Fragmented Image Classification Using Local and Global Neural Networks: Investigating the Impact of the Quantity of Artificial Objects on Model Performance

Kwabena Frimpong Marfo[1][0000−0003−2226−9097], Małgorzata Przybyła-Kasperek[1][0000−0003−0616−9694], and Piotr Sulikowski[2][0000−0002−8704−8925]

[1] University of Silesia in Katowice, Institute of Computer Science,
Będzińska 39, 41-200 Sosnowiec, Poland
{kmarfo,malgorzata.przybyla-kasperek}@us.edu.pl

[2] West Pomeranian University of Technology in Szczecin, Faculty of Computer Science and Information Technology, ul. Żołnierska 49, 71-210 Szczecin, Poland
piotr.sulikowski@zut.edu.pl

Abstract. This paper addresses the challenge of classifying objects based on fragmented data, particularly when dealing with characteristics extracted from images captured from various angles. The complexity increases when dealing with fragmented images that may partially overlap. The paper introduces a classification model utilizing neural networks, specifically multilayer perceptron (MLP) networks. The key concept involves generating local models based on local tables comprising characteristics extracted from fragmented images. Since the local tables may have different sets of attributes due to varying perspectives, missing attributes in the tables are imputed by introducing artificial objects. The local models, now with identical structures are created and the aggregation of these models into a global model is carried out using weighted averages. The model's efficacy is evaluated against existing literature methods using various metrics, demonstrating superior performance in terms of F-measure and balanced accuracy. Notably, the paper investigates the impact of the number of generated artificial objects on classification quality, revealing that a higher number generally improves results.

Keywords: Fragmented Image Classification · Neural Networks · Artificial Objects · Characteristics Generated From Images.

1 Introduction

Every so often in computer vision and object recognition tasks, the goal is not necessarily to create a virtual representation of an object, but to assign it to a certain class, based on its characteristics. Such a situation occurs in instances such as recognizing the architectural style of a building, the type of vehicle or type of land based on a satellite image. The situation becomes more complicated

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63783-4_21

when not a single image represents an entire object, but rather many fragmented images that may partially overlap. Here, one can think about a set of cameras that perceive an image of an object from different angles. Fragmented images can partially overlap – the cameras can observe (in some part) the same fragment of the object. In this paper, an assumption of not having fragmented images as such, but rather characteristics that have been extracted from these images is made. These object characteristics are stored in decision tables, also known as local tables and may contain common attributes and objects. By common objects in tables, we mean a situation where characteristics extracted from images of the same object are stored in different tables. There may be inconsistencies among tables in that an image in question may have been distorted in some way, resulting in a completely different value on conditional attributes or even decision attribute for the same object.

The paper proposes a classification model based on such fragmented data. The main idea of the model is to use neural networks (specifically MLP networks) to generate local models based on each local table. These models are then aggregated into a global model using trained weights from the local models. However, aggregation of MLP networks is not possible to realize without the local models having the same structure. This constraint can be satisfied by ensuring the presence of the same conditional attributes in all local tables, which, of course is not originally fulfilled due to different cameras observing different parts of an object and, consequently, different set of characteristics being stored in local tables. So, to achieve homogeneity in local tables, it is necessary to modify them before generating local models. This is done by generating artificial objects – supplemented with values for missing attributes. After such modification, local models with identical structures are built from local tables. Local models are then aggregated into a global model. Finally, the global model is refined using a small set of objects.

Figure 1 shows the stages of building the global model. In the first and second steps, characteristics of objects are extracted from fragmented images and local decision tables are created with different sets of conditional attributes – sets of attributes are not necessarily disjoint. It should be noted that this part is not addressed directly in this paper, since the data used retains characteristics of objects extracted from images and were obtained from the repository. Then, in order to unify the local tables, values are imputed for missing attributes in all local tables. For an original object, more than one artificial object can be created, making the cardinality objects in local tables dynamic. The fourth step describes building local neural networks, and the fifth aggregates these local networks into a global model – weights from local models are used for this purpose. Finally, this global model is trained using a small set of objects.

The main contribution of this paper is a proposal of a classification model based on characteristics obtained from fragmented images. Comparing the classification quality of the proposed model with known methods from the literature, it was shown that the proposed model, on average, generates better F-measure and balanced accuracy results. An important result of the paper is to examine

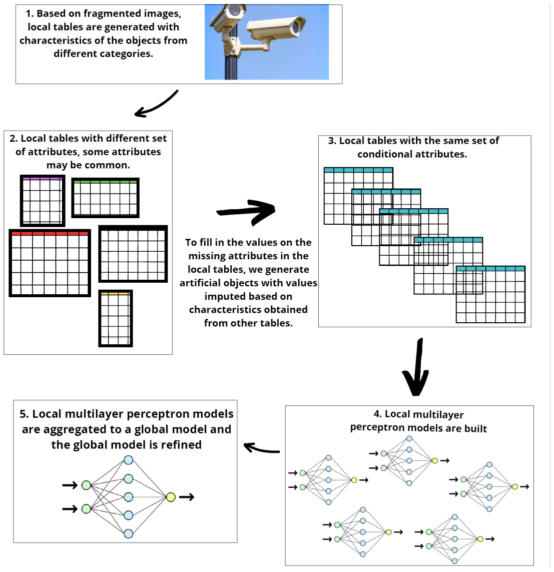
1. Based on fragmented images, local tables are generated with characteristics of the objects from different categories.
2. Local tables with different set of attributes, some attributes may be common.
To fill in the values on the missing attributes in the local tables, we generate artificial objects with values imputed based on characteristics obtained from other tables.
3. Local tables with the same set of conditional attributes.
4. Local multilayer perceptron models are built
5. Local multilayer perceptron models are aggregated to a global model and the global model is refined

Fig. 1. Global model generation stages.

the effect of the number of artificial objects generated on classification quality. Here, usually a higher number of objects improves the quality of classification. The paper also provides some guidance for which data the proposed model is suitable.

An effective and excellent model for image recognition is convolution neural networks [15]. There are many applications of image-based object recognition, among which are handwriting recognition [13], X-ray analysis [5], plant disease identification [3], facial emotion recognition [1]. However, it should be admitted that there are not many papers dedicated to the subject of processing fragmented images. Papers on fragmented images are usually concerned with completing the object that are presented in a fragmentary way on the image – examples are cracks in a roadway [14] or fragments of a plant leaf [2], coloring and fragmentation of image where the objects are located [9]. Identifying fragmented images is an infrequently addressed area of study. This challenge often involves working with sets of images captured from diverse perspectives, where each image is obtained by a distinct camera viewing the object from different angles [11]. When dealing with fragmented image data, recognizing the depicted object becomes more challenging. In a related paper [7], the proposed methodology involves assembling fragmented portions of photos to reconstruct a complete image, aligning and merging components to form a cohesive whole. It is important to note that this approach assumes non-overlapping fragmented data for effective implementation. The first research on classification based on dispersed and fragmented images was presented in the paper [10]. In this paper we present an approach that builds a common model which given the characteristics of an object, can recognize the class from which it comes from.

