Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2022) 30: 1621 – 1636

doi:10.55730/1300-0632.3869

Two person interaction recognition based on a dual-coded modified metacognitive (DCMMC) extreme learning machine

Saman NIKZAD, Afshin EBRAHIMI*
ICT Research lab, Electrical Engineering Faculty, Sahand University of Technology, Tabriz, Iran

Received: 01.04.2021 • **Accepted/Published Online:** 31.01.2022 • **Final Version:** 31.05.2022

Abstract: Human action recognition has been an active research area for over three decades. However, state-of-the-art proposed algorithms are still far from developing error-free and fully-generalized systems to perform accurate interaction recognition. This work proposes a new method for two-person interaction recognition from videos, based on well-known cognitive theories. The main idea is to perform classification based on a theory of cognition known as dual coding theory. The theory states that human brain processes and represents two types of information to learn/classify data named analogue and symbolic codes, i.e. (verbal as analogue and visual as symbolic). To implement such a theory in a two-person interaction classification system, we exploit dense trajectories as analogue codes and a bag of words as symbolic codes which are two code types hypothesized in the theory. In addition to dual coding theory, we propose to implement a metacognitive classifier model which adds a metalevel with its own rules to perform more accurate training process. We also propose a modification in a metacognitive component to prevent cognitive interference well known as the Stroop effect. Evaluations on both datasets revealed that the method offers comparable recognition accuracy (95.6% for the SBU interaction dataset and 91.1% for the UT-interaction dataset).

Key words: Human interaction recognition, metacognition, extreme learning machines, dual coding theory, dense trajectories

1. Introduction

Over the past few decades, considerable research efforts have been devoted to recognizing human actions and interactions from videos. The field has attracted increasing attention due to its enormous applications such as video surveillance [1], human action retrieval [2], human-machine interaction [3–5], healthcare systems [6, 7], sport analysis [8], etc. Action recognition is a challenging task because of its bottlenecks including large intraclass variations and interclass similarities [9]. As an instance of interclass similarity, one can see that the 'jogging' action is highly similar to the 'running' action. Classification of these two actions into two different classes is crucial in real-world scenarios. Intraclass variation means that the actors perform an action in different ways. The problem is more perceptible when we attempt to recognize interactions composed of various actions. To defeat intraclass variations, we have to train a system with different action performers. However, most interaction databases are developed using two persons, which cannot be enough to create a comprehensive dataset. In this study, we have tested our proposed algorithm on two interaction datasets introduced by state-of-the-art studies: the SBU Kinect interaction dataset [10] and the UT-interaction dataset [11]. The proposed algorithm is capable of recognizing different interaction categories by applying a dual-coded classifier based

*Correspondence: aebrahimi@sut.ac.ir

on modified metacognitive extreme learning machines (ELM) through hybrid feature codes hypothesized in dual coding theory (DCT). In addition to interaction datasets, we have experimental results on the well-known UCF-101 action dataset [12]. They validate that use of DCT together with a modified metacognitive classifier helps us evaluate the effect of each code type on recognizing human actions. As we are only concerned with two-person interaction recognition, we have restricted our one-person action dataset comparison with only two studies using modifications of ELM.

Figure 1 contains examples of dense trajectories referring to a boxing interaction between two individuals. As observed in the figure, the informative body parts of action performers are highlighted and automatically selected due to extracted motion information.

