symmetry

MDPI

Article

A Model for Generating Workplace Procedures Using a CNN-SVM Architecture

Justyna Patalas-Maliszewska [1,*] and Daniel Halikowski [2]

[1] Institute of Computer Science and Production Management, Faculty of Mechanical Engineering, University of Zielona Góra, ul. Licealna 9, 65-417 Zielona Góra, Poland

[2] Institute of Technical Science, University of Applied Science in Nysa, ul. Armii Krajowej 7, 48-300 Nysa, Poland; daniel.halikowski@pwsz.nysa.pl

* Correspondence: j.patalas@iizp.uz.zgora.pl

Received: 22 August 2019; Accepted: 5 September 2019; Published: 10 September 2019

check for updates

Abstract: (1) Background: Improving the management and effectiveness of employees' learning processes within manufacturing companies has attracted a high level of attention in recent years, especially within the context of Industry 4.0. Convolutional Neural Networks with a Support Vector Machine (CNN-SVM) can be applied in this business field, in order to generate workplace procedures. To overcome the problem of usefully acquiring and sharing specialist knowledge, we use CNN-SVM to examine features from video material concerning each work activity for further comparison with the instruction picture's features. (2) Methods: This paper uses literature studies and a selected workplace procedure: repairing a solid and using a fuel boiler as the benchmark dataset, which contains 20 s of training and a test video, in order to provide a reference model of features for a workplace procedure. In this model, the method used is also known as Convolutional Neural Networks with Support Vector Machine. This method effectively determines features for the further comparison and detection of objects. (3) Results: The innovative model for generating a workplace procedure, using CNN-SVM architecture, once built, can then be used to provide a learning process to the employees of manufacturing companies. The novelty of the proposed methodology is its architecture, which combines the acquisition of specialist knowledge and formalising and recording it in a useful form for new employees in the company. Moreover, three new algorithms were created: an algorithm to match features, an algorithm to detect each activity in the workplace procedure, and an algorithm to generate an activity scenario. (4) Conclusions: The efficiency of the proposed methodology can be demonstrated on a dataset comprising a collection of workplace procedures, such as the repair of the solid fuel boiler. We also highlighted the impracticality for managers of manufacturing companies to support learning processes in a company, resulting from a lack of resources to teach new employees.

Keywords: generation of a workplace procedure; CNN-SVM architecture; employee learning processes

1. Introduction

Improving the management and effectiveness of employees' learning processes in manufacturing companies is one of the needs of manufacturing enterprises in the context of Industry 4.0. Expert workers, that is, technical specialists, acquire a unique knowledge about production processes over the years [1]. The transfer of such specialist knowledge to new employees, without the involvement of experienced employees, requires the development of solutions supporting this process. Automating the process of transferring specialist knowledge, within an enterprise, is not only a modern solution to employees' learning processes, but also allows business managers to retain the knowledge of employees who leave a company. The process of acquiring knowledge, and then formalising and transferring it into a useful form, is difficult. In this article, the proposed model for the generation of workplace procedures, using Convolutional Neural Networks with a Support Vector Machine (CNN-SVM) architecture, combines the acquisition of specialist knowledge, on the one hand, and also allows the formalisation and recording of this knowledge in such a way that it is useful for new employees in the company, according to the two essential elements of a knowledge-based system (KBS) [2]. In our approach, specialist knowledge is acquired using video recordings of company specialists in action. These recordings will allow any knowledge about the activities performed to be passed on.

The second element of our approach is the formalisation of the acquired knowledge with the help of a video recording and its further distribution in the company. In the context of such as image data analysis, Support Vector Machine (SVM), Principal Component Analysis (PCA), and Linear Discriminant Analysis (LDA) have been used [3]. New object recognition models, such as the HMAX [4,5], the convolutional networks [6], and a number of deep learning models [7], have been successfully applied in solutions supporting employees in manufacturing companies in the context of Industry 4.0. Convolutional Neural Networks (CNNs) and Support Vector Machines (SVMs) are introduced to solve the problem of generating of real-time workplace procedures, in order to support the learning processes in a company where the lack of resources prevents the teaching of new employees.

The main focus of this work was to use a Convolutional Neural Network with a Support Vector Machine (CNN-SVM), in order to generate workplace instructions. In this paper, we used the CNN neural network with an SVM classifier, so that the knowledge of an experienced worker can be extracted and classified based on the video material. In our case, this process was done to study the repair of a solid fuel boiler. In the first step, the video material is divided into stages, according to established rules that will allow individual images to be assigned to a given step. Next, based on the features obtained for a given step in the video material, and using CNN-SVM, a reference model of the features will be constrcuted for each step of the work performed at work. In addition, a model of the characteristics of the station instructions (illustration) will be built with the help of CNN. In order to obtain an automatic workplace instruction, an algorithm for establishing an activity scenario has been proposed, the use of which allows comparison of the models both of the features and of the automatic generation of workstation instructions.

This paper is organised as follows. Section 2 describes an analysis of the research literature; Section 3 shows the research model. In Section 4, the research experiments are presented along with the results and an evaluation of the performance. In Section 5, the relevant discussion is presented. Section 6 provides the conclusions and direction for further work.

2. Methods and Techniques for Generating Workplace Procedures

Industry 4.0 provides new opportunities to support the training process of new employees in the company. The use, in manufacturing, of virtual reality (VR), augmented reality (AR), and mixed reality (MR) in the education process [8] has undoubted advantages, including providing operators with adequate feedback for their actions [9]. However, it also makes it necessary for a company to invest in such solutions. Our proposed approach allows the expert operators' specialist knowledge of assembly tasks to be transferred to new operators using a friendly set of instructions for the new employees in the form of a sequence of animated images.