In Section 2, the proposed classification model is described. Section 3 addresses the data sets that were used and presents the conducted experiments and discussion on obtained results. Section 4 is on conclusions.

2 Model and methods

There is an assumption that features are extracted from fragmented images and stored in a tabular form. More formally, some characteristics of images are available in a dispersed form, that is, in the form of a set of local tables. A set of decision tables $D_i = (U_i, A_i, d)$, $i \in \{1, \dots, n\}$ is available, where U_i is the universe comprising objects – images; A_i is a set of attributes that describe the image; d is a decision attribute – object shown in the image; $A = \bigcup_{i=1}^{n} A_i$ is the union of attributes present in all local tables. Objects and attributes in local tables can be different but some may be common.

The aim is to to generate local neural network models (MLP models) based on each local table. To construct a global model, the structure of such local models must be identical, and this can only be achieved if all local tables have the same sets of attributes. Each table D_i is modified so that the full set of attributes A is included. This is done by generating new objects with completed values on the missing attributes. Suppose the object $\bar{x} \in U_i$ has a decision value v, $d(\bar{x}) = v$, $v \in V^d$, where V^d is the set of values of the decision attribute d and $b \in A \setminus A_i$. For each decision table $D_j, i \neq j$ for which $b \in A_j$ the following values are computed: $MIN^b_{j,v} = min_{x \in U_j, d(x)=v}\ b(x)$, $MAX^b_{j,v} = max_{x \in U_j, d(x)=v}\ b(x)$, $AVG^b_{j,v} = avg_{x \in U_j, d(x)=v}\ b(x)$, $MED^b_{j,v} = median_{x \in U_j, d(x)=v}\ b(x)$. In this way, individual values assigned to attribute b for each local table are derived. The final value, which completes the object $\bar{x}$ in table D_i is determined by applying one of four statistical measures (minimum, maximum, mean, or median) to the local values obtained in the previous step. Consequently, there are sixteen potential combinations with one chosen randomly for determining the value of attribute b. As an illustration, consider a scenario where the maximum is selected for calculating local values, and the median is chosen for the aggregate value. In this case, the determination of the value for attribute b is as follows: $b(\bar{x}) = med_{D_j : b \in A_j} MAX^b_{j,v}$. This method is repeated for each attribute that does not belong to the set A_i but occurs in other local tables. By the procedure described above, one can generate several artificial objects based on an original object. A parameter k is used to determine the number of artificial objects generated based on a single original object. This expanded approach has been tested, and the corresponding results are detailed in the experimental section of the paper. In this way, a set of modified local tables $\bar{D}_i = (\bar{U}_i, A, d)$, $i \in \{1, \dots, n\}$ with equal sets of attributes is obtained.

The local tables $\bar{D}_i$ are used in subsequent steps for training MLP neural networks. The input layer is defined as the cardinality of A. The number of neurons in the output layer corresponds to the number of decision classes, where each neuron determines the probability of the test object belonging to a specific decision class. In the experimental section, the consideration is given to one or two hidden layers. The number of neurons in the hidden layer is determined proportionally to the number of neurons in the input layer, exploring different proportions ranging from 0.25 to 5 times the number of input layer neurons. In the case of two hidden layers, all combinations of neuron numbers are explored, with the first layer being chosen from the set $\{0.25 \times I, 0.5 \times I, 0.75 \times I, 1 \times I, 1.5 \times I, 1.75 \times I, 2 \times I, 2.5 \times I, 2.75 \times I, 3 \times I, 3.5 \times I, 3.75 \times I, 4 \times I, 4.5 \times I, 4.75 \times I, 5 \times I\}$, and the second layer chosen from the set $\{1 \times I, 2 \times I, 3 \times I, 4 \times I, 5 \times I\}$ where I is the number of neurons in the input layer. The ReLU (Rectified Linear Unit) function is employed as the activation function for the hidden layer. For the output layer, the softmax activation function is utilized. The neural network is trained using the back-propagation method, specifically employing a gradient descent method with an adaptive step size. The model employs the categorical cross-entropy loss function in conjunction with the Adam optimizer for optimal performance.

Since all the local models created have the same structure, the global model is created by a weighted sum of the trained weights from local models. Prior to this aggregation, weights from each local model are adjusted by the formula: $\omega_i = ln(\frac{1-e_i}{e_i})$, where e_i is the classification error of the $i-$th local model on the training set $\bar{U}_i$. In summary, the global model is created as follows: initially, the network's structure is specified, then its weights assigned based on the weighted sum of weights from the local models. The final stage involves retraining the global network. For this step, training objects must possess values of all attributes A. This can be a certain set of examples/objects that an expert will describe and classify by capturing all the characteristics of an object at once. In the experiments conducted in this paper, such a validation set derived from the test set.

3 Data sets and results

3.1 Data and measures

The proposed system was tested on three data sets from the UC Irvine Machine Learning Repository [6, 8, 12]. Vehicle Silhouettes – aims to classify vehicle silhouettes into one of four types, considering characteristics extracted from images taken from various angles: eighteen quantitative attributes, four decision classes, 846 objects (592 training, 254 test set). Landsat Satellite – involves classifying earth types in satellite images based on multispectral pixel values in a 3 × 3 neighborhood: thirty-six quantitative attributes, six decision classes, 6435 objects (4435 training, 1000 test set). Dry Bean – focuses on classifying types of beans using characteristics extracted from high-resolution images subjected to segmentation and feature extraction stages: seventeen quantitative attributes, seven decision classes, 13611 objects (9527 training, 4084 test set).

The data pre-processing involved random dispersion into 3, 5, 7, 9, and 11 local tables. Each local table included a reduced set of attributes with all objects from the original table. The data sets exhibited imbalance, with varying object counts across decision classes in both training and test sets. Two variants were considered for each data set: experiments on dispersed imbalanced data and on balanced data modified using the Synthetic Minority Over-sampling Technique (SMOTE) method [4].