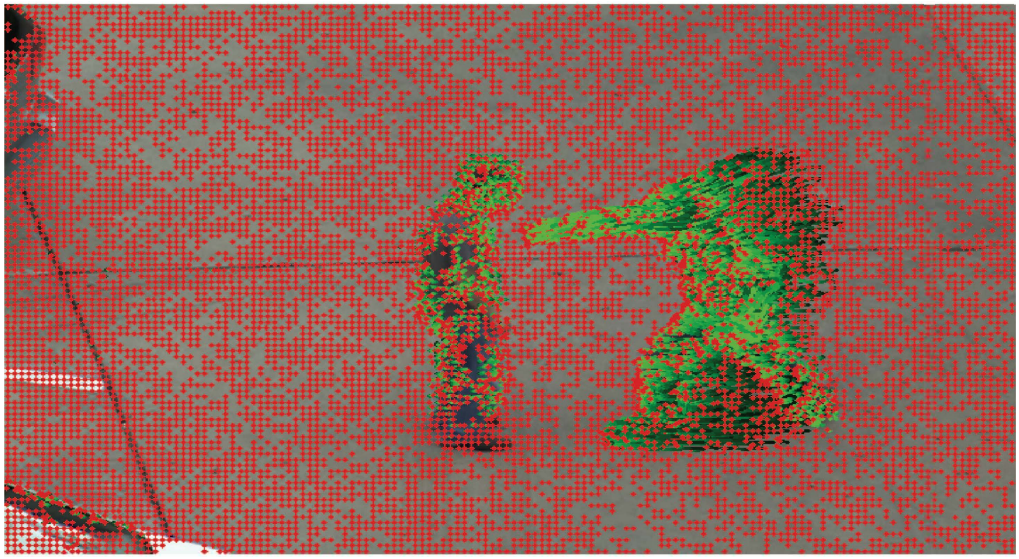

Figure 1. Extracted moving parts for three sample frames using dense trajectories.

2. Related work

As a subset of human action recognition, the task of human interaction recognition can be categorized into two classes based on image representation methods: holistic or global and local representation [13]. In local representation, the observed frame is processed as a collection of independent segments in a bottom-up manner. On the other side, global representation captures visual observation as a whole following a top-down mode. Global representation initiates the task of action recognition by locating actors, while local representation starts from pixel level to represent the action. Thus, global and local representations have major differences from the feature extraction perspective. In our case, we use a global representation, starting from actors instead of pixel level processing. As an exemplary local representation, [14] proposes an ELM-based method named double constrained bag of words (DC-BoW) which utilizes the spatial distribution information between features belonging to descriptor-level, presentation-level and hidden layer features. In another study, a deep residual 3D network is used to learn the features from both temporal and spatial sequences followed by an ELM as a classifier [15]. Extreme learning machine and support vector machines are utilized as two linear classifiers to analyze data and recognize patterns. Note that both methods use extreme learning machines as classifier kernel and we have compared our results with them in the experimental results section. An extreme learning machine classifier in [16] is trained to recognize human actions using ConvNet features. To do this,

they use temporal templates because of their ability to capture the entire motion sequence in a single image. In [17], multiple types of classifiers such as artificial neural networks (ANNs), support vector machine (SVM), multiple kernel learning (MKL), and metacognitive neural network (McNN) have been exploited to perform action recognition task. The authors proposed a local representation by using histogram of oriented gradient (HOG) from a segmented moving object. As another example, [18] used a combination of different interest point detectors and descriptors to form various STIP features. As observed in exemplary articles, local representation helps us get rid of background subtraction and object tracking steps which are common processes in action recognition. In addition, local representation summarizes an image or a video as descriptors which are robust against background clutter and occlusions. On the other hand, in the global representation methods, viewpoint problem complicates applying a specified method on different datasets. For example, in [19], the study aims at developing a skeleton-based representation which can be classified as a global representation method. The method encodes skeleton-based action instances into tensors and defines a set of operations to build different types of network cells. There are also a number of approaches which use representations directly motivated by the domain of human action recognition. As can be observed in the mentioned studies, most related works have focused on the analysis of activities induced by some individuals. However, some applications may depend on events induced by interactions between two or more individuals requiring interaction analysis. These are differences which make the employed algorithms application-specific.

3. Proposed methodology

As an effort toward development of the proposed dual-coded feature extraction, we have to capture the local motion information of a video as analogue codes. To this end, we follow the dense trajectory extraction scheme introduced by [18], which will include foreground motion and the surrounding context. Densely sampled feature points are tracked on different spatial scales. In addition to HOG and HOF descriptors, MBH is computed along dense trajectories as a motion descriptor. We aim at improving the action recognition performance using the BoW method as symbolic codes and a metacognitive ELM, which is an extension of traditional ELMs. These extensions include a modified metacognitive block as a complementary stage for training neural networks. In the second phase, we extend the principles of DCT to metacognitive extreme learning machines (McELM) and develop a dual coded classifier which employs both symbolic and analogue codes to learn videos considering the Stroop effect. The overall workflow of the system is shown in Figure 2. The paper identifies four main categories of contribution:

