Fig. 4), so the similarity of the data is measured by calculating their distance from each other (usually Euclidean distance). Let p be the total number of features (j=1,2,…,p), x_i as an input sample with p features ($x_{i1},x_{i2},\ldots,x_{ip}$) ,and n the total number of input samples (i=1,2,…,n). The Euclidean distance between sample xi and xl (l=1,2,…,n) is defined as

$$\boldsymbol{d}(\boldsymbol{x_i},\boldsymbol{x_l}) = \sqrt{\sum_{\boldsymbol{j=1}}^{\boldsymbol{p}}\left(x_{ij} - x_{lj}\right)^{\boldsymbol{2}}} \quad , \qquad (3)$$

KNN is a supervised method that performs classification based on the labels of train data. In the 1-nearest neighbor rule, the label of test data x is equal to the label of train data m_i, if $d(m_i,x) = min_j\{d(m_j,x)\}$ [22]. Different parameters affect the KNN output, but in this study, we have changed the following parameters in the Scikit-learn[1] library:

- **Number of neighbors:** Number of neighbors around the new data point used for data classification.
- **Weights:** The weights of points which has two values: 'uniform' (equal weights for all points) and 'distance' (weighting points related to the inverse of the distance between the new data point and its neighbors).
- **Power:** Power parameter for calculating distance by the Minkowski metric. for p = 2, it returns Euclidean distance (l_2).

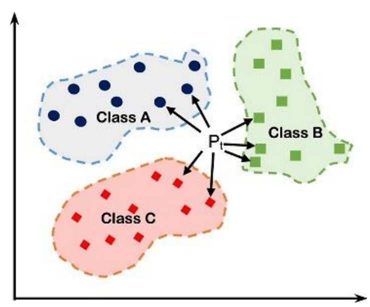

Figure 4. Example of KNN classifier [23]

B) Support Vector Machine

Support Vector Machine (SVM) is one of the most applied methods for binary classification which is a subset of machine learning algorithms. A support-vector machine method is fundamentally based on finding the best hyperplane or set of hyperplanes in a high space for classification or regression tasks. In this method as shown in Fig. 5, the functional margin is the largest distance between the hyperplane and the nearest training data which leads to separate data with higher precision [24]. In the SVM method, the selection of parameters value depends on a convex optimization problem. The SVM is a decision machine and so does not provide posterior probabilities [25].

1 - https://scikit-learn.org

The polynomial kernel is one of SVM's kernels used in several applications which describes the training samples similarity in a feature space. Because of the implementation of polynomials, also non-linear models can be learned by the SVM algorithm, and given features and their combinations are considered [26]. For example, a polynomial kernel with degree-d polynomial is defined as [27]

$$\boldsymbol{K}(\boldsymbol{x},\boldsymbol{y}) = (\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{y} + \boldsymbol{c})^{\boldsymbol{d}} \quad , \tag{4}$$

where x and y are vectors in the input space, and $c \geq 0$ is a controller parameter trading off between higher-order and lower-order terms in the polynomial [28]. All feature space includes the element with a lower degree than d. Degree d controls the flexibility of the classifier result. If d = 2, it is already flexible enough to distinguish between the two classes with a good hyperplane. In SVM with a polynomial kernel, by increasing the d value, data will be linearly separated by a hyperplane which results in high-speed operation. Usually, polynomial kernels have less time-consuming and overfitting problems. Also in many case studies, the polynomial kernel has the best hyperplane model and the lowest classification error compared to the other kernels (linear, RBF, and sigmoid) [26]. These parameters are adjusted as:

- **Kernel:** The kernel type is used in the SVM algorithm. The default value is "rbf", but the polynomial kernel ("poly.") is also used.
- **Degree:** This parameter only refers to the degree of the polynomial kernel; otherwise, it is equal to 3.
- **Coefficient:** It is adjustable only in "poly." and "sigmoid" kernels, otherwise it is 0.