In order to formalise the acquired specialist knowledge with the use of video recordings, it is necessary to apply image analysis methods. Previously, such methods dealt with the extraction of data or information from images [10]. The result of this type of analysis is not the image but the data received, such as in the numerical form [11,12]. There are several techniques to analyse images for their distinguishing features. These techniques include Principal Component Analysis (PCA), the Singular Value Decomposition algorithm (SVD), and Linear Discriminant Analysis (LDA). PCA is a method for reducing dimensions and is used to reduce the size of large data sets by converting a set of variables to a smaller set that contains most of the information of the large set [10,13–15]. PCA relies on processing a large amount of information contained in mutually correlated input data into a set of new data, with orthogonally corresponding features. SVD decomposes the given matrix into three parts: the left singular matrix, the right singular matrix, and a singular matrix [16,17]. LDA is a dimension reduction technique in machine learning. It is a supervised learning method where labelled data are used for training [18]. LDA is most commonly used as a dimensionality reduction technique in the pre-processing step for pattern-classification and machine learning applications. In this method, the dimensions of the projection subspace are related to the number of data classes and remain independent of the data's dimensions. Linear discriminant analysis is the projection of the normal vector in the linear discriminant hyperplane and renders the distance between the classes as the largest and the distance within the classes as the smallest [18].

SVD has been successfully applied to signal processing (speaker verification systems) [19], image processing (noise filtering, watermarking) [20], image compression [21], and big data, e.g., in genomic signal processing (transforming genome-wide expression data and analysis of genome-scale data) [22,23]. PCA is also used in signal processing (diagnosing and predicting machine damage) [24,25], image processing (extracting image features) [26], and analysis (pattern recognition and image compression) [27]. The case for LDA looks very similar. Here, this technique is also applied to image processing and analysis (facial recognition) [28,29]. There is a research gap in the literature on the application of SVD, PCA, and LDA techniques to the generation of workstation instructions based on previously recorded video material.

In the context of the image analysis needed in companies compatible with the concept of Industry 4.0, the most frequent methods used are classification algorithms, the Decision Tree (DT), K-Nearest Neighbour (KNN), Perceptron, Random Forest, Bayes Algorithm, and the Support Vector Machine (SVM). DTs are tree-structured models for classification and regression [30]. The Decision Tree is a tree in which the nodes correspond to the tests carried out on the values of the attributes of the rules. The branches are possible results of such tests, and the leaves represent the decision part. A decision tree is a structure in which each internal node, not the leaf, is labelled with an input feature. The arcs coming from a node labelled with this feature are labelled with each of the possible values of the feature. Each leaf of the tree is labelled with a class or a probability distribution over the classes [31]. The K Nearest Neighbours method belongs to the group of lazy algorithms, i.e., those that do not form the internal representation of the training data but look for a solution only when the test pattern appears for such as classification. This method stores all training patterns for which the distance of the test pattern is determined [32]. When using a neural network as a classifier, the input signals are image features that have been previously detected. The number of input neurons is equal to the number of features analysed, while the outputs are as numerous as the image classes. The connection of the input vector with the neurons of the output layer is usually complete, i.e., each input node is connected to each output neuron. Connections are represented by the matrix of connection weights W [33,34]. Random Forest is a machine learning method for classification, regression and other tasks that involve the construction of many decision trees during learning and generating a class that influences or controls the classes (classification) or predicted means (regression) of individual trees [35]. The Algorithm of bi- and multi-class classification is also known as Random Forest and Decision Forest. In both cases, the idea of functioning is almost identical and is based on classification using a group of decision trees; the biggest difference in the construction of both algorithms is the so-called bootstrap. The simple Bayesian classifier is an uncomplicated, probabilistic classifier used in machine learning methods to solve the problems of sorting and classification [36,37]. The task of the Bayes classifier is to assign a new case to one of the decision classes, while the set of decision classes must be finite and defined a priori.

Decision trees, Perceptron, Random Forest, The Bayesian Classifier, SVM, and K-Nearest Neighbour are commonly used techniques for image classification. Decision Tree, SVM, and K-NN are used for satellite image classification [38]. Decision trees are used for image processing and image mining (that is, the mining of large datasets of different image types) [39]. Random Forest is used for image analysis (that is, landscape analysis and the analysis of satellite images) [40], and processing, which is a tool in medicine to diagnose disease [41] by classifying images according to the object categories they contain, in the case of a large number of object categories [42]. The Bayes Classifier is used in medicine [43]. The K-NN technique is used for image recognition, in areas such as facial recognition [44]. Perceptron, K-NN, and the Bayes Classifier are also used in weather forecasting [45]. The Bayes Classifier and K-NN are also used in fraud detection [46]. In the context of improving the management and effectiveness of employee learning processes in manufacturing companies, we did not find any sources that presented the use of these techniques for generating workstation instructions based on pre-recorded video material.

In order to address the above issues, our work proposes a new methodology for the generation of workplace procedures, in which CNN-SVM architecture is also applied. To the best of our knowledge, there is no existing work available on enabling CNNs and SVMs to generate workplace procedures. One important positive aspect of CNNs are their "learning features", i.e., omitting the manual functions that are necessary for other types of networks. Feature extraction is an important element in the success of a recognition system [47]. CNN features are automatically taught. One of the strengths of CNN is that they can be invariant, in the case of transformations like translation, scaling, and rotation. Invariance, rotation, and scale are three of the most important assets of CNN, especially in problems with image recognition, such as object detection, because they allow the abstraction of identity, so the network can effectively recognise the object in cases where the real pixel values in the image can vary significantly. In the case of problems with facial recognition, CNN has changed the facial recognition field, thanks to its learning characteristics and the transformation of its features. For example, Google FaceNet and Facebook DeepFace are based on CNNs. The CNN is arguably the most popular deep learning architecture, with CNNs now being the go-to model for every image related problem; in terms of accuracy, they have no peer. CNNs have also been successfully applied to recommender systems, natural language processing, and more. CNNs have also been used in the realm of networking and, more specifically, in the traffic classification of mobile applications [48].