- A very concise clustered feature extraction model is proposed in the pipeline to extract foreground of the frames.
- We propose a new learning algorithm based on the well-known dual coding theory as a learning strategy. The use of metacognition theory to introduce a metalevel which performs training process in an extra supervisory manner is a novel idea between recently published studies about action/interaction recognition.
- We introduce a new score-based scheme to select between two defined codes in dual coding theory.
- We define a dual-coded metacognitive ELM as a modification of metacognitive network to train an interaction recognition system which helps us to design a discriminative classifier in comparison with state-of-the-art studies on the evaluated benchmark datasets.

In the remainder of this section, the proposed method is depicted in detail.

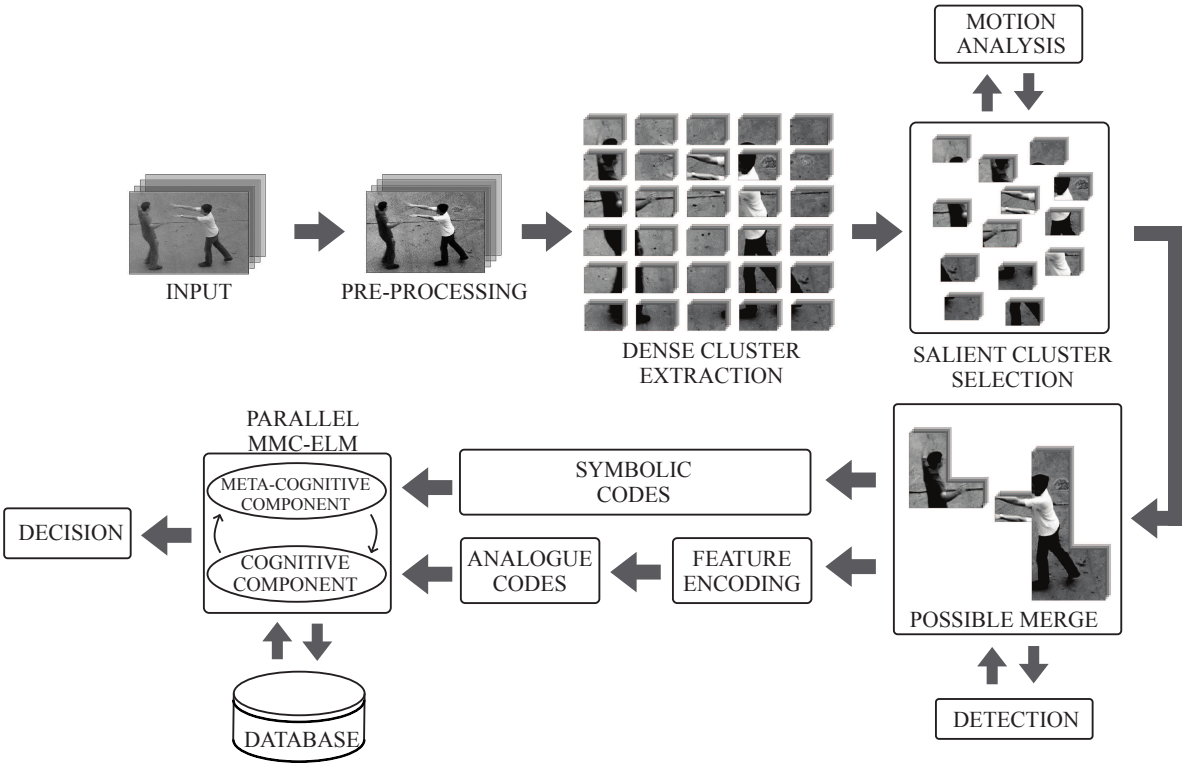

Figure 2. Overview of the proposed algorithm.