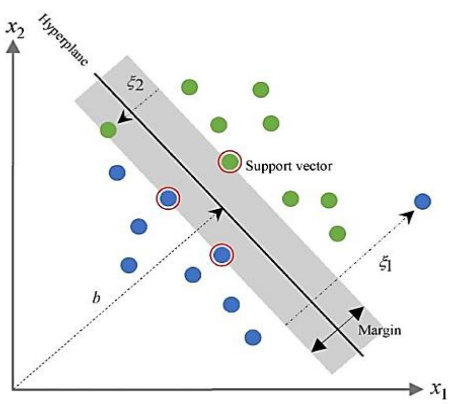

Figure 5. Illustration of Support Vector Machine [29]

C) Random Forest

The random forest algorithm is one of the efficient general-purpose methods in regression and classification. In applications where the number of variables is very large and even greater than the number of observations, the random forest method can perform extremely well by averaging the results of several randomly selected decision trees. The random forest is a comprehensive method that focuses on the more important variables of the problem, so it can be used in large-scale problems and is adapted to different problem situations. The main advantages of the random forest method are the wide range of applications, the requirement of a small number of variables for training the model, ease of use, and the high ability to analyze data in high-dimensional feature spaces. The structure of the random forest is in a such way that it is easily parallelizable, allowing this method to be used in real systems [30]. The random forest method is a supervised algorithm and the initial values of its hyperparameters are adjusted to output good results in general. Fig. 6 shows the structure of a random forest with three decision trees.

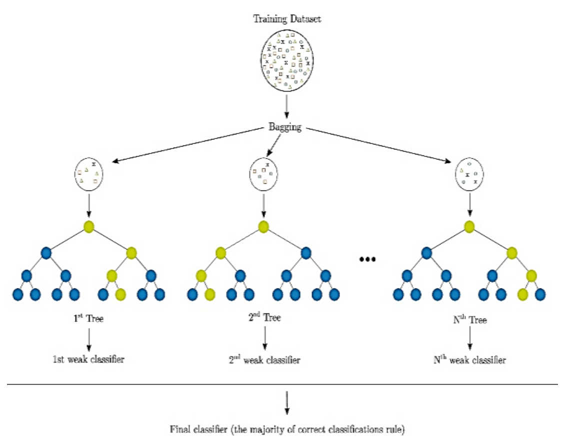

Figure 6. Random forest with three decision trees [31]

A random forest consists of a certain number of binary decision trees. In the random forest, each decision tree expands using bootstrap samples of training data. For example, suppose we have a feature vector with M features. During the expansion process, each decision tree randomly selects a number of f (f<M) features in each of its nodes, and only one feature among f features is used for node splitting. Firstly, an initial value is considered for the number of trees, but during the implementation of the algorithm and the iterative operation, the number of trees increases and random forest grows. In each level, the list of important and unimportant features is updated in four steps: 1) weighting the features and sorting them based on the weight of each feature 2) setting the threshold value for the weight 3) removing the features by weight Less than the threshold 4) select a number of remaining features as important features according to a specific algorithm and categorize others as unimportant ones. It must be considered that important features remain in the

important category until the last step and will not be removed [32]. The altered parameters of random forest in our study are:

- **Number of estimators:** The number of trees in the forest.
- **Criterion:** "gini" and "entropy" criteria for quality evaluation of a split, calculating the Gini impurity and the information gain, respectively.
- **Maximum features:** The number of features used for selecting the best split which are defined by "sqrt" and "log2".

D) Gradient Boosting

The gradient boosting algorithm is one of the most powerful methods in machine learning. In the case of deep learning, error values are divided into two categories: Bias Error and Variance Error. The purpose of implementing a gradient boosting algorithm is to minimize the bias error in the model used in regression and classification problems. The gradient boosting algorithm plays an effective role in the analysis of complex data; therefore, it is a very popular machine learning method. The gradient boosting algorithm is a combination of the Adaptive Boosting (AdaBoost) method and weighted minimization, in addition, the decision trees are used in its structure. For AdaBoost, weak learners are constructed using the initialization of decision trees with one individual split. In this algorithm, data or observations are weighted based on considering the classification difficulty of the observation. At each step, more number of weak learners are added to the model and most of them are allocated to the difficult training samples. In the AdaBoost method, the label of data is identified based on the most voted class determined by the weak learners [33], as shown in Fig. 7.

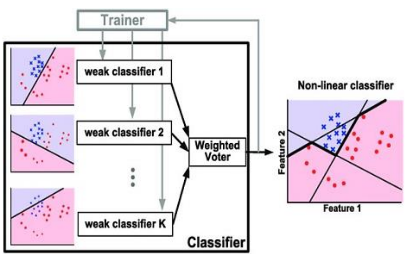

Figure 7. AdaBoost Algorithm [34]

The task of the Gradient Boosting algorithm is to minimize the error caused by the difference between the true class of data and the class predicted by the model, similar to the performance of the gradient descent method used in deep neural networks. Different parameters and hyperparameters affect the performance of the model, so by changing the value of the parameters we can get the appropriate values to have the best result in our study case [33]. These parameters are:

- **Learning rate:** it controls the training rate of the algorithm and must have a compromise with the number of estimators
- **Number of estimators:** The number of boosting stages of gradient boosting; a larger number leads to a better result.
- **Loss:** The loss function is used for optimization, which converts gradient boosting to AdaBoost algorithm by "exponential" loss function, and does logistic regression by "deviance".
- **Subsample:** a subset of samples for training the single base learners. For a value under 1.0, it is equal to Stochastic Gradient Boosting leading to variance reduction and bias growth.