The main advantage of the CNN, compared to its predecessors, is that it automatically detects [49] important features without any human supervision. For example, given many pictures of cats and dogs, it learns distinctive features for each class by itself. A CNN is also computationally efficient. It uses special convolution and pooling operations and performs parameter sharing. This enables CNN models to run on any device, making them universally attractive. CNN is a very powerful and efficient model, which performs automatic feature extraction in order to achieve superhuman accuracy to the point where, now, CNN models classify images better than humans do. Another main feature of CNNs is their weight sharing. In terms of performance, CNNs outperform NNs in conventional image recognition and many other tasks. For a completely new task or problem, CNNs are very good feature extractors. This means that useful attributes can be extracted from an already trained CNN, with its trained weights, by feeding data to each level and slightly tuning the CNN to the specific task, such as by adding a classifier after the last layer with labels specific to the task. This is also called pre-training, and CNNs are very efficient in such tasks compared to NNs. Another advantage of this pre-training is that we avoid training in CNNs and, thus, save memory and time. Convolutional Neural Networks take advantage of local spatial coherence in the input (often images), which allow them to have fewer weights, as some parameters are shared. This process, taking the form of convolutions, makes CNNs especially well-suited to extracting relevant information at low computational costs [50].

For classification processes, most "deep learning" models employ the Softmax activation function for predictions and to minimise cross-entropy loss [51,52]. Replacing Softmax with linear SVMs gives significant gains for the popular deep learning datasets, MNIST, CIFAR-10, and the 2013 ICML representation Learning Workshop's facial expression recognition challenge. SVM is a widely used alternative to Softmax for classification processes. First, a deep convolutional net is trained using supervised/unsupervised objectives to learn good, invariant, hidden, latent representations. The variables of the data samples are then treated as input and fed into linear SVMs. This technique can improve performance in the classification process. CNN-SVM was used during the development of a water leakage detection system [53] and was also used to identify the spatiotemporal motion characteristics of a human [54]. A comparison with state-of-the-art alternatives devised for image analysis applied to manufacturing (in the context of generating workstation instructions) is also done (Table 1), illustrating the novel contributions of our approach.

Table 1. A comparison with state-of-the-art alternatives.

Paper	Method	Applied to Manufacturing	Applied to Generating Workstation Instructions
[19]	SVD	speaker verification systems	No
[24,25]	PCA	diagnosing and predicting machine	No
[28,29]	LDA	facial recognition	No
[38]	Decision Tree	satellite image classification	No
[38]	SVM	satellite image classification	No
[40]	Random Forest	landscape analysis	No
[44,45]	K-NN	facial recognition weather forecasting	No
[45]	Perceptron	weather forecasting	No
[45]	The Bayes Classifier	weather forecasting	No
[48]	CNN	facial recognition recommender systems natural language processing traffic classification of mobile application	No
[53,54] This paper	CNN+SVM	water leakage detection system	No
		spatiotemporal motion characteristics of the human	No
		generating workstation instructions, based on pre-recorded video material	Yes

In this paper, we limit the provisions of our experiments according to the structure of the CNN, to extract the frames' features and the CNN-SVM for object classification. However, the principles behind the transformation of the underlying data and the overall EDLT framework are valid for other types of deep learning methods. As already pointed out, the proposed model allows knowledge of assembly tasks to be acquired from specialists and transferred to new people in the form of workplace procedures.

3. Model for Generating Workplace Procedures, in Real-Time, Using CNN-SVM Architecture

The proposed research model for generating workplace instructions using the convolutional neural network and SVM is presented below (Figure 1).

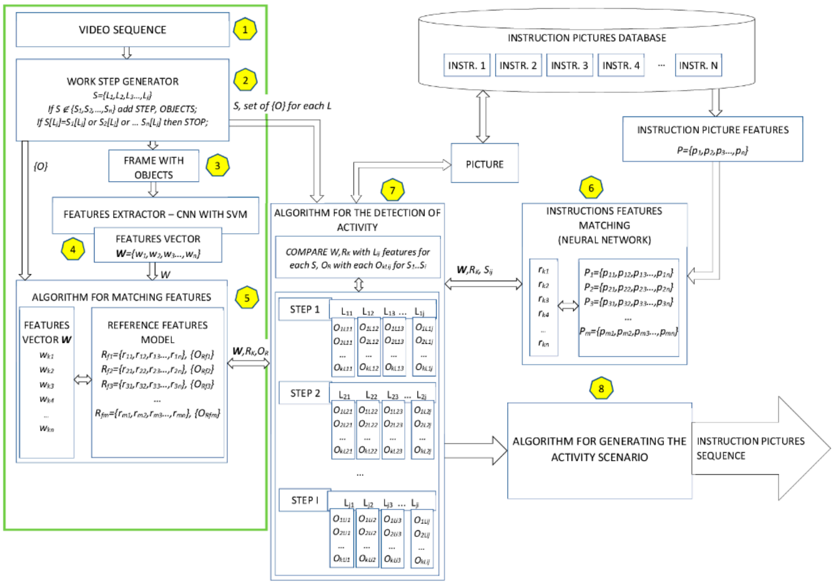

Figure 1. Model for the real-time generation of workplace procedures.

where:

W—frame features vector, k, n ∈ N
R_m—reference features vector, f, m ∈ N
S_n—step (S) number (m) and work (L) number (j), n, j ∈ N
O—set of {O} for each L, L ∈ N
P—instruction features vector, n, m ∈ N.