3.1. Salient cluster selection

Since the main contribution of this work is in feature extraction and analysis/recognition steps, we exploit an ordinary algorithm to segregate salient clusters from nonimportant ones. Further, as we work on moving pictures and not images, we select motion analysis as the basis of salient cluster selection. As a principal criterion of having motion in a video, optical flow is used in a combinatorial manner with a usual edge detector method (Canny) to extract moving edges of the cluster and calculate the average motion in the video. To this end, after applying the edge detector to the cluster, if the optical flow of the selected edge pixels is more than a predefined threshold, we count the pixels in average motion computation. Note that optical flow vectors of nonedge pixels are eliminated. After extraction of salient clusters, a simple merging step will unify clusters, which were topographically connected before clustering.

3.2. Dual-coded classifier

The process of learning new information in human brain has a well-known paradigm named Stroop effect. The problem is defined as an interference occurring when the processing of a specific stimulus feature impedes the simultaneous processing of another stimulus attribute. In other words, the interference between different types of information our brain receives causes a problem. The study in [20] looks into this area and investigates the use of a test named Stroop Color and Word Test to capture characteristics required to prevent this cognitive interference. Ridley Stroop discovered this strange phenomenon in the 1930s (see Figure 3). To avoid this phenomenon happening in our learning algorithm, we propose to give weights to the information during the training. The weights would be used in the test step to help the classifier decide based on the most important

information. The mentioned method is named dual coding theory, first hypothesized by Allan Paivio [21] as a theory of cognition to illuminate the great mnemonic effects of imagery. Paivio explains that in a learning process, there are two major ways through which a person can elaborate on materials. The first form emphasizes verbal associations and the second one creates a visual image to represent a word [22]. According to [21], an image helps the brain create a second form of memory code which is totally independent of verbal code. An image provides a second kind of memory code which is independent of a verbal code which makes images effective. The theory is called dual coding theory since rather than using only one memory code, we attempt to propose two types of memory codes, either of which can result in recall [22]. Inspired by DCT, we extend the metacognitive ELM principles to a dual coded classifier to take advantage of dual memory codes. According to Paivio, there are two manners through which one could expand on learned material: verbal associations and visual imagery [23]. Theoretically, the best learning accuracy is obtained if both verbal and visual components are involved in the learning process. This is practically impossible since we have no verbal data (e.g., subtitles or tags) in our databases. Thus, we extract visual words as symbolic codes to represent conceptual information of frames. A classifier learns each frame as a mental representation via symbolic codes and aggregates two separate classification channels into a unified form based on a scoring scheme. Most works on human action recognition have relied on either visual words or extracted features of the entire frame. In order to model the mentioned codes in the DCT theory, we need to define equal codes in image/video space. We propose two visual feature extraction schemes to simulate the codes in our application. Since the codes proposed in the DCT theory are explained as verbal and visual codes, we can assume the first one as detailed information, and the second as global information of the sample. Many convenient visual features can be found in the literature to extract detailed image information e.g., HOF, HOG, MBH, and etc. We use dense trajectory feature which captures all this information as a compact feature. There also exists many features for video analysis which give us general data of the scene. We employ bag of visual words which is a well-known feature extraction method to capture global image data. Thanks to the DCT approach, both information/feature types of the samples can be captured and used in both training plus test stages. In our model, we utilized a combination of two components using dense trajectories as analogue codes and visual words as symbolic codes. Although we are not the first to address the action recognition problem using the metacognitive learning framework, the design and use of a dual-coded classifier based on the metacognitive technique is a novel idea.

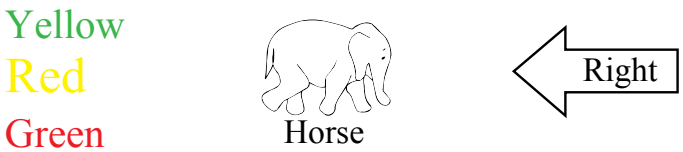

Figure 3. The well-known Stroop test which is a common diagnostic tool to determine attention problems.