The quality of classification was evaluated based on the test set with the following accuracy measures. Classification accuracy measure (*acc*) – a fraction of the total number of objects in the test set that were classified correctly; Recall – an assessment of the classifier's ability to correctly recognize a given class; Precision (Prec.) – a measure of how often the classifier does not make a mistake when classifying an object to a given class, F-measure (F-m.) – an assessment of the classifier's ability to keeping accuracies balanced. F-measure $= 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$; Balanced accuracy – an average value of Recall for all decision classes. Balanced accuracy (*bacc*) ensures that the performance assessment considers the classification accuracy of all classes equally.

The effect of the number of artificial objects created based on the original object from the local table on the quality of classification was tested. The following numbers of artificial objects used were examined $\{1, 2, 3, 4, 5\}$. During the experiments, different structures of local networks with one or two hidden layers were also tested. Moreover, different number of neurons in hidden layers were studied. The following values were tested: for the first hidden layer $\{0.25, 0.5, 0.75, 1, 1.5, 1.75, 2, 2.5, 2.75, 3, 3.5, 3.75, 4, 4.5, 4.75, 5\}\times$ the number of neurons in the input layer; for the second hidden layer $\{1, 2, 3, 4, 5\}\times$ the number of neurons in the input layer. A validation set was obtained by dividing the original test set randomly but in a stratified manner into two equal parts. First, one part is used as the validation set (for re-training process) and the second part is used to assess the quality of classification. Then the roles reverse as the second part acts as the validation set. Finally, both results are averaged. Each experiment is repeated three times; in the following tables, all results given are the average of these three runs.

3.2 Results analysis

Due to space limit, results obtained for all parameters are not shown (however, they will be made available upon request sent to the authors). Tables 1, 2 and 3 show the best (in terms of classification accuracy) results obtained. The tables also show in bold the best result for each of the considered data sets.

The proposed approach was compared with three other approaches. The first approach (MLP ensemble) uses a homogeneous ensemble of MLP networks. The final decision was determined by soft voting since networks generated from the local tables could be not aggregated due to their different structures. The second approach uses an ensemble of classifiers (KNN, DT, NB). This ensemble of classifiers method consists of creating three base classifiers: $k-$nearest neighbors, decision tree and naive bayes classifier based on each local table. The parameter $k = 3$ and the Gini index as a splitting criterion when building decision

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63783-4_21

Table 1. Results of Prec., Recall, F-m., *bacc* and *acc* for the global neural network.

Data set	No. tables	No. artif. obj.	One hidden layer					Two hidden layer				
			Prec.	Recall	F-m.	*bacc*	*acc*	Prec.	Recall	F-m.	*bacc*	*acc*
Vehicle imbalanced	3	1	0.707	0.627	0.599	0.627	0.627	0.718	0.713	0.696	0.685	0.713
		2	0.671	0.681	0.656	0.655	0.681	0.703	0.715	0.701	0.68	0.715
		3	0.665	0.665	0.665	0.665	0.665	0.711	0.713	0.705	0.684	0.713
		4	0.685	0.68	0.665	0.651	0.68	0.738	0.74	0.737	0.721	**0.74**
		5	0.69	0.685	0.669	0.656	0.685	0.726	0.739	0.723	0.705	0.739
	5	1	0.685	0.673	0.658	0.646	0.673	0.713	0.715	0.707	0.692	0.715
		2	0.682	0.689	0.664	0.653	0.689	0.727	0.734	0.724	0.714	**0.734**
		3	0.669	0.671	0.645	0.639	0.671	0.711	0.72	0.714	0.694	0.72
		4	0.647	0.672	0.641	0.631	0.672	0.741	0.732	0.731	0.714	0.732
		5	0.713	0.689	0.655	0.65	0.689	0.695	0.702	0.693	0.67	0.702
	7	1	0.653	0.682	0.65	0.645	0.682	0.735	0.735	0.727	0.708	**0.735**
		2	0.701	0.697	0.674	0.669	0.697	0.728	0.717	0.711	0.691	0.717
		3	0.703	0.696	0.672	0.669	0.696	0.732	0.724	0.722	0.704	0.724
		4	0.701	0.706	0.688	0.682	0.706	0.725	0.73	0.72	0.701	0.73
		5	0.698	0.707	0.689	0.687	0.707	0.711	0.72	0.713	0.691	0.72
	9	1	0.705	0.697	0.689	0.674	0.697	0.746	0.752	0.739	0.719	**0.752**
		2	0.669	0.678	0.662	0.656	0.678	0.728	0.74	0.727	0.706	0.74
		3	0.706	0.717	0.699	0.688	0.717	0.713	0.723	0.708	0.687	0.723
		4	0.683	0.686	0.661	0.663	0.686	0.737	0.744	0.735	0.718	0.744
		5	0.712	0.703	0.681	0.671	0.703	0.733	0.734	0.723	0.704	0.734
	11	1	0.675	0.694	0.672	0.666	0.694	0.74	0.743	0.732	0.717	0.743
		2	0.709	0.714	0.698	0.69	0.714	0.752	0.756	0.744	0.726	**0.756**
		3	0.673	0.688	0.671	0.659	0.688	0.726	0.739	0.718	0.703	0.739
		4	0.658	0.664	0.641	0.634	0.664	0.75	0.744	0.736	0.724	0.744
		5	0.712	0.694	0.678	0.665	0.694	0.73	0.735	0.726	0.711	0.735
Vehicle balanced	3	1	0.69	0.703	0.678	0.678	0.703	0.722	0.726	0.713	0.702	0.726
		2	0.714	0.685	0.664	0.67	0.685	0.713	0.731	0.717	0.699	0.731
		3	0.687	0.677	0.669	0.654	0.677	0.746	0.743	0.734	0.712	0.743
		4	0.693	0.697	0.684	0.676	0.697	0.751	0.761	0.741	0.728	0.761
		5	0.694	0.703	0.683	0.665	0.703	0.756	0.762	0.753	0.736	**0.762**
	5	1	0.699	0.71	0.689	0.674	0.71	0.748	0.748	0.73	0.725	0.748
		2	0.704	0.72	0.698	0.682	0.72	0.728	0.714	0.705	0.694	0.714
		3	0.692	0.717	0.696	0.684	0.717	0.734	0.734	0.721	0.701	0.734
		4	0.733	0.714	0.671	0.685	0.714	0.749	0.749	0.742	0.725	**0.749**
		5	0.686	0.703	0.682	0.675	0.703	0.723	0.739	0.717	0.702	0.739
	7	1	0.715	0.723	0.708	0.692	0.723	0.751	0.759	0.749	0.732	0.759
		2	0.715	0.726	0.708	0.691	0.726	0.755	0.753	0.75	0.731	0.753
		3	0.724	0.735	0.72	0.7	0.735	0.755	0.765	0.753	0.732	0.765
		4	0.713	0.724	0.684	0.679	0.724	0.767	0.756	0.751	0.734	0.756
		5	0.732	0.724	0.711	0.7	0.724	0.769	0.77	0.758	0.741	**0.77**
	9	1	0.74	0.743	0.732	0.715	0.743	0.744	0.745	0.743	0.721	0.745
		2	0.704	0.724	0.703	0.689	0.724	0.718	0.72	0.71	0.692	0.72
		3	0.707	0.711	0.695	0.681	0.711	0.743	0.741	0.739	0.718	0.741
		4	0.699	0.714	0.701	0.682	0.714	0.744	0.757	0.743	0.723	0.757
		5	0.728	0.722	0.704	0.693	0.722	0.757	0.76	0.747	0.732	**0.76**
	11	1	0.706	0.714	0.701	0.681	0.714	0.768	0.768	0.762	0.746	**0.768**
		2	0.709	0.713	0.704	0.68	0.713	0.702	0.722	0.694	0.684	0.722
		3	0.687	0.705	0.689	0.667	0.705	0.75	0.747	0.742	0.727	0.747
		4	0.71	0.71	0.705	0.682	0.71	0.723	0.73	0.721	0.698	0.73
		5	0.726	0.738	0.719	0.703	0.738	0.749	0.755	0.745	0.726	0.755

