applied sciences

MDPI

Article

Classification of Objects by Shape Applied to Amber Gemstone Classification

Armantas Ostreika [1,*], Marius Pivoras [1], Alfonsas Misevičius [1], Tomas Skersys [1] and Linas Paulauskas [2,*]

[1] Faculty of Informatics, Kaunas University of Technology, Studentų Str. 50, LT-51368 Kaunas, Lithuania; marius.pivoras@ktu.edu (M.P.); alfonsas.misevicius@ktu.lt (A.M.); tomas.skersys@ktu.lt (T.S.)
[2] MB Supakis, K. Donelaičio g. 20, LT-44239 Kaunas, Lithuania
* Correspondence: armantas.ostreika@ktu.lt (A.O.); lpaul@ktu.lt (L.P.); Tel.: +370-614-21-727 (A.O.)

Abstract: To properly and quickly evaluate an object's shape, in a manner that is suitable for real-time applications, a set of parameters has been created and the shape parametric description (SPD) has been elaborated. This solution is focused on the classification of amber gemstones according to shape. To improve the results obtained by SPD, the most popular machine learning classification algorithms were applied and tested. The proposed method (i.e., SPD) achieved the fastest classification, requiring the least computational resources, while providing an accuracy of approximately 80%. The best results were achieved when the SPD parameters were used in a feedforward neural network (FFNN), and an accuracy of 91.5% was obtained, while the time required for the computations remained in a range that is acceptable for real-time applications.

Keywords: image processing; computer vision; machine learning; expert systems; amber gemstones

check for updates

Citation: Ostreika, A.; Pivoras, M.; Misevičius, A.; Skersys, T.; Paulauskas, L. Classification of Objects by Shape Applied to Amber Gemstone Classification. *Appl. Sci.* **2021**, *11*, 1024. https://doi.org/10.3390/app11031024

Received: 13 November 2020
Accepted: 17 January 2021
Published: 23 January 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction and Related Work

Nowadays, amber is still extracted from the Baltic Sea and adjacent mines and is used to create impressive jewelry, souvenirs, and mosaics. In order to provide amber art craftsmen with suitable raw materials, amber gemstones are selected and sorted according to their size, shape, and shade. Amber can be considered a specific object because there are uncountable varieties of shades, shapes, and sizes.

In this work, we aimed to create an image analysis algorithm capable of classifying amber gemstones on a conveyor, working under real-time conditions. As a result, the submitted object for processing must be assigned to the identified class, or the "other" class when it cannot be set to the given object with acceptable accuracy. The main purpose of this work was to achieve a classification accuracy comparable to expert's work, but for real-time operations (fractions of a second) at the same time.

Scientists in this field have achieved good results in obtaining visual properties and using them to classify and sort objects into a small number of categories. First- and second-order statistical properties [1] are used for visual surface evaluation. Sorting is often used in the food industry [2] to separate objects in waste recycling operations [3]. These systems are based on the acquisition of the optical properties of objects' surfaces using different types of sensors, such as CCD (charge-coupled device) cameras, spectroscopy [4], stereo vision, and infrared light. The optical properties depend on the lighting conditions, which makes it very important to isolate objects from the environment and to install a reliable source of artificial lighting. Such systems have strict operational requirements.

In high-speed automation, qualitative classification is important when classifying color images. Using histogram limitation techniques, pixel counting, different types of lighting, and removal of contour color tones make it possible to obtain a good classification accuracy [5].

Some authors [6,7] have proposed amber classification according to the shape, based on contour properties. The algorithm begins with the processing of each photo. Initially, a photo

is converted from an RGB (red, green, blue) color palette to a binary form. Later, "empty" spaces in the object are filled up and shadows are removed. The image is normalized so that the longitudinal part of the object is parallel to the *X*-axis. Finally, the object image is resized to fit a 256 × 256 pixel size. The centroid distance function (CDF) has been used to identify object forms. This is a one-dimensional function is expressed at a distance from the reference point to the centroid of the object. For each photo, 64 referrals are selected. The referral coordinates are found using the Moore–Neighbor tracking algorithm modified by Jacob's stopping criteria, which provides an array of the object's contour coordinates. Circular shift is applied to eliminate the angle of rotation, resulting in a 64 × 64 matrix. The resulting matrix is normalized (to eliminate the effect of amber size) and inverted (to eliminate the mirror reflection of amber), which results in a 128 × 64 matrix that is used for classification.

The algorithm uses a decision tree ensemble consisting of several decision trees that have been trained with different training and testing data sets. The ultimate solution is the class provided by the decision tree ensemble. For each amber photo, a 128 × 64 matrix is created and passed to the decision tree ensemble. Due to the relatively large amount of data needed to describe the form of amber, the algorithm is not suitable for real-time operating systems because it requires large computational resources. This approach [6,7] requires an expert who must first select amber forms, classify them by hand, and then train the decision trees. Therefore, in this investigation, we also asked ourselves, are there any methods that could help us avoid manual work in such classification tasks?

In [8,9], the authors suggested similar methods to identify forms, where the perimeter and the area ratio values are assigned to the pre-defined classes. The authors [8,9] achieved excellent (over 90%) results in the classification of standard forms such as squares, triangles, rectangles, ovals, and circles, but this method did not produce the desired results in terms of sorting amber gemstones, since their form is difficult to evaluate and decide upon; therefore, in our investigation, other ways for form classification were explored.