3. Inspection System

The proposed inspection system includes two main parts: image acquisition and image processing. Inspections will be carried out in two stages. **Step 1:** After the plastic injection operation is completed and the clamp is opened, our system captures the surface of the mold, which includes plastic parts, and then the images are processed by a machine learning-based algorithm. At this stage, the system checks that all plastic parts are present and the injection operation is performed correctly. **Step 2:** In this step, the ejector operation is done by the plastic injection machine, and immediately after that, the image capturing is done for the second time in this step, the system checks that no plastic pieces are left on the mold and the ejector operation is performed correctly. If in any of these two steps the image processing section detects that the plastic injection machine has not worked properly, the system will stop the machine by the control circuit. In the following, the image acquisition and image processing sections will be explained.

3.1. Image Acquisition

Image acquisition of the mold surface is done by three main elements: industrial camera, lens, and light. The location of all these elements is always fixed relative to the surface of the mold and they are placed on the fixed part of the mold (Fig. 1). The image acquisition system design depends on the application, so there is no general rule in choosing the type of elements. In our case, because the cycle time of the mold machines is usually enough for processes and the location of the inspected parts is fixed, an area scan camera with a low frame rate and high resolution is selected to increase the quality of the received images. In addition, because in our application the mold size is generally medium, lenses with 6 to 16 mm focal length could be used considering the FOV required for image processing. IR lighting is also used to make the system resistant to environmental light changes. Using the mentioned Image acquisition system leads to a suitable dataset and a stable real-time system operation.

3.2. Image Processing

In this section, the received images are processed and the continuation or cessation of the machine operation is determined based on the image processing results. This section includes three stages: preprocessing, extracting image features, and image classification by a machine learning-based model, which will be explained in the following.

Pre-processing stage: The accuracy increases and the processing time reduction are made by doing the processing only in the areas of the image where the products are present. Due to the constant position of the camera relative to the mold surface, the geometrical coordinates of the location of the products are always fixed. Therefore, considering these coordinates, ROIs are cropped from the image depending on the products number. For example, for a template with 4 products, each of these ROIs is processed separately from the cropped image.

Dataset: Before the feature extraction examination, it is necessary to explain how the dataset is built. This dataset is provided for train, validating, and testing the classifier model. As mentioned, the processing is done on cropped images. Therefore, our dataset includes 700 cropped images and is made into two classes "empty" and "full", and is divided into 490/210 train/test images. Some examples of "empty" and "full" data are shown in Fig. 8.

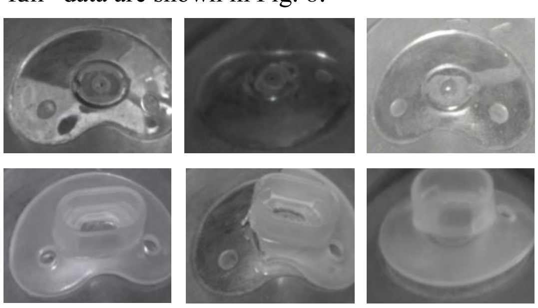

Figure 8. Some dataset samples (the row 1 and 2 correspond to "empty" / "full" class, respectively) [15]

Feature extraction stage: The Local Binary Pattern is considered as the basic method for feature extraction. With the purpose of system enhancement, we have modified this algorithm using an innovative technique that will be explained in section 4.

Classification stage: Finally, the extracted features from the image are classified using a two-class classifier. In this case, four types of machine learning-based classifiers are used to choose the best one. As mentioned, in order to choose the best classifier, the results of four classifiers (KNN, SVM, Random Forest, and Gradient Boosting) are compared based on the F1-score, training and processing time.

The data is classified into two classes "empty" and "full" using the classifier. As an example, for the SVM classifier, the polynomial kernel is commonly used in image processing applications. Using the cross-validation technique, we analyzed several SVM classifiers with different kernels and parameters; and finally, the polynomial kernel with the values of degree = 3 and coefficient = 0.1 achieved the highest classification accuracy among all SVM-based classifications.