3.2.1. Dense trajectories (analogue codes)

The aforementioned idea of dense trajectories [18] is exploited here to capture the mutual information between two individuals in each frame. Even from a cursory inspection, we can see that dense trajectories are insensitive to rapid moves, especially at shot boundaries. The method outperforms sparse tracking techniques such as KLT tracking [18]. This is mainly due to its ability to capture local motion information from the actors' body parts. The good coverage of foreground motion helps us concentrate on the foreground, while eliminating useless

information of the background. To extract dense trajectories, we have to perform a dense sampling on the scene containing actors and their background. The process is performed in different spatial scales on a grid space by W pixels. All sampled points have to be carefully chosen. As we are not concerned with points in homogeneous image segments, we restrict the tracking process to heterogeneous points. To monitor the quality of extracted features, points in homogeneous areas are ignored based on the Shi-Tomasi criterion [24]. The method uses feature dissimilarity as a measure which quantifies the change of appearance of a feature in consecutive frames. In such a method, choosing points with two small eigenvalues means a roughly constant intensity profile within a window which cannot be tracked reliably. By the way of contrast, two large eigenvalues may represent corners, salt and pepper textures, edges, or other patterns that are good features to track objects. We know that it is hard to track points in homogeneous segments of an image. Thus, we use a criterion with a threshold of T, which removes the points with small auto-correlation matrix eigenvalues. For each frame I, we can set the threshold based on the eigenvalues as:

$$T = 0.001 \times \max_{i \in I} \min(\lambda_i^1, \lambda_i^2), \tag{1}$$

where we define $(\lambda_i^1, \lambda_i^2)$ as the eigenvalues of point i for the frame I. We have also studied the impact of the eigenvalues on tracking accuracy. A larger value of eigenvalue threshold would degrade the performance, while a lower value adversely affects the processing speed. To find the best value of threshold parameters for tracking tasks, we set the definition domain of factors to be 0.001, 0.002, 0.003, 0.004, 0.005. As proven by [18], tracking quality increases using a threshold value near 0.001 (threshold values under this range did not show any improvement). Thus, a value of 0.001 as the threshold of eigenvalues is considered to be a balance between the saliency and density of sampled points.

Using fisher vector encoding, we can aggregate dense trajectories into a discriminative representation before being fed into a dual-coded modified metacognitive classifier. This procedure is a common practice as use of all data extracted from dense trajectory features is computationally expensive and requires large amounts of memory.

3.2.2. Bag of words (symbolic codes)

Based on the bag-of-words (BoW) model, our proposed model combines the benefit of an analogue code analysis from dense trajectories and symbolic codes as words. The well-known BoW model treats image features as an orderless collection of visual words (i.e. a document), which are iconic image fragments. The proposed classification scheme can be divided into analogue and symbolic classifiers, with the BOW algorithm in this paper belonging to the latter one. We convert each patch to a 128-dimensional vector to describe our visual word using SIFT which is known to be locally scale invariant. Using the k-means clustering algorithm, the extracted vectors are quantized to create a codebook. Thus, the initial codebook contains patches that may be equated with words in a hypothetical document. A video sample plays the role of a document in our version of the BoW model. While the speed and accuracy of our two-level classification system change systematically according to the amount of the codebook templates, we have studied the codebook size impact in our experimentation. Finally, the most similar codebook entry is assigned to each component during the evaluation process.

3.3. Metacognitive classifier

Here, we introduce the metacognitive extreme learning machine model used in our proposal, which is based on the ELM model as a combinatorial human interaction recognition mechanism. Partly inspired by [25], our

classification scheme has two main components: metacognitive and cognitive. The cognitive component contains an extreme learning machine while the metacognitive component contains the dynamic model of the mentioned cognitive component, selfregulated thresholds, and knowledge measures. Based upon the above analyses, the next section presents the metacognitive component, and later, we describe the combination of a modification on metacognition and ELMs.