trees were used. The final decision of the ensemble was also made by soft voting. Both approaches are implemented in the Python programming language using implementations available in the sklearn library.

The results obtained for the proposed approach and the two approaches discussed above are given in Table 4. The best obtained F-measure, balanced accuracy and accuracy values for each data set and dispersed version are presented

Table 2. Results of Prec., Recall, F-m., *bacc* and *acc* for the global neural network.

Data set	No. tables	No. artif. obj.	One hidden layer					Two hidden layer				
			Prec.	Recall	F-m.	*bacc*	*acc*	Prec.	Recall	F-m.	*bacc*	*acc*
Satellite imbalanced	3	1	0.8	0.795	0.783	0.762	0.795	0.806	0.818	0.803	0.775	0.818
		2	0.798	0.796	0.778	0.759	0.796	0.82	0.822	0.818	0.798	0.822
		3	0.796	0.792	0.78	0.759	0.792	0.822	0.829	0.822	0.8	**0.829**
		4	0.782	0.799	0.778	0.758	0.799	0.821	0.828	0.822	0.798	0.828
		5	0.795	0.799	0.786	0.763	0.799	0.819	0.822	0.815	0.792	0.822
	5	1	0.806	0.81	0.798	0.773	0.81	0.818	0.824	0.818	0.795	0.824
		2	0.806	0.805	0.787	0.763	0.805	0.827	0.829	0.826	0.803	0.829
		3	0.789	0.801	0.781	0.761	0.801	0.831	0.833	0.829	0.803	0.833
		4	0.791	0.802	0.785	0.763	0.802	0.836	0.841	0.834	0.809	**0.841**
		5	0.793	0.795	0.779	0.757	0.795	0.832	0.839	0.831	0.803	0.839
	7	1	0.799	0.805	0.792	0.77	0.805	0.817	0.821	0.814	0.792	0.821
		2	0.814	0.809	0.798	0.776	0.809	0.831	0.838	0.831	0.807	0.838
		3	0.811	0.804	0.786	0.767	0.804	0.831	0.84	0.832	0.81	**0.84**
		4	0.8	0.804	0.791	0.768	0.804	0.826	0.833	0.824	0.793	0.833
		5	0.802	0.793	0.777	0.759	0.793	0.837	0.839	0.834	0.808	0.839
	9	1	0.8	0.808	0.799	0.775	0.808	0.831	0.832	0.825	0.803	0.832
		2	0.813	0.81	0.791	0.769	0.81	0.825	0.831	0.821	0.797	0.831
		3	0.806	0.813	0.8	0.778	0.813	0.835	0.839	0.827	0.8	**0.839**
		4	0.806	0.808	0.79	0.766	0.808	0.831	0.836	0.829	0.804	0.836
		5	0.818	0.804	0.781	0.759	0.804	0.834	0.839	0.834	0.807	**0.839**
	11	1	0.808	0.813	0.802	0.777	0.813	0.833	0.837	0.832	0.807	0.837
		2	0.815	0.808	0.792	0.77	0.808	0.832	0.839	0.832	0.809	**0.839**
		3	0.817	0.812	0.793	0.769	0.812	0.825	0.833	0.823	0.796	0.833
		4	0.805	0.81	0.796	0.772	0.81	0.825	0.835	0.825	0.798	0.835
		5	0.8	0.808	0.791	0.769	0.808	0.832	0.835	0.827	0.801	0.835
Satellite balanced	3	1	0.734	0.78	0.749	0.712	0.78	0.799	0.806	0.791	0.758	0.806
		2	0.753	0.77	0.747	0.721	0.77	0.802	0.803	0.797	0.768	0.803
		3	0.787	0.773	0.765	0.745	0.773	0.793	0.8	0.792	0.762	0.8
		4	0.773	0.772	0.758	0.734	0.772	0.805	0.809	0.802	0.776	**0.809**
		5	0.784	0.783	0.774	0.748	0.783	0.807	0.808	0.799	0.772	0.808
	5	1	0.771	0.776	0.766	0.732	0.776	0.812	0.811	0.803	0.776	0.811
		2	0.766	0.777	0.755	0.723	0.777	0.805	0.808	0.798	0.765	0.808
		3	0.791	0.791	0.784	0.757	0.791	0.796	0.803	0.789	0.759	0.803
		4	0.801	0.79	0.776	0.747	0.79	0.798	0.803	0.796	0.769	0.803
		5	0.794	0.791	0.78	0.755	0.791	0.807	0.814	0.806	0.775	**0.814**
	7	1	0.796	0.79	0.781	0.753	0.79	0.804	0.807	0.793	0.763	0.807
		2	0.773	0.791	0.768	0.739	0.791	0.808	0.814	0.803	0.774	**0.814**
		3	0.771	0.785	0.768	0.738	0.785	0.812	0.814	0.81	0.784	**0.814**
		4	0.811	0.791	0.771	0.749	0.791	0.807	0.813	0.806	0.778	0.813
		5	0.789	0.78	0.772	0.746	0.78	0.809	0.814	0.807	0.778	**0.814**
	9	1	0.778	0.788	0.764	0.734	0.788	0.818	0.822	0.813	0.786	**0.822**
		2	0.786	0.786	0.775	0.745	0.786	0.808	0.81	0.8	0.77	0.81
		3	0.782	0.78	0.764	0.737	0.78	0.805	0.815	0.803	0.773	0.815
		4	0.787	0.787	0.771	0.742	0.787	0.812	0.819	0.807	0.775	0.819
		5	0.781	0.792	0.771	0.742	0.792	0.803	0.81	0.801	0.769	0.81
	11	1	0.79	0.798	0.785	0.756	0.798	0.812	0.815	0.809	0.783	**0.815**
		2	0.778	0.791	0.773	0.739	0.791	0.803	0.81	0.803	0.773	0.81
		3	0.801	0.795	0.773	0.745	0.795	0.795	0.805	0.795	0.758	0.805
		4	0.79	0.788	0.775	0.745	0.788	0.807	0.812	0.804	0.779	0.812
		5	0.782	0.791	0.774	0.744	0.791	0.797	0.812	0.8	0.77	0.812