Good results were achieved by using convolutional neural networks in aluminum profile classification, harnessing a state of art Mask-RCNN (Region Based Convolutional Neural Network generating a segmentation mask) [10]. This work provides a good reference; however, convolutional neural networks require many computational resources to extract additional features, such as detection of the region of interest and mask prediction. Our current solution is intended to be transferred to low computational power computers such as "ODROID" with ARM processors. Therefore, convolutional neural networks (CNNs) seem too "heavy" for this hardware. Moreover, due to the principle of classification, where only one amber gemstone at a time is dropped onto the conveyor and pushed to the classification "pocket" by air, there will never be multiple objects in a photo. Despite the mentioned restrictions, even small computing systems are becoming powerful enough, and convolutional neural networks could be the next step in pursuing this task.

2. Proposed Methodology

2.1. Shape Parametric Description

To properly and quickly evaluate an object's shape, in a manner suitable for real-time applications, we decided to create a set of parameters that could describe the features of an object's shape but that would require only modest amounts data, thus allowing us to make fast calculations. As a result, shape parametric description (SPD) was elaborated and a corresponding algorithm for shape evaluation was created.

The algorithm evaluates the length of the *X*- and *Y*-axes, as well as diagonals passing through the center of the object, rotated by 45° from the axes mentioned above, and adjacent to the edge of the object, the actual area of the object, and the rectangular area limiting the object. Figure 1 illustrates the parameters for the object form identification.

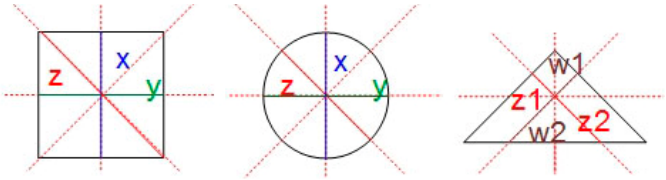


Figure 1. Parameters for form identification.

Here, W1 and W2, analogous to Z1 and Z2, are the distances between the center of the object and the contour of the corresponding diagonals rotated at 45°.

The shape of an amber gemstone can often be described as ambiguous. For example, one side of the stone can be oval-like, while the other side is shaped like a triangle; therefore, asymmetric form was included in the list of investigated forms.

To properly evaluate the object's shape, additional image processing steps are required, e.g., the long axis of an object is calculated and rotated parallel to the X coordinate axis. The image is subjected to a few more rotation procedures (when necessary), where the narrowest part of the object is aligned to the right with respect to the X-axis, and at the top with respect to the Y-axis.

The proposed algorithm for object form classification applying the new SPD is presented in Codes 1–4 below.

Code 1

Pseudocode for the form identification algorithm.
Herein, *tol1* and *tol2* refer to the tolerance values (allowed parameter deviation limits), P is the area of the rectangle limiting the object's shape ($P = x * y$), and p is the real area of the investigated object in pixels.
IF Z1/Z2 < 1 − tol1 OR 1 + tol1 < Z1/Z2
THEN triangle form check
ELSEIF W1/Z2 < 1 − tol1 OR 1 + tol1 < W1/Z2
THEN Asymmetrical form
ELSEIF 1 − tol1 < X/Y AND X/Y < 1 + tol1
THEN symmetric proportional form check
ELSEIF X/Y > 1 + tol1
THEN symmetric nonproportional form checking;
END IF

Code 2

Pseudocode for checking triangle forms.
IF X/Y < 1 − tol1
THEN Isosceles triangle
ELSEIF X/Y > 1 − tol1
THEN Right triangle
ELSE
THEN Equilateral triangle
END IF

Code 3

Pseudocode for checking proportional symmetric forms.
IF p/P < 1 − tol1
THEN Circle
ELSE
THEN Square
END IF

Code 4

Pseudocode for checking nonproportional symmetric forms.
IF p/P > 1 − tol1 AND X/Z < 1 − tol2
THEN Asymmetrical form
ELSEIF p/P > 1 − tol2 AND X/Z > 1 + tol1
THEN Rectangle
ELSEIF p/P < 1 − tol1 AND X/Z > 1 + tol1
THEN Oval
END IF

A complete list of the parametric conditions for each of the "basic" shapes is listed below.
Circle:
x/z = 1 *AND* x/y = 1 *AND* Z1 = Z2 *AND* p < P
Oval:
x/z > 1 *AND* x/y > 1 *AND* Z1 = Z2 *AND* p < P
Rectangle:
x/z > 1 *AND* x/y > 1 *AND* Z1 = Z2 *AND* p = P
Square:
x/z < 1 *AND* x/y = 1 *AND* Z1 = Z2 *AND* p = P
Equilateral triangle:
x/z < 1 *AND* x/y = 1 *AND* Z1 $\neq$ Z2 *AND* p < P
Isosceles triangle:
x/z < 1 *AND* x/y < 1 *AND* Z1 $\neq$ Z2 *AND* p < P
Right triangle:
x/z > 1 *AND* x/y > 1 *AND* Z1 $\neq$ Z2 *AND* p < P
Uneven rhombus:
x/z < 1 *AND* x/y < 1 *AND* Z1 = Z2 *AND* p = P

2.2. Machine Learning Algorithms

The main problem of the SPD, proposed above, is that in real circumstances, exact correspondence to all conditions described here rarely occurs. Such conditions as "x/y = 1" or "Z1 = Z2" have no sense for practical applications unless we can allow some tolerance or some deviation of the parameters. However, if we allow deviation, then some decisions become questionable, e.g., when a circle becomes an oval or when an equilateral triangle turns into an isosceles triangle and so on. During the first experimental testing, there were no clear indications about how big the tolerances should be allowed to be in order to achieve classification results similar to experts' classifications. Furthermore, for different shapes, the analytical results suggested different tolerances for the same parameters. Therefore, the decision was made to apply the most popular machine learning classification algorithms. The algorithms that we selected for the experiments are listed below.