4. System Novelty

As we have mentioned in section 2.3, the LBP method is one of the best feature extraction approaches in computer vision. However, using this method might raise the processing time in real-time applications. In this work, we have modified the LBP-based feature extraction by a novel strategy.

Due to the high resolution of cropped images and in order to reduce the processing time, appropriate features of the image must be extracted which is done using the LBP algorithm in the proposed system. Resistance to light changes in the environment is the most important reason for choosing LBP. In the first row of Fig. 9, an empty cavity is considered in two different illumination conditions. The brightness level histogram of the two images is shown in Fig. 9 (second row). As shown in Fig. 9, the difference between the two images is significant when the brightness level is considered as a feature. So, we apply the LBP feature (for example, an LBP with N = 64, R = 1, and the uniform method is used). The graphs in the third row of Fig. 9 show the high similarity between the two graphs. Using the histogram correlation criterion, brightness level and LBP are compared for the two images by:

$$\boldsymbol{d(H_1, H_2)} = \frac{\sum_I (H_1(I) - \overline{H_1})(H_2(I) - \overline{H_2})}{\sqrt{\sum_I (H_1(I) - \overline{H_1})^2 \sum_I (H_2(I) - \overline{H_2})^2}} \ , \tag{5}$$

where

$$\overline{H_k} = \frac{1}{N} \sum_J H_k(J) \ , \tag{6}$$

The values of 0.113 and 0.986 are the obtained results of correlation criteria for the evaluation of the gray value histogram and LBP respectively. It shows that using an LBP extractor leads to high system resistance to the ambient light conditions changes.

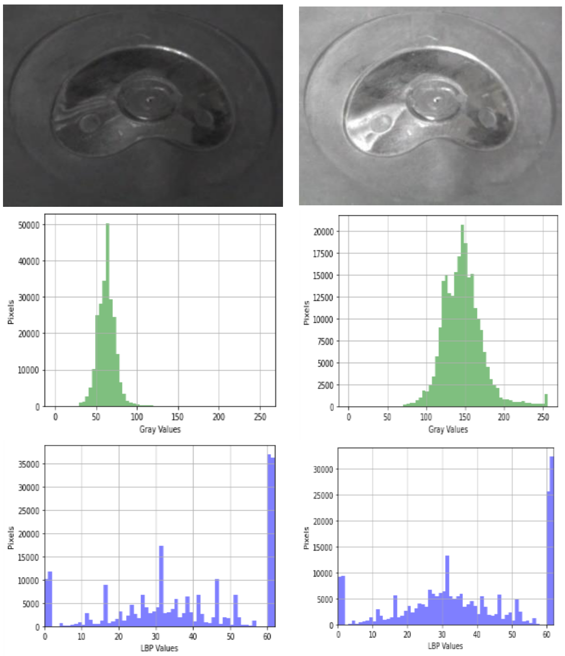

Figure 9. Comparison of one test image in two different illuminations (first row: grayscale images, second and third rows: gray values and LBP values histograms corresponding to the images respectively) [15]

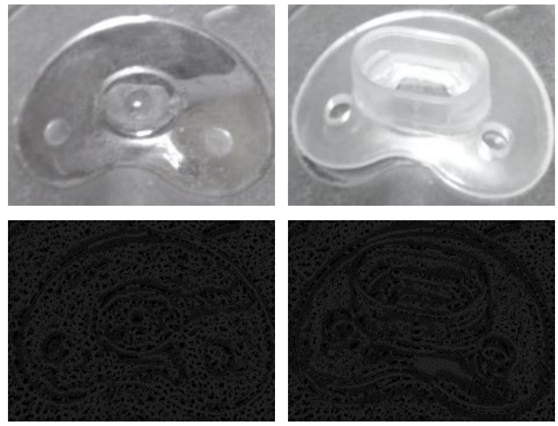

Figure 10. Original Images (first row) and visual results of LBP (second row) [15]

In the parameter selection of the LBP algorithm, the parameters that make the most class differences are selected. In the proposed system, visual results for considering the N and R parameters 32 and 4 respectively with the uniform method, are shown in Fig. 10.