At first, the employee's specialist knowledge is obtained through video recording (Stage 1, Figure 1). Next (Stage 2, Figure 1), the conditions that allow the extraction of individual stages from the actions recorded on video are defined. It has been assumed that the rules are created according to:

(1) If $S \notin \{S_1, S_2, \ldots, S_n\}$ add STEP, OBJECTS;
(2) If $S[L_j] = S_1[L_j]$ or $S_2[L_j]$ or … $S_n[L_j]$ then STOP; where:

S_n—is a step number
L_j—is a work number
n, j ∈ N.

Next, the frame for the video sequence was subjected to the input of the neural network. For each of the frames, a feature vector and the set of objects thereupon were generated.

In Stage 4 (Figure 1), the architecture of CNN-SVM was employed as Resnet18 architecture, where: For experiments, the classification layer was changed from Softmax to SVM. The CNN network expresses the mapping between the objects on images and their classes. Conventionally, the Softmax function is the classifier used in the last layer of that kind of network. However, there have been studies in which a linear support vector machine (SVM) was used in an artificial neural network architecture. Studies proved that the CNN-SVM model was able to achieve a test accuracy of ≈99.04% using the MNIST dataset [51].

- conv1
- conv2_x
- conv3_x
- conv4_x
- conv5_x
- average pool
- fully connected
- SVM

The total time complexity of all convolutional layers (1) is [55]:

$$O\left(\sum_{l=1}^{d} n_{l-1} \cdot s_l^2 \cdot n_l \cdot m_l^2\right) \tag{1}$$

where l is the index of a convolutional layer and d is the depth (number of convolutional layers). n_l is the number of filters (also known as "width") in the l-th layer. n_{l-1} is also known as the number of input channels of the l-th layer. s_l is the spatial size (length) of the filter. m_l is the spatial size of the output feature map. The computational complexity of the SVM algorithm is between O(n^2) a O(n^3) [56]. More specifically, the complexity is O(max(n,d) min (n,d)ˆ2), where n is the number of points, and d is the number of dimensions [57].

In Stage 5, an algorithm for matching features (Table 2) was developed. The use of this algorithm enables the features of the individual frames of the analysed and defined video sequence to be determined according to the characteristics of the reference frames stored in the system.

Table 2. An algorithm for matching features—pseudocode.

An Algorithm for Matching Features

```
initialisation
for i: = 1 to m do
begin
    matching[i]←0;
    for j: = 1 to n do
    begin
        if compare($w_j$,$r_{ij}$) = true then inc(matching[i]);
    end;
end;
find max(matching[])
z = index of max(matching[]);
send→(W,$R_z$)
```

where:

i,j—loop counters
m—number of reference features vector
n—number of reference features vector elements
matching[]—a matrix of compatibility of vectors elements
w_j—frame vector element
r_{ij}—reference features vector element
z—index of matching[] matrix.

The computational complexity of this algorithm is O(nm) + O(n).

In Stage 6 (Figure 1), the database contains examples of animated illustrations matching the task for which the real-time workplace procedure generation was built. Next to be extracted (using CNN for each image feature) were:

$$P_m = \{p_1, p_2, p_3 \ldots , p_n\},$$

where:

P_m—instruction features vector
p_n—is a single feature of a vector frame
r_{kn}—is a feature of the reference frame features vector R, where $R_{k1} = \{r_{k1}, r_{k2}, r_{k3} \ldots , r_{kn}\}$
$n, m \in N$

In Stage 7 (Figure 1), an algorithm for the detection of activity (Table 3) was developed. The task of this algorithm is to analyse the characteristics of individual frames in the video sequence along with the reference features of these sequences and compare them with the characteristics of each step of the the service activity performed. In addition, information about the objects registered in the reference model and objects of a particular service step is compared.

Table 3. An algorithm of activity detection—pseudocode.

An Algorithm of Activity Detection

```
initialisation
for i: = 1 to x do
begin
    Step←0;
    Work←0;
    for j: = 1 to y do
    begin
        if (compare(($w_j$,$r_{ij}$),$S_i[L_{ij}]$) and compare($O_R$,$O_{Lij}$))=true then
        begin
            Step = i;
            Work = j;
        end;
    end;
end;
send→(W,$R_k$,$S_{step\ work}$)
```

where:

i,j—loop counters
x—number of steps
y—number of works
Step—index of step
Work—index of work
w_j—frame vector element
r_{ij}—reference of features vector element
$S_i[L_{ij}]$—the element of the features vector for *i* step
O_R—a set of objects for analysed frame
O_{Lij}—a set of objects for work

The computational complexity of this algorithm is O (nm).

An algorithm for generating the activity scenario (Stage 8, Figure 1) enables analysis of the model video material to be carried out which, in turn, allows the video frames responsible for the various stages of service activities to be identified. The algorithm that creates the activity scenario (Table 4) will generate a set of graphical instructions on the output depending on the activity currently performed by the employee.

Table 4. An algorithm for generating the activity scenario: pseudocode.