in bold in the table. These three measures were chosen for analysis as F-measure and balanced accuracy best illustrate the model's overall ability to correctly identify all decision classes and balance between precision and recall. The accuracy measure was also compared, but it is less significant in general, as it can lead to incorrect conclusions in the case of imbalanced data. As can be seen in the vast majority of cases, the proposed approach gives better results for all three

Table 3. Results of Prec., Recall, F-m., *bacc* and *acc* for the global neural network.

Data set	No. tables	No. artif. obj.	One hidden layer					Two hidden layer				
			Prec.	Recall	F-m.	*bacc*	*acc*	Prec.	Recall	F-m.	*bacc*	*acc*
Dry Bean imbalanced	3	1	0.913	0.912	0.912	0.923	0.912	0.921	0.92	0.92	0.931	0.92
		2	0.918	0.917	0.917	0.927	0.917	0.921	0.921	0.921	0.931	**0.921**
		3	0.912	0.911	0.911	0.921	0.911	0.92	0.919	0.919	0.93	0.919
		4	0.914	0.913	0.913	0.922	0.913	0.92	0.92	0.92	0.93	0.92
		5	0.915	0.914	0.914	0.925	0.914	0.919	0.919	0.919	0.93	0.919
	5	1	0.912	0.912	0.912	0.922	0.912	0.918	0.918	0.918	0.928	0.918
		2	0.916	0.915	0.915	0.925	0.915	0.92	0.92	0.919	0.93	**0.92**
		3	0.914	0.913	0.913	0.923	0.913	0.918	0.917	0.917	0.927	0.917
		4	0.914	0.914	0.914	0.924	0.914	0.918	0.918	0.917	0.927	0.918
		5	0.915	0.915	0.915	0.925	0.915	0.918	0.918	0.918	0.928	0.918
	7	1	0.914	0.914	0.913	0.923	0.914	0.917	0.917	0.917	0.928	0.917
		2	0.914	0.914	0.914	0.924	0.914	0.919	0.919	0.919	0.93	**0.919**
		3	0.914	0.913	0.913	0.923	0.913	0.915	0.915	0.915	0.925	0.915
		4	0.913	0.913	0.913	0.923	0.913	0.916	0.916	0.916	0.927	0.916
		5	0.913	0.913	0.913	0.922	0.913	0.916	0.915	0.915	0.924	0.915
	9	1	0.915	0.914	0.914	0.924	0.914	0.916	0.915	0.915	0.925	0.915
		2	0.914	0.913	0.913	0.923	0.913	0.918	0.918	0.918	0.929	**0.918**
		3	0.914	0.913	0.913	0.923	0.913	0.915	0.915	0.915	0.924	0.915
		4	0.913	0.912	0.912	0.922	0.912	0.914	0.913	0.913	0.922	0.913
		5	0.913	0.912	0.912	0.921	0.912	0.915	0.915	0.915	0.924	0.915
	11	1	0.914	0.913	0.913	0.922	0.913	0.915	0.914	0.914	0.924	0.914
		2	0.914	0.914	0.914	0.924	0.914	0.918	0.918	0.918	0.928	**0.918**
		3	0.912	0.911	0.911	0.92	0.911	0.909	0.908	0.908	0.916	0.908
		4	0.913	0.912	0.912	0.922	0.912	0.912	0.912	0.912	0.92	0.912
		5	0.913	0.913	0.913	0.922	0.913	0.91	0.91	0.91	0.918	0.91
Dry Bean balanced	3	1	0.913	0.912	0.912	0.922	0.912	0.92	0.919	0.919	0.93	0.919
		2	0.916	0.915	0.915	0.927	0.915	0.924	0.923	0.923	0.935	**0.923**
		3	0.912	0.911	0.911	0.921	0.911	0.92	0.92	0.919	0.931	0.92
		4	0.916	0.916	0.916	0.926	0.916	0.92	0.919	0.919	0.929	0.919
		5	0.914	0.914	0.914	0.923	0.914	0.92	0.92	0.919	0.929	0.92
	5	1	0.916	0.916	0.916	0.926	0.916	0.918	0.917	0.917	0.927	0.917
		2	0.917	0.916	0.916	0.928	0.916	0.921	0.921	0.921	0.932	**0.921**
		3	0.914	0.913	0.913	0.923	0.913	0.92	0.919	0.919	0.929	0.919
		4	0.914	0.913	0.913	0.923	0.913	0.918	0.918	0.918	0.929	0.918
		5	0.918	0.917	0.917	0.927	0.917	0.919	0.919	0.919	0.928	0.919
	7	1	0.913	0.912	0.911	0.922	0.912	0.919	0.919	0.919	0.93	**0.919**
		2	0.914	0.914	0.914	0.923	0.914	0.918	0.918	0.918	0.928	0.918
		3	0.914	0.913	0.913	0.922	0.913	0.918	0.918	0.917	0.927	0.918
		4	0.913	0.912	0.912	0.921	0.912	0.919	0.918	0.918	0.928	0.918
		5	0.914	0.913	0.913	0.922	0.913	0.918	0.918	0.917	0.927	0.918
	9	1	0.913	0.913	0.913	0.922	0.913	0.917	0.917	0.916	0.926	0.917
		2	0.908	0.907	0.907	0.915	0.907	0.919	0.919	0.919	0.929	**0.919**
		3	0.912	0.912	0.912	0.921	0.912	0.916	0.916	0.916	0.925	0.916
		4	0.914	0.913	0.913	0.923	0.913	0.917	0.917	0.917	0.926	0.917
		5	0.913	0.912	0.912	0.921	0.912	0.918	0.917	0.917	0.926	0.917
	11	1	0.912	0.912	0.912	0.922	0.912	0.917	0.917	0.917	0.926	0.917
		2	0.91	0.909	0.909	0.918	0.909	0.916	0.915	0.914	0.925	0.915
		3	0.913	0.912	0.912	0.921	0.912	0.92	0.92	0.919	0.929	**0.92**
		4	0.912	0.911	0.911	0.92	0.911	0.917	0.916	0.916	0.926	0.916
		5	0.913	0.913	0.913	0.922	0.913	0.914	0.914	0.913	0.922	0.914