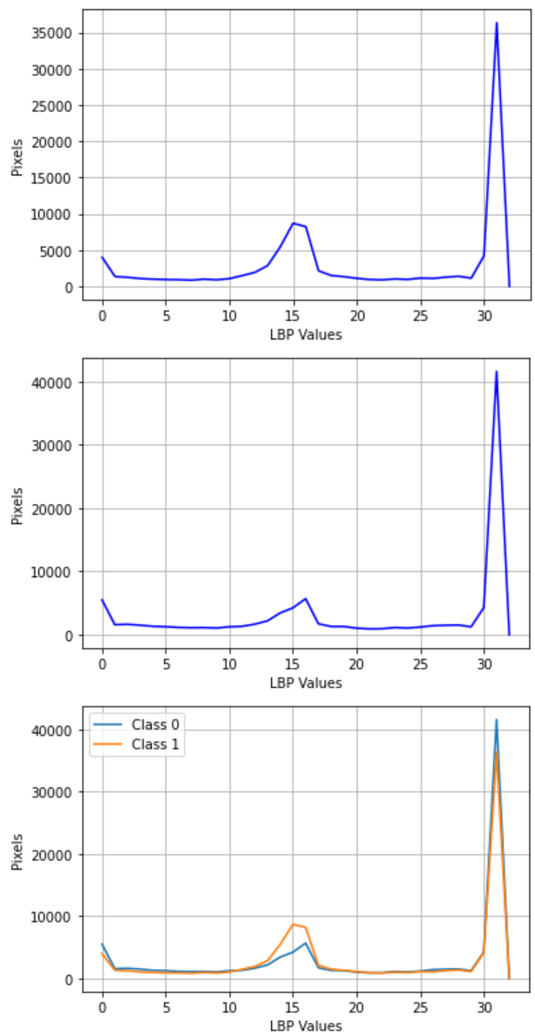

Figure 11. Comparison between LBP values of Figure 10. Images: (top left) LBP histogram of "Empty" image; (top right) LBP histogram of "Full" image; (bottom) overlapped plots [15]

Using the correlation relation, it has the greatest difference between histograms relative to when other parameters are selected. In Fig. 11, LBP values of visual results corresponding to Fig. 10 are presented. We performed this method by examining a number of training data, and their graphs analysis shows that there is the highest difference in histograms for LBP values between 10 and 20 (Fig. 12). As a result, considering the modified LBP based on mentioned values, the final features of every class are the values between 10 and 20. Thus, this modification reduces the feature space size that is the system novelty, leading to increase classification and system inference speed.

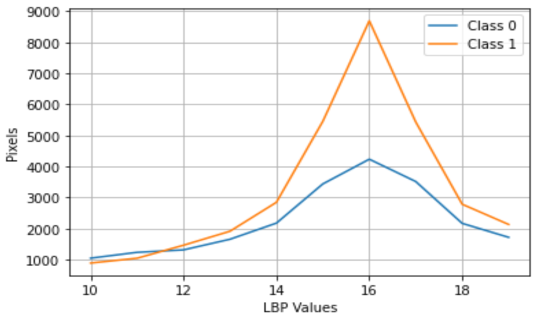

Figure 12. LBP values between 10 and 20 [15]

5. Experimental Results

As mentioned in the previous section, we used different LBP and classifiers to validate the proposed system. For all classifiers, the feature extraction time is 59.81ms. The output result of image classification using each classifier is evaluated by three metrics “Precision, Recall, and F1-score”:

$$\boldsymbol{Precision} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FP}} \quad , \tag{7}$$

$$\boldsymbol{Recall} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FN}} \quad , \tag{8}$$

$$\boldsymbol{F} = \frac{2(\boldsymbol{Precision})(\boldsymbol{Recall})}{\boldsymbol{Precision+Recall}} \quad , \tag{9}$$

where TP, FP, FN, and F, are True Positive, False Positive, False Negative, and F1-score, respectively. In our case, TP means that the ROI which is processed by the system is full of products and the inspection system classifies it as true. FP represents the ROI that is not filled by products and must be in the “empty” class, while the system decides the “full” class for it; and FN results are the filled ROI that is wrongly classified in the “empty” class by the inspection system. The obtained results on the test data are represented in Table 1 to Table 4 based on the executed classifier method. Considering the Local Binary Pattern as the feature extractor, the modified LBP with values N=32, R=4, and the “uniform” method extracts the best features and has the best performance compared to other values [15]. Therefore, N=32, R=4, and the “uniform” method is tuned for LBP in all four classifiers (SVM, KNN, Random Forest, and Gradient Boosting) application. As is shown, there are several results according to set different values of parameters in each classifier. Totally among all classifiers, the top 10 results of experiments, besides the corresponding classifier and the row number in the related table (Table 1 to Table 4), are reported in Table 5. In addition, a comparison between the execution time of all classifiers, including training time and process time, is reported in Table 6.