An Algorithm for Generating the Activity Scenario

```
Initialisation
S []←Steps
$L_S$ []←Work for each step
Initialise SCENARIO[]
for i:= 1 to count(S) do
begin
    for j:= 1 to count($L_{Si}$) do
    begin
        for k:= 1 to count(INSTRUCTION_FEATURES) do
        begin
            if compare($L_{Sij}$,$P_k$) = true then
            begin
                inc_size(SCENARIO[]);
                z = find_picture_in_database($I_k$);
                add(SCENARIO[],z);
            end;
        end
    end;
end;
```

where:

i,j,k—loop counters
S[]—a matrix of steps
L_s[]—a matrix of works for each step
scenario[]—a matrix instruction picture
z—index of the picture
L_{Sij}—element of work feature vector for each step
P_k—element of instruction feature vector

The computational complexity of this algorithm is O (nmk).

As the result of the use of the proposed algorithm, we can obtain a sequence of instruction pictures for a workplace procedure. Our whole proposed approach enables acquired specialist knowledge to be transformed into a useful form for new employees in the company. The use of our model eliminates the need for an experienced employee to participate in the process of educating new employees and shows increased effectiveness for this process in the company.

In our research model (Figure 1), the area marked in green includes elements of the model used during the experiment, while individual elements, marked with numbers, are explained in the description of the experiments and placed under the picture (based on the example of the repair of a solid fuel boiler).

4. Research Experiments

The aim of this experiment was to determine features and information about the occurrence of objects on each frame of video material and compare them with a test recording. In the next stage, the features of individual video frames were analysed for similarities. This whole process will allow future comparison of the video sequence obtained from the camera that recorded the employee's actions within a reference sequence. This, in turn, will be the basis of the algorithm for generating the activity scenario.

The experiment was videoed and presents the implementation of a single service operation, namely the procedure for repairing a solid fuel boiler.

In the first stage (Figure 1), the following sets of learning data were prepared (with the network being trained based on these sets):

- Reference material in the form of a video sequence, from which single frames were extracted as reference objects for the frames of the test material.
- Test video material, lasting about 13 s, also processed to form single frames (30 frames per second) from the 407 images. In addition to the collection, modified single frames were added that did not contain some of the objects, viz., they had no key and no bucket. Also inserted were several sets of frames, which were the final stages of the service activity performed (that is, a set of frames depicting the activity performed from the back). This collection totaled 597 frames. Assuming 30 frames per second, the video material obtained in this way takes about 20 s.
- The training set of the 3000 images from the network containing objects consisted of a hand, a flat key, a geared motor, an organizer, a burner auger, and a bucket.

For research purposes, the reference frames came from the same video material as the test frames. However, the reference frames were removed from the test set.

In Stage 2 (Figure 1), the conditions that allow the extraction of individual stages in the action recorded on the video are defined:

1. Motoreductor + hand—START CONDITION
2. Motoreductor + hand + bucket
3. Motoreductor + hand + bucket + key
4. Motoreductor + hand + bucket + organiser
5. Motoreductor + hand + bucket + organiser + burner auger
6. Motoreductor + hand + bucket + organizer—STOP CONDITION

Conditions were defined manually.

Next, the frame of the test video sequence was subjected to the input of the neural network. For each frame, a feature vector and the set of objects on it were generated.

In Stage 4 (Figure 1), an experiment was carried out in the MATLAB 2019a programme using the ResNet-18 network structure, in which the classification layer was changed from Softmax to SVM. SVM is designed for classification and regression tasks. For linearly separable data, the Support Vector Machine method allows a hyperplane to be established, thereby dividing the data set into two classes with a maximum separation margin. For linearly non-separable data a hyperplane that classifies objects with a minimum of error can be defined. When creating a hyperplane, keep in mind that the maximum margin of trust or margin of separation is the distance of the hyperplane from the nearest data [51,52].

In the next stage, the vectors of the referenced frame features, along with the set of objects for these frames and the vectors of the test frames (also with the set of objects on these frames) were compared using the algorithm for matching features (Stage 5, Figure 1).

Reference frames with objects representing individual activities are shown in Table 5.

Table 5. Objects on the reference frames.

Activity	Object of the Frame Tested	Reference Frame Name
START	Motoreductor + hand	RFrame0
Emptying the cleaner	Motoreductor + hand + bucket	RFrame1
Unscrewing the gearmotor	Motoreductor + hand + bucket + key	RFrame7
Unscrewing the gearmotor	Motoreductor + hand + bucket + organizer	RFrame147
Checking the burner auger	Motoreductor + hand + bucket + organizer + burner auger	RFrame408
Inserting the burner auger STOP	Motoreductor + hand + bucket + organizer	RFrame594

Results for the comparison of features are presented in Table 6.

Table 6. Results for the comparison of features.

Name of Reference Frame	Name of Tested Frame	Deviation
RFrame0	TFrame1	0.4909
RFrame1	TFrame2	0.0808
RFrame7	TFrame6	0.0323
RFrame147	TFrame148	0.0754
RFrame408	TFrame409	0
RFrame594	TFrame223	0

Results for the comparison of sets of objects for CNN -SVM are presented in Table 7.

Table 7. The result of the comparison of sets of objects for CNN with SVM.

Name of Reference Frame	Name of Tested Frame	Number of Objects Detected (Tested Frame/Reference Frame)
RFrame0	TFrame1	2/2
RFrame1	TFrame2	3/3
RFrame7	TFrame6	3/4
RFrame147	TFrame148	3/4
RFrame408	TFrame409	3/5
RFrame594	TFrame223	3/4

Results for the comparison of sets of objects for CNN-Softmax are presented in Table 8.

Table 8. Results for the comparison of sets of objects for CNN with Softmax.