compared measures. However, in the case of the Satellite data set, the proposed approach does not perform well. This is due to the data having the greatest variation in attributes present in local tables(very few overlapping attributes). To conclude, when a camera points at an object and generates attributes the majority of are not present in any other local tables, then the proposed approach does not perform better than the classifier ensemble approach. However, in other cases, when the cameras are more densely arranged, overlapping in terms of attributes then the proposed approach definitely performs better. It should also be noted that the number of objects in the training set does not affect the quality of classification generated by the proposed approach, i.e. for both small and large training sets the proposed approach gives good results.

Now, the results for all three measures generated by the analyzed approaches will be compared. To prove that the obtained differences in F-measure values are significant, the Friedman test was performed. Three dependent samples of 30 observations was used, with the test confirming that there is a statistically significant difference in the F-measure obtained for the three approaches considered, $\chi^2(29, 2) = 10.034, p = 0.007$. Additionally, comparative box-plot for the F-measure with three methods was created (Figure 2). As can be observed, on average, the values of the F-measure for the proposed approach are the largest. The post-hoc Dunn Bonferroni test was also performed which confirmed a significant difference in average F-measure values between the three approaches. The results (significant were presented in bold) can be found in Table 5. In the end, it can be said that the proposed approach improves the quality of classification compared to approaches known from the literature in terms of the F-measure.

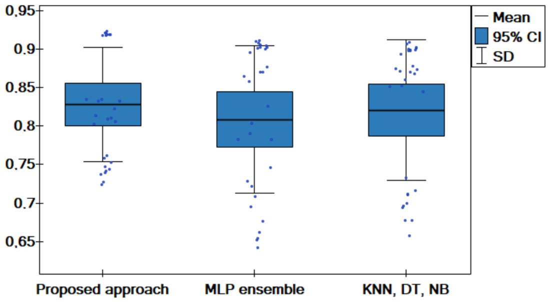

Fig. 2. Comparison of F-measure obtained for approaches: the proposed approach with global model; homogeneous ensemble with MLP networks (MLP ensemble) and the ensemble of classifiers: $k-$nearest neighbors, decision tree and naive bayes classifier (KNN, DT, NB).

Next the Friedman test was performed in order to show that the obtained differences in balanced accuracy values are significant. For balanced accuracy, the Friedman statistics was $7.983, p = 0.018$ indicating that there is a statistically significant difference in the balanced accuracy obtained for the three approaches

Table 4. Results of Prec., Recall, F-m., *bacc* and *acc* for the global neural network; homogeneous ensemble with MLP networks (MLP ensemble) and the ensemble of classifiers (KNN, DT, NB).

Data set	No. tables	The proposed approach					MLP ensemble					KNN, DT, NB				
		Prec.	Recall	F-m.	*bacc*	*acc*	Prec.	Recall	F-m.	*bacc*	*acc*	Prec.	Recall	F-m.	*bacc*	*acc*
Vehicle imbalanced	3	0.738	0.74	**0.737**	**0.721**	**0.74**	0.725	0.74	0.728	0.715	**0.74**	0.701	0.709	0.694	0.686	0.709
	5	0.727	0.734	**0.724**	0.714	**0.734**	0.761	0.724	0.695	**0.722**	0.724	0.718	0.709	0.696	0.69	0.709
	7	0.735	0.735	**0.727**	0.708	0.735	0.73	0.74	0.722	**0.709**	**0.74**	0.69	0.693	0.677	0.67	0.693
	9	0.746	0.752	**0.739**	**0.719**	**0.752**	0.655	0.685	0.652	0.658	0.685	0.708	0.717	0.699	0.694	0.717
	11	0.752	0.756	**0.744**	**0.726**	**0.756**	0.67	0.685	0.676	0.66	0.685	0.705	0.685	0.677	0.672	0.685
Vehicle balanced	3	0.756	0.762	**0.753**	**0.736**	**0.762**	0.742	0.756	0.746	0.729	0.756	0.733	0.732	0.716	0.705	0.732
	5	0.749	0.749	**0.742**	**0.725**	**0.749**	0.582	0.717	0.642	0.661	0.717	0.726	0.728	0.711	0.698	0.728
	7	0.769	0.77	**0.758**	**0.741**	**0.77**	0.711	0.728	0.708	0.699	0.728	0.748	0.752	0.733	0.721	0.752
	9	0.757	0.76	**0.747**	**0.732**	**0.76**	0.66	0.689	0.654	0.649	0.689	0.718	0.728	0.712	0.7	0.728
	11	0.768	0.768	**0.762**	**0.746**	**0.768**	0.683	0.685	0.662	0.656	0.685	0.667	0.677	0.657	0.646	0.677
Satellite imbalanced	3	0.822	0.829	0.822	0.8	0.829	0.838	0.849	0.826	0.8	0.849	0.868	0.87	**0.868**	**0.848**	**0.87**
	5	0.836	0.841	0.834	0.809	0.841	0.843	0.842	0.804	0.783	0.842	0.863	0.864	**0.86**	**0.835**	**0.864**
	7	0.831	0.84	0.832	0.81	0.84	0.758	0.835	0.79	0.768	0.835	0.855	0.857	**0.852**	**0.823**	**0.857**
	9	0.834	0.839	0.834	0.807	0.839	0.75	0.829	0.783	0.758	0.829	0.856	0.858	**0.851**	**0.82**	**0.858**
	11	0.832	0.839	0.832	0.809	0.839	0.754	0.828	0.783	0.757	0.828	0.851	0.854	**0.844**	**0.811**	**0.854**
Satellite balanced	3	0.805	0.809	0.802	0.776	0.809	0.865	0.868	0.864	0.839	0.868	0.879	0.872	**0.874**	**0.859**	**0.872**
	5	0.807	0.814	0.806	0.775	0.814	0.875	0.88	**0.876**	0.851	**0.88**	0.877	0.871	0.873	**0.856**	0.871
	7	0.812	0.814	0.81	0.784	0.814	0.873	0.869	0.87	0.859	0.869	0.881	0.878	**0.878**	**0.861**	**0.878**
	9	0.818	0.822	0.813	0.786	0.822	0.858	0.86	0.858	0.838	0.86	0.874	0.871	**0.871**	**0.851**	**0.871**
	11	0.812	0.815	0.809	0.783	0.815	0.873	0.87	**0.87**	**0.856**	**0.87**	0.871	0.87	**0.87**	0.848	**0.87**
Dry Bean imbalanced	3	0.921	0.921	**0.921**	**0.931**	**0.921**	0.911	0.91	0.91	0.92	0.91	0.908	0.906	0.906	0.916	0.906
	5	0.92	0.92	**0.919**	**0.93**	**0.92**	0.903	0.903	0.902	0.908	0.903	0.904	0.902	0.901	0.908	0.902
	7	0.919	0.919	**0.919**	**0.93**	**0.919**	0.902	0.901	0.9	0.906	0.901	0.901	0.899	0.899	0.905	0.899
	9	0.918	0.918	**0.918**	**0.929**	**0.918**	0.898	0.896	0.895	0.9	0.896	0.897	0.894	0.893	0.897	0.894
	11	0.918	0.918	**0.918**	**0.928**	**0.918**	0.902	0.901	0.901	0.907	0.901	0.902	0.9	0.9	0.905	0.9
Dry Bean balanced	3	0.924	0.923	**0.923**	**0.935**	**0.923**	0.911	0.911	0.911	0.926	0.911	0.91	0.909	0.909	0.919	0.909
	5	0.921	0.921	**0.921**	**0.932**	**0.921**	0.907	0.907	0.907	0.919	0.907	0.9	0.899	0.898	0.907	0.899
	7	0.919	0.919	**0.919**	**0.93**	**0.919**	0.906	0.905	0.905	0.917	0.905	0.901	0.9	0.899	0.909	0.9
	9	0.919	0.919	**0.919**	**0.929**	**0.919**	0.903	0.902	0.902	0.913	0.902	0.899	0.898	0.898	0.907	0.898
	11	0.92	0.92	**0.919**	**0.929**	**0.92**	0.905	0.904	0.904	0.916	0.904	0.903	0.903	0.902	0.912	0.903