Ensembles of decision trees (EDTs): A non-parametric supervised learning method used for classification and regression. Decision trees learn from data to approximate a sine curve with a set of if–then–else decision rules. The deeper the tree, the more complex the decision rules and the better the fit of model.

Decision trees build classification or regression models in the form of a tree structure. They break down a data set into smaller and smaller subsets, while at the same time incrementally developing an associated decision tree. The final result is a tree with decision nodes and leaf nodes: a decision node has two or more branches, while a leaf node represents a classification or decision. The topmost decision node in a tree, which corresponds to the best predictor, is called the root node. Decision trees can handle both categorical and numerical data. EDTs combine several decision trees to achieve better predictive performance than utilizing a single decision tree by grouping weak learners.

K-nearest neighbors (KNN): A classification model that classifies, for each unlabeled instance, its K-nearest neighbors. The process of clustering K-means begins with randomly assigning objects to a predetermined number of clusters. The objects are then distributed to

other clusters to minimize cluster distribution, which is essentially a square distance from each observation to the center of the associated cluster. If redistributing an object to another cluster results in reduced cluster distribution, this object is moved to that cluster [11]. During the K-means clustering method, the cluster dependence may change at each stage of the clustering iteration. The important thing is that the number of clusters must be specified in advance in the clustering of K-means before analysis.

Naïve Bayes (NB): An algorithm that works on the assumption that all data parameters are considered independent of each other and that each parameter equally affects the final classification result. Naïve Bayes methods are a set of supervised learning algorithms based on applying Bayes' theorem with the "naïve" assumption of conditional independence between every pair of features, given the value of the class variable

Support vector machine (SVM): The essence of this classifier is to create hyperplanes that would separate the data into different classes. For creating a hyperplane, the training set objects are divided into parts, such that the distance between the nearest elements belonging to different classes to that hyperplane are maximal. Creating a hyperplane depends solely on a subset of the training set consisting of so-called support vectors.

Feedforward neural network (FFNN): An information processing structure that mimics some of the information transfer processes that take place in the brains of living organisms. A neural network consists of many interconnected, very simple computational elements (artificial neurons). These elements, connected by joints of various weights, are an approximate model of biological neurons. The artificial neural network aims to emulate some properties of biological systems, such as the ability of biological systems to learn and adapt. During learning, the strength of the connections bind neurons in the brain of living organisms.

To compare and evaluate the algorithms, we adopted the confusion matrix, which is a good technique to summarize the performance results of classification calculations, since it is often used to describe the performance of a classification model (or "classifier") based on a set of test data for which the true values are known. As the classes were unbalanced, the decision was made not just to calculate the prediction accuracy and calculation speed for the model evaluation, but the F1 macro- and F1 micro-scores as well.

3. Experiments and Results

3.1. Hardware and Implementation

For the experimental testing, a real conveyor with amber splinters was used. The operation was carried out as follows: pieces of amber fell off of the vibrating bowl onto the conveyor; the laser fork detected a piece of amber that interrupted the laser beam and sent the signal to the digital camera (type FFMV-03MTC, mpg Point Gray, Richmond, B.C. V6W 1K7 Canada), which captured an image and transmitted it for processing. MATLAB® by the MathWorks, Inc. version 2020b and a "AK4" computer by Mikrotestas UAB, containing Intel® Core™ i9-9900K 3.6 GHz processor, 32 GB RAM, Windows 10 x64, SSD, were used for algorithm implementation and image processing.

During the first phase of the experiment, 4311 photos were produced with a resolution of 640 × 480 pixels. The expert manually sorted all of these images into corresponding classes. The distribution of the classes is presented in Figure 2.

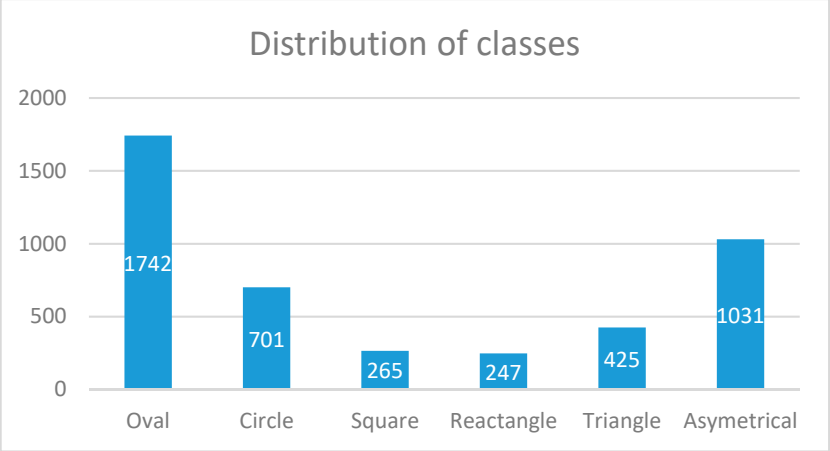


Figure 2. Expert-classified amber gemstones before processing.