Table 1. Classification Result for SVM

Method	**SVM**			**P**	**R**	**F**
	kernel	***parameters***				
1	Poly.	Degree=3	Coefficient =0.2	0.96664	0.96667	0.96663
2	Poly.	Degree=2	Coefficient =0.2	0.95710	0.95714	0.95710
3	RBF [35]	C=2		0.95238	0.95238	0.95238
4	Poly.	Degree=2	Coefficient =0.1	0.95236	0.95238	0.95228
5	Poly.	Degree=4	Coefficient =0.2	0.88841	0.88571	0.88392

Table 2. Classification Result for KNN

Method	**Number of neighbors**	**weights**	**power**	**P**	**R**	**F**
1	16	distance	2, 1	0.96664	0.96667	0.96663
2	12,14,15,17	distance	2	0.96190	0.96190	0.96190
3	3	distance, uniform	2	0.95778	0.95714	0.95726
4	25	distance	2	0.95727	0.95714	0.95718
5	7	uniform	2	0.95343	0.95255	0.95255
6	9	distance	2	0.95238	0.95238	0.95238
7	7	distance	2	0.94830	0.94762	0.94776
8	5	distance, uniform	2	0.93972	0.93810	0.93836

Table 3. Classification Result for Random forest

Method	**Number of estimators**	**criterion**	**maximum features**	**P**	**R**	**F**
1	200	gini	sqrt	0.95862	0.95714	0.95733
2	200	gini	Log2	0.95778	0.95714	0.95726
3	100, 250,300	gini	sqrt	0.95343	0.95238	0.95255
4	200	entropy	sqrt	0.94917	0.94762	0.94785
5	10	gini	sqrt	0.92749	0.92381	0.92429

Table 4. Classification Result for Gradient Boosting

Met.	**Learning rate**	**Number of estimators**	**loss**	**Sub-sample**	**P**	**R**	**F**
1	0.1	200	deviance	1	0.95238	0.95238	0.95238
2	0.1	250	deviance	1	0.94776	0.94762	0.94767
3	0.1	100	deviance	1	0.93801	0.93810	0.93803
4	0.1	200	exponential	1	0.93801	0.93810	0.93803
5	0.1	200	deviance	0.5	0.93376	0.93333	0.93346
6	0.01	100	deviance	1	0.92504	0.92381	0.92408

Table 5. Top 10 results among all experiments

	Method	**Row number**	**F**
1	SVM	Table 1- R 1	0.96663
2	KNN	Table 2- R 1	0.96663
3	KNN	Table 2- R 2	0.96190
4	Random forest	Table 3- R 1	0.95733
5	KNN	Table 2- R 3	0.95726
6	Random forest	Table 3- R 2	0.95726
7	KNN	Table 2- R 4	0.95718
8	SVM	Table 1- R 2	0.95710
9	KNN	Table 2- R 5	0.95255
10	Random forest	Table 3- R 3	0.95255

Table 6. Comparison of execution speed of classifiers

Method	Training time (ms)	Processing time (us)
KNN	3.61	53.54 (distance) 12.88 (uniform)
SVM	11.87	9.04
Random forest	285.36	269.44
Gradient boosting	310.84	7.76

As a result, the table data clearly shows that for the selected features in LBP, the accuracy is high and the results are close to each other. In terms of the F1-score in Table 5, the SVM and KNN classifiers have the best results. In terms of classifier training time in Table 6, KNN has the fastest training time with an average of 3.61ms, and the Gradient Boosting has the slowest training time. Google Colab is used to train the models. It must be noticed that the classification process time is more important than the training time because the inspection system works in real-time. Therefore, it can be seen that the Gradient Boosting and SVM method has a faster image classification process, with averages of 7.67ms and 9.04ms, respectively. Considering the power of GPU used in Google Colab, it is clear that these execution times are calculated based on an ideal condition; thus, although using CPU-based processing systems leads to less cost for users, more classification time must be devoted to this analysis. Finally, considering the results of all mentioned criteria, SVM is selected as an ideal classifier for implementation in the mold inspection system in general.