3.3.1. Metacognitive component

We can define the lexical meaning of metacognition as cognition about cognition which exactly means having higher-order thinking skills [26]. A metacognitive process is defined as a metalevel with respect to an object-level cognitive process, where human beings develop new strategies to evaluate their memory information and improve their cognitive skills [27]. Briefly, in the metacognitive learning strategy, we set out to explore three main questions: what to learn, when to learn, and how to learn. Many convenient models for metacognition in human physiology can be found in the literature [28]. Based on [25], we propose to leverage recent advances in metacognitive learning techniques using the well-known Nelson and Narens model [29]. From Figure 4, it can clearly be observed that there are two relations between the object-level and the metalevel based on the information flow direction between the two levels. The metacognitive component is designed to control the learning process of the cognitive component using four predefined learning strategies derived from the basic principles of selfregulated human learning:

Sample delete strategy: Before being fed into the classifier, a new training sample is eliminated from the learning process if it carries any information similar to cognitive component knowledge. This occurs when the predicted category label is same as the actual one. The strategy helps avoid overtraining in the cognitive component. The criterion for sample delete strategy is given by:

$$\hat{c}^t == c^t \quad \text{AND} \quad \hat{p}(c^t|x^t) \geq \beta_d \tag{2}$$

where $\hat{c}^t$ is the predicted class label, $c^t \in (1, n)$ denotes the actual class label, and $\hat{p}(c^t|x^t)$ represents the classification confidence level. In other words, we can define it as predicted posterior probability:

$$\hat{p}(j|x^t) = \frac{\min(1, \max(-1, \hat{y}_j^t)) + 1}{2} \quad , \quad j = c^t \tag{3}$$

$\hat{y}_j^t$ is defined as the predicted output of the jth output neuron. Choosing β_d close to 1 may result in overtraining since in such a condition, all samples will participate in the learning process. We have examined a set of values from 0.9 to 0.95 (as recommended by [25]) to find an optimal metacognitive deletion threshold value β_d. This rule gives a –1 score to the code, on which the strategy is tested if it satisfies the deletion criterion.

Neuron growth strategy: Rather than designing a predefined network architecture, the metacognitive component attempts to add new neurons to the cognitive component using new training samples by considering the sample overlapping problem. In contrast to the sample deletion strategy, we use this strategy when the predicted class label is different from the actual class label in the new training sample, and we conclude that it contains significant information. Thus, we can upgrade our network by adding a new hidden neuron using the criterion:

$$(\hat{c}^t \neq c^t \quad \text{OR} \quad E^t \geq \beta_a) \quad \text{AND} \quad \psi_c(x^t) \leq \beta_c \tag{4}$$

$$E^t = \max_{j \in n} |e_j^t| \quad , \quad e^t = [e_1^t, ..., e_n^t]^T \tag{5}$$

where E^t represents the maximum hinge error and $\psi_c(x^t)$ is defined as class-wise significance. Here, β_c is a threshold to measure metacognitive knowledge which can be chosen within [0.3–0.7]. Further, the initial value for β_a (the threshold for selfadaptive metacognitive addition) is selected within [1.3–1.7], which will be adopted using the hinge error. For more details, one can refer to [25]. This rule gives a +1 score to the code on which the strategy is tested.

Parameter update strategy: We update cognitive component parameters with the new training samples based on:

$$c^t == \hat{c}^t \quad \text{AND} \quad E^t \geq \beta_u \tag{6}$$

where we define β_u as the selfadaptive metacognitive parameter update threshold. Thus, if the given criterion is satisfied, we use the current training sample in the weight update process of the cognitive component. However, it is slightly unreasonable to set a too high or too low threshold based on the explanations in [25]. Thus, we set the definition range for the parameter update threshold to be [0.4–0.7], which will be adopted based on the hinge error. The difference between parameter update and neuron growth strategies is that in neuron growth strategy, we improve our network by adding neurons to it. However, in parameter update strategy, we improve our network by updating weights. In the first strategy, we know that our data has enough information which persuades us to add a new neuron to the network. In the second strategy, the sample may not be rich enough to add a new neuron, but the hinge error is larger than a predefined threshold. In this situation, we use a new sample to upgrade the network weights with no neurons added. This rule gives a +1 score to the code on which the strategy was tested.

Sample reserve strategy: Training samples which do not contain significant information do not get involved in the learning process. These samples cannot satisfy deletion, neuron growth, and update criterions related to the cognitive component parameters. Thus, we may use them in subsequent phases to fine tune the parameters of the cognitive component. This rule gives a –1 score to the code on which the strategy was tested.