3.2. Preprocessing

It is important to properly separate an object from a background before starting to calculate its properties. This is preceded by the automatic evaluation of the histogram and contrast adjustment, which removes shadows. If the shadows are not removed properly, the contour of the amber gemstone becomes similar to an oval in most cases, thus losing the most important properties for the classification. Figure 3 shows examples of amber gemstone pictures before processing.



Figure 3. Images of amber gemstones before processing.

Shadow intensity is highly dependent on the type of artificial lighting and amber color, which can vary in terms of transparency—being transparent, semi-transparent, and white—and can have many yellow or brown shades, etc. The best results were achieved with Dome-type lighting, where the light was evenly distributed. Figure 4 illustrates images of amber after shadow removal. Although the shades of the amber gemstones were different, the shadows were successfully removed and an exact representation of the object was obtained in white pixels, as shown in Figure 5.



Figure 4. Images of amber gemstones after shadow removal.



Figure 5. Amber gemstone representation in white pixels.

In order to identify the shape of the objects, the dimensions described in Section 2 were calculated for each amber photograph: the length of the *X*- and *Y*-axes passing through the

center of the object, the diagonals of the object rotated 45° from the main axes, and the actual and rectangular areas of the object.

Before starting the calculations, the long axis of the gemstone was rotated parallel to the *X*-axis. The long axis of the object was considered to be the long axis of an ellipse, which has the same second momentum as the white region representing the object. Then, if necessary, the photo of the amber was rotated so that the narrowest part of the object was on the right with respect to the *X*-axis and at the top with respect to the *Y*-axis. In Figure 6, a picture of an amber gemstone before and after the rotation operation is shown.

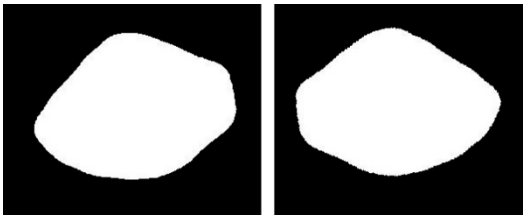


Figure 6. Gemstone image before and after rotation operations.

The required dimensions were calculated after the initial steps had been performed in order to determine the shape of the object (shown in Figure 7). The sizes were relative to the image, which allowed the form to be unrelated to the image scale and actual height position of the camera.

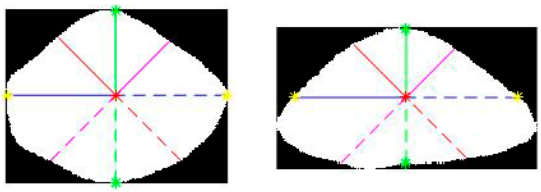


Figure 7. Required dimensions to determine the shape of the object.

3.3. Experimental Results for the SPD and CDF Approaches

The decision was made to test the SPD approach by allowing some tolerance of the parameters. During the experiments, different tolerance limits were tested, from 5% to 15%, to classify the objects when checking the ratio of the corresponding dimensions. It appeared that the classification results greatly depended on the value of the selected tolerance, which directly affected the distribution of the analyzed objects to the form classes. Lower tolerance values allowed more objects to be assigned to the triangle shape, which was not approved by the human expert. Therefore, the tolerance parameter needed to be adapted by performing additional experiments.

At higher tolerance values, the analyzed objects were assigned more to the asymmetric form, since the conditions of this particular class were met. When the tolerance value was chosen to be approximately 10%, then the more complex forms were distinguished and this provided the most acceptable results in comparison to the human expert assessment (see Table 1). During the experiment, it was determined that different tolerance values should be used to evaluate the ratio of the real and rectangular limited areas.

Table 1. Classification by form using the shape parametric description (SPD) approach.

Class	Correct	Incorrect	Accuracy
1. Oval	1437	305	82.5%
2. Circle	563	138	80.3%
3. Rectangle	185	80	69.8%
4. Square	173	74	70.0%
5. Triangle	321	104	75.5%
6. Asymmetrical	780	251	75.7%
Total	3459	952	78.4%

By applying the SPD approach, the average accuracy of the classification algorithm reached 78.4% of the expert's (human) decisions. There were cases where the form of an amber piece was difficult to identify unambiguously even for human eyes due to the uncertainty of the shape; for example, when one side of an amber stone was similar to a circle and the other half to a square or the like.

Using the same set of photographs taken during the real-time experiment, the method mentioned in [12,13] (i.e., classification by the form using centroid distance function (CDF)) was tested and compared. The results of the method are presented in Table 2.

Table 2. Classification by form using the centroid distance function (CDF) approach.

Class	Correct	Incorrect	Accuracy
1. Oval	1437	305	82.5%
2. Circle	503	198	71.8%
3. Rectangle	176	89	66.4%
4. Square	155	92	62.8%
5. Triangle	303	122	71.3%
6. Asymmetrical	753	278	73.0%
Total	3267	1084	74.1%

The experiments showed that the proposed SPD method provided better accuracy results than CDF.

One of the main differences between these methods was the speed of the object class identification. The class identification time using the CDF method took 0.33 s on average, while classification using the proposed SPD took an average of 0.03 s. Thus, the SPD method was up to 11 times faster than the CDF method.

Since the form of amber could not be defined unambiguously in some cases, the decision was made to carry out additional tests using a well-known set of three basic shapes. The set consisted of a circle (set of 3720 samples), a square (set of 3765 shapes), and a triangle (set of 3720 shapes). Table 3 shows examples of these basic forms. The size of the images analyzed was 200 × 200 pixels. Each form was individually tested for classification. Table 4 shows the results of the classification of the basic form data set.