Name of Reference Frame	Name of Tested Frame	Number of Objects Detected (Tested Frame/Reference Frame)
RFrame0	TFrame1	0/2
RFrame1	TFrame2	1/3
RFrame7	TFrame6	1/4
RFrame147	TFrame148	1/4
RFrame408	TFrame409	1/5
RFrame594	TFrame223	1/4

The accuracy test was carried out on a set consisting of 3000 images. The network was trained and then subjected to class object prediction tests from photos. For the CNN with an SVM classifier, the accuracy was 98.14%. For the CNN with Softmax, the accuracy was 97.83%

The results of the experiment indicate that the video material, which is the basis upon which the system is taught and tested at the same time, is able to meet the task of controlling the work performed based on similarities of the features of individual system frames. However, how exactly the system will behave for other video materials recorded at the workstation should be determined.

For the analysis of objects occurring in individual frames, the CNN-SVM detected three types of objects out of six possible objects. CNN-Softmax detected only 1 object. Objects that were detected on the image by CNN-SVM were the hand, the motoreductor, and the bucket. Objects that were not detected correctly by CNN-SVM were the key, the burner auger, and the organiser. For CNN-Softmax, the only detected object was the bucket.

In the case of the burner auger and the organiser, the system classified the objects detected as the motoreductors. In the case of the key, the system detected the hand. This situation may have arisen on account of the:

- The presence of another object or part thereof on the frame analysed;
- The small size of the object appearing on the frame (this being the key);

- Too few training samples (500 for each class);
- The low resolution of the analysed image; in the CNN network used, it was 227 × 227 pixels.

5. Discussion

In the literature one can find the adoption of the AI technologies, comprising ANNs in solutions that support managers in manufacturing companies, within the context of Industry 4.0 [58]. As presented in the experiments section, the use of CNN-SVM to develop a reference model of features for real-time workplace procedures, based on the acquisition of specialist knowledge in the form of a video recording, is more effective than using CNN-Softmax. This result is due to the number of objects detected. In the model proposed (Figure 1), this number is especially important because it is necessary to determine the correct order for the performance of actions in a given workplace procedure. The proposed model contributes to apparel classification research by presenting improved performance compared to other methods and techniques (introduced in Section 2) but also has strength as a new solution for the acquisition of specialist knowledge, as well as the accumulation of knowledge and sharing it in a useful manner. In this article, we worked with the data involved in the procedure for repairing a solid fuel boiler, which takes 20 s to complete. Before training the CNN-SVM classifier, we had to first define the conditions that allow the extraction of individual action stages in the recorded video. The frame of the test video sequence was then input, and for each of the frames, a feature vector and set of objects were generated on it. The reference values and the tested frame names were then compared. Finally, the effectiveness of the detection of objects using CNN-SVM was shown. The novelty of the prosed model is an innovative CNN-SVM architecture for the detection and classification, through video recording, of the frames' features, as obtained from an employee's specialist knowledge. CNN-SVM was used to reduce the classification error that is smaller than the classification error obtained by the CNN [59].

In further work, a model of the characteristics of the station instructions (illustration) will also be built with the help of CNNs. In order to obtain an automatic workplace instruction, we built the algorithm used to generate the activity scenario. However, we find this part of the analysis goes beyond the scope of the present study and discounts further experiments for future study, as this paper focuses on proposing and testing an algorithm for features matching and an algorithm for activity detection, rather than on the ways to illustrate an algorithm for generating activity scenarios.

6. Conclusions

The work presented in this paper describes a new approach to improving the effectiveness of the learning process of employees in manufacturing companies, using CNN-SVM. The specialist knowledge acquired and then recorded in the form of video material is formalised as a set of images and as a benchmark dataset in order to determine a reference model for features for real-time workplace procedures. Based on the results obtained by the experiments performed, it was possible to validate the pre-trained CNN with an SVM to extract features relative to the CNN with Softmax. Due to the detection of the type of object, we could determine the correct order for performing individual actions in a given procedure. In addition, the main advantage of the proposed approach (which allows it to perform workplace procedures) is that it connects the acquisition of specialist knowledge and the sharing of this knowledge without the engagement of an experienced employee. In future work, we will implement our approach in the form of an information system. New image databases from real manufacturing companies with larger amounts of data will be used, thereby allowing this system to be validated and used within a company to support employees' learning processes.

Author Contributions: Conceptualization, J.P.-M.; Data curation, D.H.; Formal analysis, J.P.-M. and D.H.; Funding acquisition, J.P.-M.; Methodology, J.P.-M.; Resources, D.H.; Software, D.H.; Validation, J.P.-M. and D.H.; Visualization, J.P.-M. and D.H.; Writing—original draft, J.P.-M. and D.H.; Writing—review & editing, J.P.-M.

Funding: This research received no external funding.