6. Conclusion

In this work, an intelligent inspection system for monitoring a mold component in order to protect it from potential problems was employed. The performance of the injection molding machine is directly influenced by system faults like the presence of a product on the mold surface before the clamping. Thus, we utilized machine vision technology to detect the presence of products on the mold. We selected Local Binary Pattern as a feature extractor and modified that using a novel technique for LBP values between 10 and 20 with N=32 and R=4. Thus, it shows a significant difference in data histograms and leads us to the best possible features. Then we evaluated four main machine learning classification methods based on F1-score, training and processing time criteria in order to classify the products images. According to the results, it was observed that SVM has a better performance than KNN, Random Forest, and Gradient Boosting. SVM with a polynomial kernel achieves the best efficiency so could be selected as the system classifier. As the final conclusion, it could be noticed that the implementation of such modern inspection systems leads to industrial activities become more automated, and human failure costs decrease.

References

[1] Y. Yang, B. Yang, S. Zhu, and X. Chen, Online quality optimization of the injection molding process via digital image processing and model-free optimization, Journal of Materials Processing Technology, Vol. 226, (2015), 85-98.

[2] Polyplastics Co., The Outline of Injection Molding, polyplastics.com, [Online]. Available:https://www.polyplastics.com/en/support/mold/outline/index.html#inj2, [Accessed: Jan. 2022].

[3] S. Kim, S. Kim, and K. R. Ryu, Deep Learning Experiments with Skewed Data for Defect Prediction in Plastic Injection Molding, In 2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA), IEEE, (2018), 1-2.

[4] H. Tercan, et al., Transfer-learning: Bridging the gap between real and simulation data for machine learning in injection molding, Procedia Cirp , Vol. 72, (2018), 185-190.

[5] F. W. Meng, Q. Wang, K. J. Li, and Z. P. Huang, A survey of machine vision-based monitoring methods for abnormalities in molds, In IOP Conference Series: Materials Science and Engineering, IOP Publishing, Vol. 439(3), (2018), 032086.

[6] R. Deepa, S. Usha, and P. S. Kumar, Automated Vision Inspection System for a Plastic Injection Mold Component, In 5th International and 26th All India Manufacturing Technology, Design and Research Conference, (2014), 156-1.

[7] D. Im, and et al., A Data-Centric Approach to Design and Analysis of a Surface-Inspection System Based on Deep Learning in the Plastic Injection Molding Industry, Processes, Vol. 9(11), (2021), 1895.

[8] F. Bai, H. Zheng, Y. Wang, and C. Wang, Research on machine vision system of monitoring injection molding processing, In Seventh International Symposium on Precision Mechanical Measurements, International Society for Optics and Photonics, Vol. 9903, (2016), 99030R.

[9] S. Hwang, and J. Kim, Injection mold design of reverse engineering using injection molding analysis and machine learning, Journal of Mechanical Science and Technology, Vol. 33(8), (2019), 3803-3812.

[10] Q. Wang, F. Meng, Z. Huang, and K. Li, Anomaly detection in periodic motion scenes

based on multi-scale feature Gaussian weighting analysis, Measurement Science and Technology, Vol. 30(5), (2019), 055602.

[11] P. J. Wang, C. J. Xu, J. H. Chen, and X. Q. Tang, Injecting mold protection method based on machine vision, In Advanced Materials Research, Trans Tech Publications Ltd, Vol. 590, (2012), 475-482.

[12] K. Bowes, Applying machine vision to detect parts stuck in injection molds, plasticstoday.com, Apr. 2016, [Online]. Available:https://www.plasticstoday.com/injection-molding/applying-machine-vision-detect-parts-stuck-injection-molds, [Accessed: Jan. 2022].

[13] SICK Co., Monitoring of the injection mold, sick.com, [Online]. Available: https://www.sick.com/us/en/industries/rubber-and-plastics/injection-molding/automated-production-cell/monitoring-of-the-injection-mold/c/p427245, [Accessed: Jan. 2022].

[14] CIMREC Co., Cognex Vision System: Keeps the Injection Molding Process Running, cimtecautomation.com , [Online]. Available: http://www.cimtecautomation.com/industrial-automation-resources/automation-case-studies/newsletter0610/cognex-injection-molding.htm, [Accessed: Jan. 2022].

[15] M. Asadi, S. S. Hashemi, and M. T. Sadeghi, Injection Molding Inspection System Based on Machine Vision, In 2021 9th RSI International Conference on Robotics and Mechatronics (ICRoM), IEEE, (2021), 536-541.

[16] B. A. Samuel, and D. Folley, Design and Construction of a Sensor Analytics System for the Monitoring of the Parameters of a Plastic Injection Mould, International Journal of Sensors and Sensor Networks , Vol. 9(1), (2021), 25-29.

[17] M. L. Smith, L. N. Smith, and M. F. Hansen, The quiet revolution in machine vision-a state-of-the-art survey paper, including historical review, perspectives, and future directions, Computers in Industry, Vol. 130, (2021), 103472.