Table 3. Examples from the basic form data set.

Circle	Square	Triangle
		

Table 4. Results of the classification of the main form data set.

Form	SPD Method		CDF Method	
	Properly Identified Class	Incorrectly Identified Class	Properly Identified Class	Incorrectly Identified Class
Circle	3484	99	3619	101
Square	3277	100	3651	114
Triangle	3286	103	3610	110

The SPD method showed slightly better results than the CDF method (97.3% vs. 97.1%), but the achieved difference can be treated as negligible. However, the average classification time provided similar proportions for the methods as in the previous experiment with real amber photographs. This time, the CDF method identified a class in 0.12 s on average, while the proposed SPD method showed an average of 0.01 s. The size of the images and the central processing unit speed influenced the average time of the calculations, but since the data sets were the same and the calculations were made on the same central processing unit (CPU), the proportional speed of both methods could be clearly identified.

3.4. Experimental Results after Applying Machine Learning

As mentioned above, we met some difficulties in assigning tolerance values to the introduced shape parameters. Therefore, machine learning methods (i.e., EDT, Naïve Bayes, SVM, KNN, and FFNN) were tested in order to achieve a better association of the assigned parameters, proposed in the SPD algorithm, with the shapes classified by the expert.

In order to understand whether the parameters defined in the SPD algorithm clearly describe the shapes under investigation, the distributions of the shape parameters were drawn and analyzed for the manually classified amber gemstones. As can be seen in Figure 8, some parameters, such as p/P (see SPD approach described above), had quite clear spatial locations, while others, such as Y1/Y2, overlapped quite a lot. Nevertheless, by combining the parameters into sets, good classification results could be achieved.

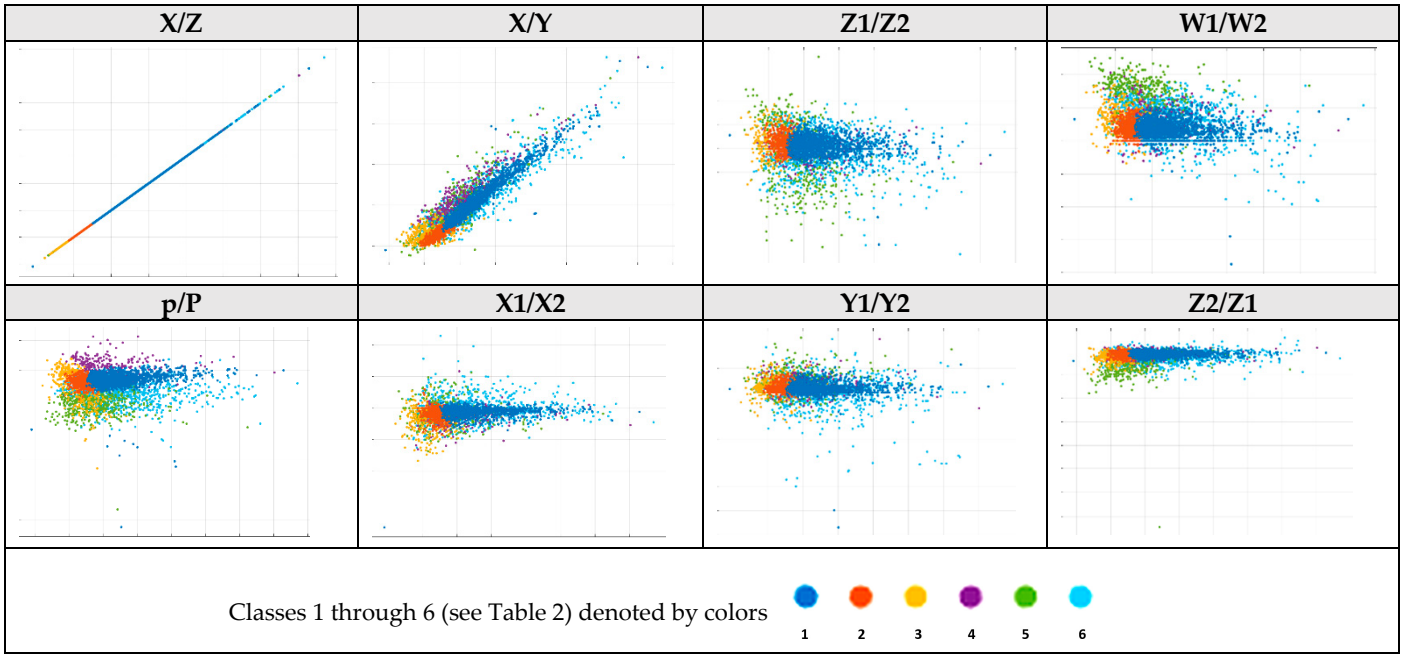
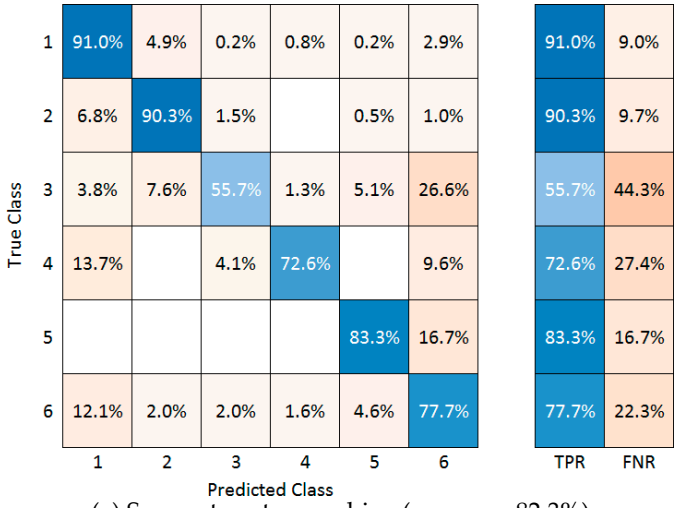