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

References

1. Roldán, J.J.; Crespo, E.; Martín-Barrio, A.; Peña-Tapia, E.; Barrientos, A. A training system for Industry 4.0 operators in complex assemblies based on virtual reality and process mining. *Robot. Comput. Integr. Manuf.* **2019**, *59*, 305–316. [CrossRef]
2. Madhusudanan, N.; Chakrabarti, A. A questioning based method to automatically acquire expert assembly diagnostic knowledge. *Comput. Aided Des.* **2014**, *57*, 1–14.
3. Taigman, Y.; Yang, M.; Ranzato, M.A.; Wolf, L. Deepface: Closing the gap to human-level performance in face verification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1701–1708.
4. Mutch, J.; Lowe, D.G. Object class recognition and localization using sparse features with limited receptive fields. *Int. J. Comput. Vis.* **2008**, *80*, 45–57. [CrossRef]
5. Serre, T.; Oliva, A.; Poggio, T. A feedforward architecture accounts for rapid categorization. *Proc. Natl. Acad. Sci. USA* **2007**, *104*, 6424–6429. [CrossRef] [PubMed]
6. Jarrett, K.; Kavukcuoglu, K.; Ranzato, M.; LeCun, Y. What is the best multi-stage architecture for object recognition? In Proceedings of the IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; pp. 2146–2153.
7. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *2012*, 1097–1105. [CrossRef]
8. Roldán, J.J.; Peña-Tapia, E.; Garzón-Ramos, D.; de León, J.; Garzón, M.; del Cerro, J.; Barrientos, A. Multi-robot systems, virtual reality and ros: Developing a new generation of operator interfaces. *Robot Oper. Syst.* **2019**, *3*, 29–64.
9. Mavrikios, D.; Karabatsou, V.; Fragos, D.; Chryssolouris, G. A prototype virtual reality-based demonstrator for immersive and interactive simulation of welding processes. *Int. J. Comput. Integr. Manuf.* **2006**, *19*, 294–300. [CrossRef]
10. Sun, Y.; Li, L.; Zheng, L.; Hu, J.; Li, W.; Jiang, Y.; Yan, C. Image Classification base on PCA of Multi-view Deep Representation. *J. Vis. Commun. Image Represent.* **2019**, *62*, 253–258. [CrossRef]
11. Materka, A.; Strzelecki, M. Texture analysis methods—A review. *Tech. Univ. Łodz Inst. Electron.* **1998**, *25*, 9–11.
12. Flook, A. In Encyclopedia of Food Sciences and Nutrition (Second Edition). 2003. Available online: https://www.sciencedirect.com/topics/medicine-and-dentistry/image-analysis (accessed on 10 August 2019).
13. Tran, N.M.; Burdejová, P.; Ospienko, M.; Härdle, W.K. Principal component analysis in an asymmetric norm. *J. Multivar. Anal.* **2019**, *171*, 1–21. [CrossRef]
14. Machine Learning—Singular Value Decomposition (SVD) & Principal Component Analysis (PCA). Available online: https://medium.com/@jonathan_hui/machine-learning-singular-value-decomposition-svd-principal-component-analysis-pca-1d45e885e491 (accessed on 1 August 2019).
15. Principal Components Analysis. Available online: https://stats.idre.ucla.edu/sas/output/principal-components-analysis/ (accessed on 2 August 2019).
16. Ansari, I.A.; Pant, M.; Ahn, C.W. Robust and false positive free watermarking in IWT domain using SVD and ABC. *Eng. Appl. Artif. Intell.* **2016**, *49*, 114–125. [CrossRef]
17. Available online: https://web.stanford.edu/class/cme335/lecture6.pdf (accessed on 1 August 2019).
18. Deng, P.; Wang, H.; Li, T.; Horng, S.J.; Zhu, X. Linear discriminant analysis guided by unsupervised ensemble learning. *Inf. Sci.* **2019**, *480*, 211–221. [CrossRef]
19. Sahidullah, M.; Kinnunen, T. Local spectral variability features for speaker verification. *Digit. Signal Process.* **2016**, *50*, 1–11. [CrossRef]
20. Sadek, R.A. SVD based image processing applications: State of the art, contributions and research challenges. *arXiv* **2012**, arXiv:1211.7102.
21. Mathews, B. Image Compression using Singular Value Decomposition (SVD). *Univ. Utah* **2014**. [CrossRef]
22. Alter, O.; Brown, P.O.; Botstein, D. Singular value decomposition for genome-wide expression data processing and modeling. *Proc. Natl. Acad. Sci. USA* **2000**, *97*, 10101–10106. [CrossRef] [PubMed]