[18] T. Ahonen, A. Hadid, and M. Pietikäinen, Face recognition with local binary patterns, In European conference on computer vision, Springer, Berlin, Heidelberg, (2004), 469-481.

[19] M. Pietikäinen, and G. Zhao, Two decades of local binary patterns: A survey, In Advances in independent component analysis and learning machines, Academic Press, (2015), 175-210.

[20] T. Ojala, M. Pietikainen, and T. Maenpaa, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Transactions on pattern analysis and machine intelligence, Vol. 24(7), (2002), 971-987.

[21] O. Barkan, J. Weill, L. Wolf, and H. Aronowitz, Fast high dimensional vector multiplication face recognition, In Proceedings of the IEEE international conference on computer vision, (2013), 1960-1967.

[22] S. Zhang, X. Li, M. Zong, X. Zhu, and R. Wang, Efficient kNN classification with different numbers of nearest neighbors, IEEE transactions on neural networks and learning systems, Vol. 29(5), (2017), 1774-1785.

[23] D. M. Atallah, M. Badawy, A. El-Sayed, and M. A. Ghoneim, Predicting kidney transplantation outcome based on hybrid feature selection and KNN classifier, Multimedia Tools and Applications, Vol. 78(14), (2019), 20383-20407.

[24] T. Hastie, T. Robert, F. Jerome, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, (2008).

[25] C. M. Bishop, Pattern recognition, Machine learning, Vol. 128(9), (2006).

[26] Y. Goldberg, and M. Elhadad, splitSVM: fast, space-efficient, non-heuristic, polynomial kernel computation for NLP applications, In Proceedings of ACL-08: HLT, Short Papers, (2008), 237-240.

[27] A. Patle, and D. S. C., SVM kernel functions for classification, In 2013 International Conference on Advances in Technology and Engineering (ICATE), IEEE, (2013), 1-9.

[28] A. Shashua, Introduction to machine learning: Class notes 67577, arXiv preprint arXiv:0904.3664, (2009).

[29] M. A. Nanda, K. B. Seminar, D. Nandika, and A. Maddu, A comparison study of kernel functions in the support vector machine and its application for termite detection, Information, Vol. 9(1), (2018), 5.

[30] G. Biau, and E. Scornet, A random forest guided tour, Test, Vol. 25(2), (2016), 197-227.

[31] N. Sapountzoglou, J. Lago, and B. Raison, Fault diagnosis in low voltage smart distribution grids using gradient boosting trees, Electric Power Systems Research, Vol. 182, (2020), 106254.

[32] A. Paul, D. P. Mukherjee, P. Das, A. Gangopadhyay, A. R. Chintha, and S. Kundu, Improved random forest for classification, IEEE Transactions on Image Processing, Vol. 27(8), (2018), 4012-4024.

[33] C. Bentéjac, A. Csörgő, and G. Martínez-Muñoz, A comparative analysis of gradient

boosting algorithms, Artificial Intelligence Review, Vol. 54(3), (2021), 1937-1967.

[34] Z. Wang, J. Zhang, and N. Verma, Realizing low-energy classification systems by implementing matrix multiplication directly within an ADC, IEEE transactions on biomedical circuits and systems, Vol. 9(6), (2015), 825-837.

[35] Y. W. Chang, C. J. Hsieh, K. W. Chang, M. Ringgaard, and C. J. Lin, Training and testing low-degree polynomial data mappings via linear SVM, Journal of Machine Learning Research, Vol. 11(4), (2010).

Biography

Mohammadreza Asadi Gandomani (also published under Asadi G.) graduated as a telecommunication engineer in the Department of Electrical Engineering at Yazd University. He has been employed as a machine vision engineer, developing vision-based systems in industrial environments. His research interests include computer vision, machine learning, and deep learning.

Seyedeh Sogand Hashemi is a machine learning and deep learning researcher. She received her BSc. degree in Electrical Engineering from the Isfahan University of Technology, and her MSc. degree in telecommunication engineering from Yazd University. Her research interests include signal processing, robotics, machine learning, and deep learning.

Mohammad Taghi Sadeghi is an Associate Professor in the faculty of Electrical Engineering at Yazd University. He received his BSc. degree in Electrical Engineering from Sharif University of Technology, MSc. Degree from Tarbiat Modares University, and his Ph.D. degree from the University of Surrey in Telecommunication engineering. His research interests include computer vision, pattern recognition, image processing, and biometrics.