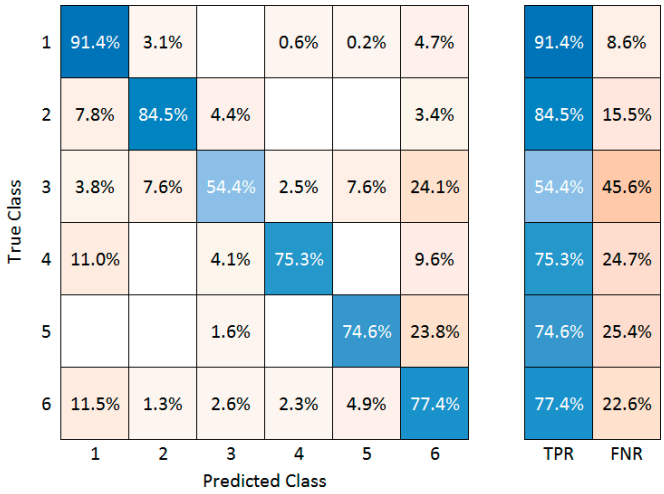
Figure 8. Distribution of the shape parameters.

To confirm this, machine learning was applied. Calculations were performed using MATLAB functions. Training and testing data were randomly separated into two different

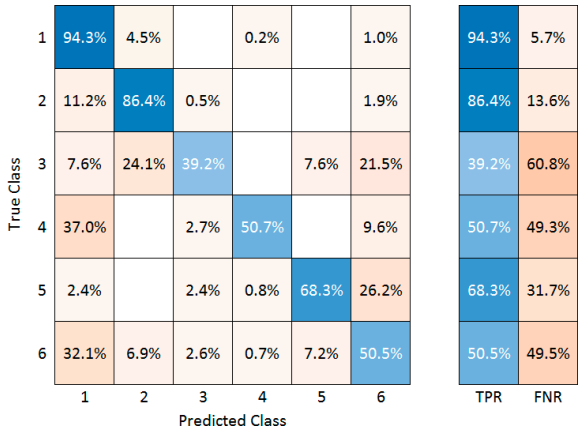
data sets (training data set comprising 70% and testing data the remaining 30%). These data sets were kept the same during all of the experiments performed. The training data set was used to train the models, while the testing data set was used for validation. To evaluate the performance and accuracy of the algorithms, confusion matrixes were created and true positive rate (TPR) as well as false negative rate (FNR) values were calculated (see Figure 9).



(**a**) Support vector machine (accuracy 82.3%)

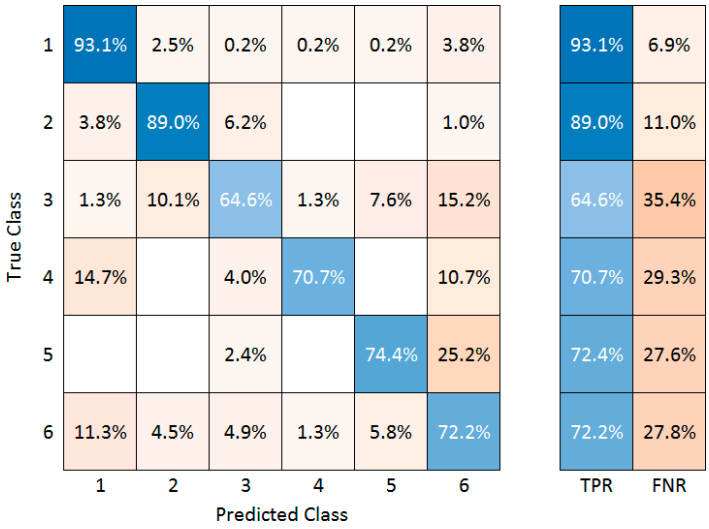


(**b**) Ensembles of decisions trees (accuracy 83.9%)

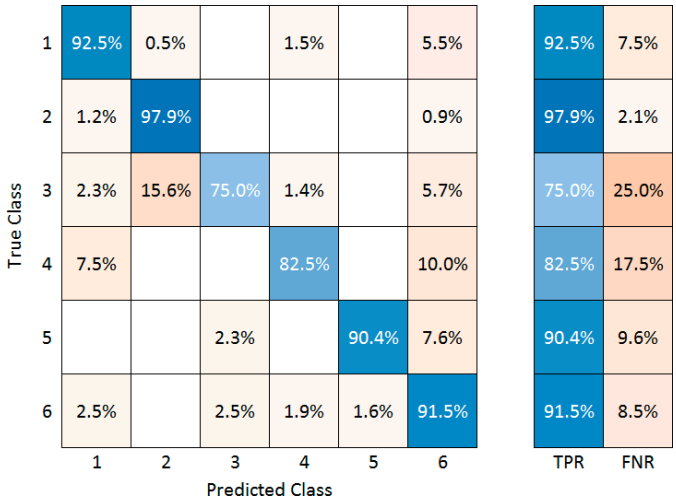


(**c**) K-nearest neighbors (accuracy 74.5%)

Figure 9. *Cont.*



(**d**) Naïve Bayes (accuracy 82.6%)



(**e**) Feedforward neural network (accuracy 91.5%)

Figure 9. Experimental results with machine learning algorithms.