considered in the paper. Additionally, comparative box-plot for the balanced accuracy was created (Figure 3). As can be seen also here, the average of balanced accuracy is the highest for the proposed approach. The post-hoc Dunn Bonferroni test confirmed a significant difference in balanced accuracy values between one pair: the proposed approach & MLP ensemble with $p = 0.035$. So it can be

Table 5. p-values for the post-hoc Dunn Bonferroni test for F-measure

	Proposed approach	MLP ensemble	KNN, DT, NB
Proposed approach	–	**0.011**	**0.035**
MLP ensemble	**0.011**	–	1
KNN, DT, NB	**0.035**	1	–

concluded that both the proposed approach and the ensemble of classifiers KNN, DT, NB get the best balanced accuracy results.

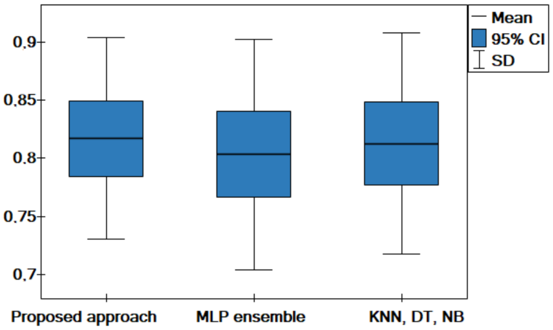

Fig. 3. Comparison of balanced accuracy obtained for approaches: the proposed approach with global model; homogeneous ensemble with MLP networks (MLP ensemble) and the ensemble of classifiers: $k-$nearest neighbors, decision tree and naive bayes classifier (KNN, DT, NB).

As we know, accuracy values can be deceptive and often do not take minority classes into account, nonetheless, comparative analysis was carried out for this measure. For accuracy, the Friedman statistics was $6.889, p = 0.032$ indicating a reject of the null hypothesis, but as can be seen in Figure 4, the average accuracy values are similar. Also, the post-hoc Dunn Bonferroni test did not confirm a significant difference between any pair of approaches. Thus, as a conclusion, it can be confirmed that the proposed approach on average improves values of F-measure and balanced accuracy. Of course, comparing the results obtained for each data set separately (Table 4), it can be seen that for some data sets this improvement is significant, while for others the proposed approach does not improve the quality of classification. The situation in which the proposed approach does not do well is when we have a very large number of conditional/descriptive attributes and a relatively small number of local tables/cameras.

Next, the impact of the number of artificial objects used in the proposed approach on the quality of classification is analyzed. The comparison, as before, was be made using three measures: F-measure, balanced accuracy and accuracy. To test whether the different number of artificial objects used generated a significant difference in results, five groups were created 1AO, 2AO, 3AO, 4AO, 5AO

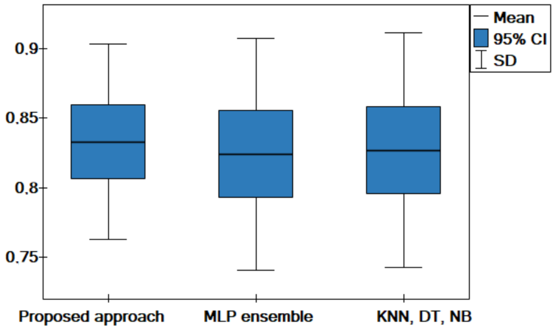

Fig. 4. Comparison of accuracy obtained for approaches: the proposed approach with global model; homogeneous ensemble with MLP networks (MLP ensemble) and the ensemble of classifiers: $k-$nearest neighbors, decision tree and naive bayes classifier (KNN, DT, NB).

– results obtained for 1, 2, 3, 4, 5 artificial objects used. Each group contained 60 observations (results obtained for all data sets and all versions of dispersion). The Friedman test confirmed a statistically significant difference in the F-measure obtained for the five groups considered, $\chi^2(59, 4) = 10.830, p = 0.029$. The Wilcoxon each-pair test confirmed the significant differences between the average F-measure values for the following pairs: 1AO & 4AO with $p = 0.01$, 1AO & 5AO with $p = 0.001$, 3AO & 4AO with $p = 0.005$, 3AO & 5AO with $p = 0.0002$. Additionally, a comparative graph for the F-measure with different number of artificial objects used was created (Figure 5). It can be seen that the results obtained for using 4 and 5 artificial objects are better than those obtained with fewer artificial objects.