23. Alter, O.; Golub, G.H. Integrative analysis of genome-scale data by using pseudoinverse projection predicts novel correlation between DNA replication and RNA transcription. *Proc. Natl. Acad. Sci. USA* **2004**, *101*, 16577–16582. [CrossRef]
24. Jasiński, M.; Radkowski, S. Application of main components in machine diagnostics. *Diagnostics* **2004**, *30*, 207–210.
25. Cempel, C.; Tabaszewski, M. Scaling of observations in multidimensional diagnostics of non-stationary machines. *Diagnostics* **2005**, *34*, 23–30.
26. Ma, J.; Yuan, Y. Dimension Reduction of Image Deep Feature using PCA. *J. Vis. Commun. Image Represent.* **2019**, *63*, 102578. [CrossRef]
27. Mudrova, M.; Procházka, A. Principal component analysis in image processing. In Proceedings of the MATLAB Technical Computing Conference, Prague, Czech Republic, 15 November 2005.
28. Ye, F.; Shi, Z.; Shi, Z. A comparative study of PCA, LDA and Kernel LDA for image classification. In Proceedings of the 2009 International Symposium on Ubiquitous Virtual Reality, Gwangju, Korea, 8–11 July 2009; pp. 51–54.
29. Prakash, N.S.; Shetty, P.; Kedilaya, K.; Nithesh Sunil, B.N. Comparative Study of PCA, KPCA, KFA and LDA Algorithms for Face Recognition. *Int. Res. J. Eng. Technol.* **2018**, *5*, 3460–3464.
30. Building Decision Trees. Available online: https://www.cs.cmu.edu/~{}bhiksha/courses/10-601/decisiontrees/ (accessed on 2 August 2019).
31. Learning Decision Trees. Available online: https://artint.info/html/ArtInt_177.html (accessed on 1 August 2019).
32. López, V.; Fernández, A.; Moreno-Torres, J.G.; Herrera, F. Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification. Open problems on intrinsic data characteristics. *Expert Syst. Appl.* **2012**, *39*, 6585–6608. [CrossRef]
33. Zhao, C.; Gao, Y.; He, J.; Lian, J. Recognition of driving postures by multiwavelet transform and multilayer perceptron classifier. *Eng. Appl. Artif. Intell.* **2012**, *25*, 1677–1686. [CrossRef]
34. Osowski, S. *Algorithmic Approach to Neural Networks*; WNT: Warszawa, Poland, 1996.
35. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
36. Nguyen, T.T.T.; Nguyen, T.T.; Sharma, R.; Liew, A.W.C. A lossless online Bayesian classifier. *Inf. Sci.* **2019**, *489*, 1–17. [CrossRef]
37. Geng, Z.; Meng, Q.; Bai, J.; Chen, J.; Han, Y.; Wei, Q.; Ouyang, Z. A model-free Bayesian classifier. *Inf. Sci.* **2019**, *482*, 171–188. [CrossRef]
38. Available online: https://www.researchgate.net/publication/326316293 (accessed on 2 August 2019).
39. Lu, K.C.; Yang, D.L. Image Processing and Image Mining using Decision Trees. *J. Inf. Sci. Eng.* **2009**, *25*, 989–1003.
40. Lowe, B.; Kulkarni, A. Multispectral image analysis using random forest. *Int. J. Soft Comput.* **2015**, *6*, 1–14. [CrossRef]
41. Tsymbal, A.; Kelm, M.; Costa, M.J.; Zhou, S.K.; Comaniciu, D.; Zheng, Y.; Schwing, A.G. Image Processing Using Random Forest Classifiers . U.S. Patent No. 8,744,172, 2014.
42. Bosch, A.; Zisserman, A.; Munoz, X. Image classification using random forests and ferns. In Proceedings of the 2007 IEEE 11th International Conference on Computer Vision, Rio de Janeiro, Brazil, 14–21 October 2007; pp. 1–8.
43. Solà-Soler, J.; Fiz, J.A.; Morera, J.; Jané, R. Multiclass classification of subjects with sleep apnoea-hypopnoea syndrome through snoring analysis. *Med Eng. Phys.* **2012**, *34*, 1213–1220. [CrossRef]
44. Chen, S.B.; Xu, Y.L.; Ding, C.H.; Luo, B. A nonnegative locally linear KNN model for image recognition. *Pattern Recognit.* **2018**, *83*, 78–90. [CrossRef]
45. Barde, N.C.; Patole, M. Classification and Forecasting of Weather using ANN, k-NN and Naïve Bayes Algorithms. *Int. J. Sci. Res.* **2016**, *5*, 1740–1742.
46. Kiran, S.; Kumar, N.; Guru, J.; Katariya, D.; Kumar, R.; Sharma, M. Credit card fraud detection using Naïve Bayes model based and KNN classifier. *Int. J. Adv. Res. Ideas Innov. Technol.* **2018**, *4*, 44–47.
47. Niu, X.-X.; Suen, C.Y. A Novel Hybrid CNN-SVM Classifier for Recognizing Handwritten Digits. *Pattern Recognit.* **2012**, *45*, 1318–1325. [CrossRef]

48. Aceto, G.; Ciuonzo, D.; Montieri, A.; Pescapé, A. Mobile encrypted traffic classification using deep learning: Experimental evaluation, lessons learned, and challenges. *IEEE Trans. Netw. Serv. Manag.* **2019**, *16*, 445–458. [CrossRef]
49. Arthur, F.; Hossein, K.R. Deep learning in medical image analysis: A third eye for doctors. *J. Stomatol. Oral Maxillofac. Surg.* **2019**, *120*, 279–288.
50. What Are the Advantages of a Convolutional Neural Network (CNN) Compared to a Simple Neural Network from the Theoretical and Practical Perspective? Available online: https://www.quora.com/What-are-the-advantages-of-a-convolutional-neural-network-CNN-compared-to-a-simple-neural-network-from-the-theoretical-and-practical-perspective (accessed on 10 August 2019).
51. Agarap, A.F. An architecture combining convolutional neural network (CNN) and support vector machine (SVM) for image classification. *arXiv* **2017**, arXiv:1712.03541.
52. Tang, Y. Deep learning using linear support vector machines. *arXiv* **2013**, arXiv:1306.0239.
53. Kang, J.; Park, Y.-J.; Lee, J.; Wang, S.-H.; Eom, D.-S. Novel leakage detection by ensemble CNN-SVM and graph-based localization in water distribution systems. *IEEE Trans. Ind. Electron.* **2018**, *65*, 4279–4289. [CrossRef]
54. Khan, M.H.; Farid, M.S.; Grzegorzek, M. Spatiotemporal features of human motion for gait recognition. *Signal Image Video Process.* **2019**, *13*, 369–377. [CrossRef]
55. He, K.; Sun, J. Convolutional neural networks at constrained time cost. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5353–5360.
56. Blachnik, M.; Wieczorek, T. Review of incremental learning methods. *Studia Inform.* **2015**, *36*, 47–60.
57. Chapelle, O. Training a support vector machine in the primal. *Neural Comput.* **2007**, *19*, 1155–1178. [CrossRef] [PubMed]
58. Patalas-Maliszewska, J.; Kłos, S. An Approach to Supporting the Selection of Maintenance Experts in the Context of Industry 4.0. *Appl. Sci.* **2019**, *9*, 1848. [CrossRef]
59. Elleuch, M.; Maalej, R.; Kherallah, M. A new design based-SVM of the CNN classifier architecture with dropout for offline Arabic handwritten recognition. *Procedia Comput. Sci.* **2016**, *80*, 1712–1723. [CrossRef]