For the first machine learning (ML) experiment, linear and non-linear SVM kernels were tested. As seen from the distribution of the shape parameters (see Figure 8), it was hard to separate the classes using linear classification. Non-linear SVM kernels—quadratic kernels provided better results—provided an overall accuracy of 82.3 and F1 macro- and micro-scores of 0.74 and 0.9, respectively.

An EDT was used when every tree in the ensemble was grown on an independently drawn bootstrap replica of the input data. The observations not included in this replica were "out of the bag" for this tree. Overall, this algorithm showed an 83.9% classification accuracy with F1 macro- and micro-scores of 0.76 and 0.91, respectively.

The Manhattan, Minkowski, and Euclidean distance functions were tested with the KNN algorithm. The best result was achieved using the Euclidean distance function, achieving 74.5% with F1 macro- and micro-scores of 0.66 and 0.85, respectively. KNN provided the lowest accuracy of all of the methods tested in our experiments.

The Naïve Bayes algorithm assumes that all instances are intended for one another and it takes individual measurements for prediction. The overall accuracy of this method achieved 82.6% with F1 macro- and micro-scores of 0.77 and 0.9, respectively.

In the FFNN, one deep layer containing 12 nodes was used. Bayesian regularization backpropagation was used for the training function, which updates the weight and bias

values according to Levenberg–Marquardt optimization. It also minimizes a combination of squared errors and weights and then determines the correct combination to produce a network that generalizes well. In training, 50 epochs were used with a learning rate of 0.0001 and a damping factor (Mu) of 0.005. For training the neural network, CPU, as the main resource for calculation, was used and, on average, it took 5 s to train. This method allowed to achieve a good accuracy of 91.5% with F1 macro- and micro-scores of 0.89 and 0.96, respectively.

Most of the analyzed algorithms had trouble separating squares and rectangles. This can be explained by the unbalanced data set, since those two classes were the smallest ones. Naturally, amber stones are rarely discovered in the form of a square or rectangle, but these classes share some similarities; therefore, the training data set must be intentionally supplemented with new examples of these two shapes.

3.5. Classification Performance

To summarize the results, the time of the computations and the accuracy of the performance were compared for the tested methods. Table 5 and Figure 10 show all of the results for the classification processes, with the timing including object preprocessing and classification operations.

Table 5. Classification performance of the tested methods.

Method	CDF	SVM	EDT	NB	FFNN	KNN	SPD
Time (s)	0.063	0.025	0.022	0.019	0.015	0.012	0.008
Accuracy	75.9%	82.3%	83.9%	82.6%	91.5%	74.5%	78.4%
F1 macro-score	0.62	0.74	0.76	0.77	0.89	0.66	0.73
F1 micro-score	0.86	0.90	0.91	0.90	0.96	0.85	0.88

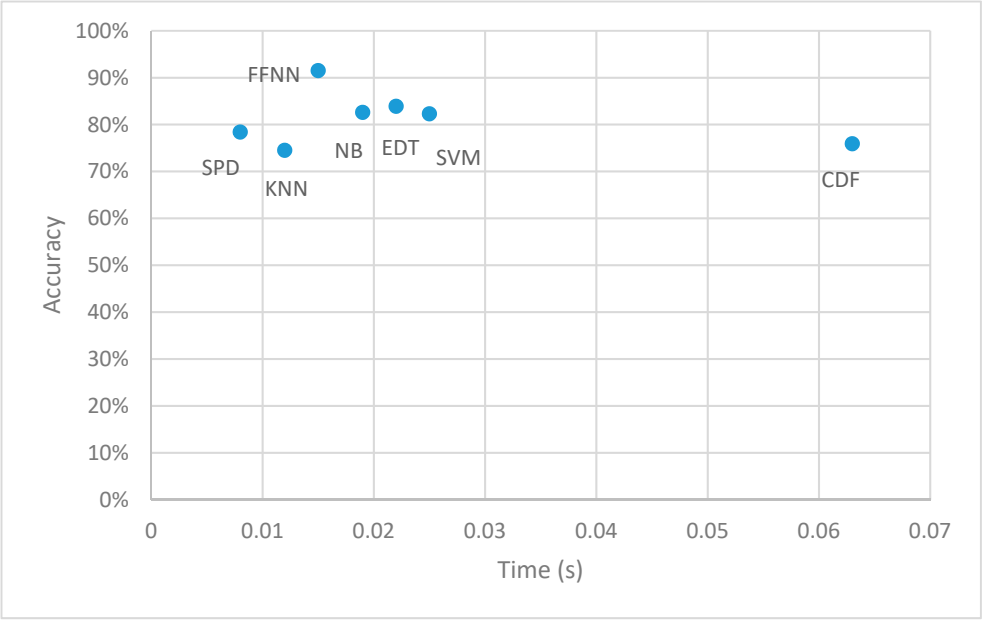


Figure 10. Classification speed vs. accuracy comparison.

An additional evaluation of the model's performance provided us with F1 macro- and micro-scores, which are usually used in the statistical analysis of binary classification. Here, one is the best value and zero is the worst value. The F1 macro-score is the mean of classwise F1 scores, while the F1 micro-score measures the F1 scores of aggregated contributions of all classes. Since square and rectangle shapes are rare in nature, causing an

imbalance of the classes, the F1 score was slightly lower for those classes, thereby causing lower F1 macro-scores.