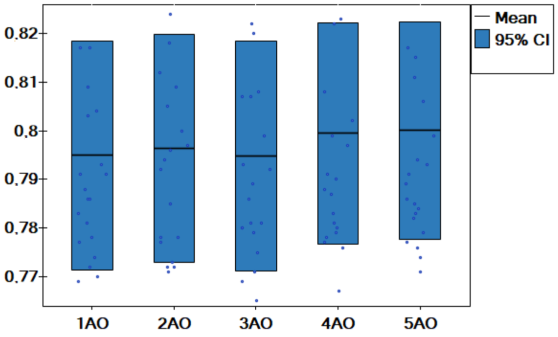

Fig. 5. Comparison of F-measure accuracy obtained for the proposed approach with 1, 2, 3, 4, 5 artificial objects generated (1AO, 2AO, 3AO, 4AO, 5AO).

Similar analyses were performed for balanced accuracy and accuracy. Friedman's test confirmed a significant difference for accuracy with statistics $15.501, p = 0.004$. The Wilcoxon each-pair test confirmed the significant differences between the average accuracy values for the following pairs: 1AO & 4AO with $p = 0.044$, 1AO & 5AO with $p = 0.002$, 2AO & 5AO with $p = 0.009$, 3AO & 4AO with $p = 0.002$, 3AO & 5AO with $p = 0.0001$. Also, the comparative graph for accuracy values (Figure 6) proves that for 5 and 4 artificial objects the generated results are better. For the balanced accuracy, the Friedman test does not confirm a significant difference in the mean value for different numbers of artificial objects used. Nonetheless, it can be concluded that larger numbers of artificial objects used to build the global model improve its quality. Thus, the proposed method of generating artificial objects with missing values in local tables has a positive effect on the model accuracy, and a larger number of artificial objects increases the quality of the model.

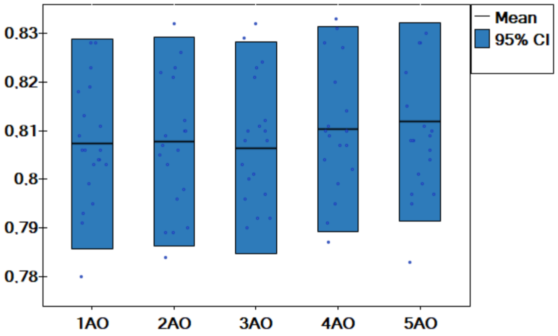

Fig. 6. Comparison of accuracy accuracy obtained for the proposed approach with 1, 2, 3, 4, 5 artificial objects generated (1AO, 2AO, 3AO, 4AO, 5AO).

4 Conclusion

In this paper, a situation in which local decision tables containing partial characteristics of objects from fragmented images was considered. Then, based on the local tables, tables with artificial objects were generated by filling the missing values of characteristics. Finally, local models were built based on local tables, which were finally aggregated into a global model.

In conclusion, it is important to note that while the proposed approach consistently enhances classification quality on average, its efficacy may vary across different data sets. The method excels particularly in scenarios with a substantial number of attributes and a relatively small number of local tables/cameras.

In summary, our study establishes the superiority of the proposed approach in terms of F-measure and balanced accuracy, showcasing its potential to elevate classification performance. The positive correlation between the number of artificial objects and model quality reinforces the practical applicability of our method. These findings contribute valuable knowledge to the field of classification based on fragmented images, offering a promising avenue for further research and application in real-world scenarios.

References

1. Canal, F. Z., Müller, T. R., Matias, J. C., Scotton, G. G., de Sa Junior, A. R., Pozzebon, E., Sobieranski, A. C. (2022). A survey on facial emotion recognition techniques: A state-of-the-art literature review. Information Sciences, 582, 593-617.
2. Chaki, J., Dey, N., Moraru, L., Shi, F. (2019). Fragmented plant leaf recognition: Bag-of-features, fuzzy-color and edge-texture histogram descriptors with multi-layer perceptron. Optik, 181, 639-650.
3. Chen, J., Chen, J., Zhang, D., Sun, Y., Nanehkaran, Y. A. (2020). Using deep transfer learning for image-based plant disease identification. Computers and Electronics in Agriculture, 173, 105393.
4. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: synthetic minority over-sampling technique. J. Artif. Intell. Res., 16:321–357, 2002.
5. Çallı, E., Sogancioglu, E., van Ginneken, B., van Leeuwen, K. G., Murphy, K. (2021). Deep learning for chest X-ray analysis: A survey. Medical Image Analysis, 72, 102125.
6. Dua, D. and Graff, C.: UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science, (2019)
7. Fornasier, M., Toniolo, D.: Fast, robust and efficient 2D pattern recognition for re-assembling fragmented images. Pattern Recognition, 38(11), 2074–2087, (2005)
8. Koklu, M., Ozkan, I. A.: Multiclass classification of dry beans using computer vision and machine learning techniques. Computers and Electronics in Agriculture, 174, 105507, (2020)
9. Lin, G., Tang, Y., Zou, X., Cheng, J., Xiong, J. (2020). Fruit detection in natural environment using partial shape matching and probabilistic Hough transform. Precision Agriculture, 21, 160-177.
10. Marfo, K. F., Przybyła-Kasperek, M.: Radial Basis Function Neural Network with a Centers Training Stage for Prediction Based on Dispersed Image Data. In International Conference on Computational Science (pp. 89-103). Cham: Springer Nature Switzerland, (2023, June)
11. Shelepin, Y. E., Chikhman, V. N., Foreman, N.: Analysis of the studies of the perception of fragmented images: global description and perception using local features. Neuroscience and behavioral physiology, 39(6), 569–580, (2009)
12. Siebert, J. P.: Vehicle Recognition Using Rule Based Methods, Turing Institute Research Memorandum TIRM-87-0.18, March 1987.
13. Vashist, P. C., Pandey, A., Tripathi, A. (2020, January). A comparative study of handwriting recognition techniques. In 2020 International Conference on Computation, Automation and Knowledge Management (ICCAKM) (pp. 456-461). IEEE.
14. Wu, L., Mokhtari, S., Nazef, A., Nam, B., Yun, H. B. (2016). Improvement of crack-detection accuracy using a novel crack defragmentation technique in image-based road assessment. Journal of Computing in Civil Engineering, 30(1), 04014118.
15. Xin, M., Wang, Y. (2019). Research on image classification model based on deep convolution neural network. EURASIP Journal on Image and Video Processing, 2019, 1-11.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63783-4_21