The proposed SPD method achieved the fastest classification. It required the least computation resources but provided an accuracy of up to 80%. The machine learning methods (i.e., KNN, NB, EDT, SVM, and FFNN) showed comparable classification times between 0.012 and 0.025 s. The fastest among them was KNN with a 74.5% accuracy, while the slowest was SVM, which was also not the best performer from a qualitative point of view. The overall slowest tested method was CDF (0.063 s), because of the specifics of this method in describing an object, and because this model uses three different decision trees, trained using different parameters.

4. Conclusions

The proposed SPD method acted quite well in terms of speed performance, which is one of the most important aspects when working under real-time conditions, e.g., conveyor. However, the accuracy of the classification was 78.4%, which could have been better. EDT, Naïve Bayes, SVM, KNN, and FFNN were tested with the same parameters as SPD. The accuracy of the tested methods (see Figure 9) showed the worst results for KNN (74.5%), while the best results were achieved with FFNN (91.5%). As some classes overlapped with one another, the classic machine learning classifiers (i.e., EDT, Naïve Bayes, SVM, and KNN) showed worse results than FFNN.

A comparison of the classification speed showed that the proposed SPD method is up to 11 times faster compared to the initially referenced method described in [6,7]. FFNN also showed good results in terms of timing, meaning it is applicable in real production processes, since it has an accuracy of more than 90%. Additionally, convolutional neural networks should be considered in future work, as neural networks showed great potential in handling this classification problem.

Author Contributions: Conceptualization, A.O. and M.P.; methodology, A.O. and M.P.; software, A.O. and M.P.; validation, Alfonsas Misevičius and Linas Paulauskas; formal analysis, A.M. and T.S.; investigation, A.O. and M.P.; resources, L.P.; data curation, T.S.; writing—original draft preparation, A.O. and M.P.; writing—review and editing, A.O., M.P. and A.M.; visualization, M.P.; supervision, A.O.; project administration, A.O.; funding acquisition, A.O. and L.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lorencs, A.; Mednieks, I.; Sinica-Sinavskis, J. Simplified Classification of Multispectral Image Fragments. *Elektron. Elektrotech.* **2014**, *20*, 136–139. [CrossRef]
2. Pearson, T.; Moore, D.; A Pearson, J.R. A machine vision system for high speed sorting of small spots on grains. *J. Food Meas. Charact.* **2012**, *6*, 27–34. [CrossRef]
3. Shahrani, S.; Aini, H.; Dzuraidah, A.W.; Mohd, M.M.; Suzaimah, R. Support vector machines for automated classification of plastic bottles. In Proceedings of the 6th International Colloquium Signal Processing and Its Applications (CSPA), Melaka, Malaysia, 21–23 May 2010; pp. 1–5. [CrossRef]
4. O'Farrell, M.; Lewis, E.; Flanagan, C.; Lyons, W.B.; Jackman, N. Design of a system that uses optical-fiber sensors and neural networks to control a large-scale industrial oven by monitoring the food quality online. *IEEE Sens. J.* **2005**, *5*, 1407–1420. [CrossRef]
5. Pearson, T.; Brabec, D.; Haley, S. Color image based sorter for separating red and white wheat. *Sens. Instrum. Food Qual. Saf.* **2008**, *2*, 280–288. [CrossRef]
6. Sinkevičius, S.; Lipnickas, A.; Rimkus, K. Organic shapes classifications by similarity to basic geometric shapes. *Int. J. Comput. Inf. Technol.* **2014**, *3*, 503–507.
7. Sinkevicius, S.; Lipnickas, A.; Rimkus, K. Automatic amber gemstones identification by color and shape visual properties. *Eng. Appl. Artif. Intell.* **2015**, *37*, 258–267. [CrossRef]

8. Jorge, J.; Fonseca, M.J. A Simple Approach to Recognise Geometric Shapes Interactively. Available online: web.tecnico.ulisboa.pt/~{}jorgej/papers/grec99.pdf (accessed on 5 November 2020).
9. Bochkarev, S.O.; Litus, I.B.; Kravchenko, N.S. Irregular Objects. Shape Detection and Characteristic Sizes. Available online: Ceur-ws.org/Vol-1814/paper-04.pdf (accessed on 5 November 2020).
10. Mazzeo, P.L.; Argentieri, A.; De Luca, F.; Spagnolo, P.; Distante, D.; Leo, M.; Carcagnì, P. Convolutional neural networks for recognition and segmentation of aluminum profiles. In Proceedings of the SPIE 11059, Multimodal Sensing: Technologies and Applications, 110590S, Munich, Germany, 21 June 2019.
11. Sarstedt, M.; Mooi, E. *Concise Guide to Market Research*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2014; ISBN 978-3-642-53965-7. [CrossRef]
12. Sinkevicius, S.; Lipnickas, A.; Rimkus, K. Multiclass amber gemstones classification with various segmentation and committee strategies. In Proceedings of the IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS), Berlin, Germany, 12–14 September 2013. Available online: https://ieeexplore.ieee.org/document/6662694 (accessed on 5 November 2020). [CrossRef]
13. Sinkevicius, S.; Lipnickas, A.; Rimkus, K. Amber Gemstones Sorting by Colour. *Elektron. Elektrotech.* **2017**, *23*, 10–14. [CrossRef]