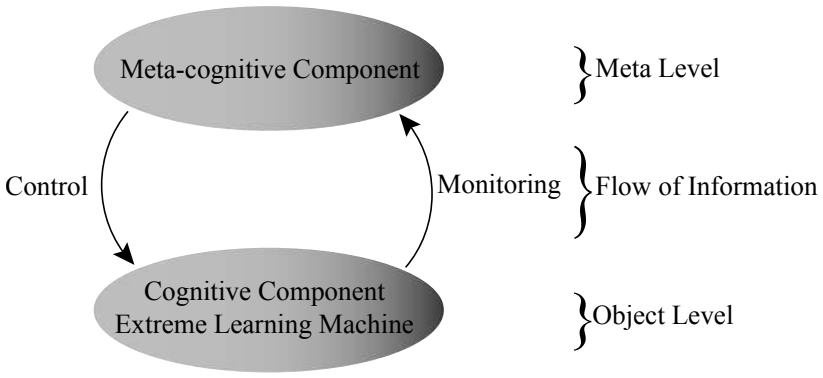

Figure 4. Theoretical mechanism of the proposed metacognitive classifier based on Nelson and Narens Model.

3.3.2. Cognitive component

There exist different types of neural networks based on the topology and learning algorithm. One of the most commonly used models which relies on feed-forward topology is single hidden layer feed-forward neural network

(SLFN) that has been applied to several problems in a wide range of areas including classification tasks [30], approximation [31], forecasting [32], etc. Different from initial models, ELM randomly chooses hidden nodes and analytically determines the output weights of SLFNs [33]. Briefly, random assignment of input weights and hidden layer biases is the idea that makes the learning process extremely fast.

Let us suppose there are N arbitrary distinct observations (x_i, t_i), where $\boldsymbol{x_i} = [x_{i1}, x_{i2}, ..., x_{in}]^T \in \mathbb{R}^n$ and $\boldsymbol{t_i} = [t_{i1}, t_{i2}, ..., t_{im}]^T \in \mathbb{R}^m$. A single-hidden layer feed-forward network with $\tilde{N}$ hidden nodes and activation function $g(x)$ can approximate the data with zero error based on following equation:

$$\sum_{i=1}^{\tilde{N}} \boldsymbol{\beta_i} \boldsymbol{g}(\boldsymbol{w_i}.x_j + \boldsymbol{b_i}) = o_j \quad \boldsymbol{j} = 1, ..., N \tag{7}$$

where $\boldsymbol{w_i} = [w_{i1}, w_{i2}, ..., w_{in}]^T$ is the weight vector which connects the ith hidden neuron and the input neurons, $\boldsymbol{\beta_i} = [\beta_{i1}, \beta_{i2}, ..., \beta_{im}]^T$ is the weight vector which connects the ith hidden neuron and the output neurons. b_i is defined as the threshold of ith hidden neuron and $w_i.x_j$ denotes the inner product of w_i and x_j. Theoretically the best classification property is obtained if the learning process reaches the minimal training error, and also the minimal norm of output weights:

$$\min ||\boldsymbol{H\beta} - T||^2 \quad \text{and} \quad ||\boldsymbol{\beta}|| \tag{8}$$

where H is the hidden-layer output matrix and T is the training label matrix for training samples. Hidden-layer output matrix can be defined as follows:

$$\boldsymbol{H}(\boldsymbol{w_1}, ..., \boldsymbol{w_{\tilde{N}}}, \boldsymbol{b_1}, ..., \boldsymbol{b_{\tilde{N}}}, \boldsymbol{x_1}, ..., \boldsymbol{x_N}) = \begin{bmatrix} \boldsymbol{g}(\boldsymbol{w_1}.\boldsymbol{x_1} + \boldsymbol{b_1}) & \dots & \boldsymbol{g}(\boldsymbol{w_{\tilde{N}}}.\boldsymbol{x_1} + \boldsymbol{b_{\tilde{N}}}) \\ \vdots & \dots & \vdots \\ \boldsymbol{g}(\boldsymbol{w_1}.\boldsymbol{x_N} + \boldsymbol{b_1}) & \dots & \boldsymbol{g}(\boldsymbol{w_{\tilde{N}}}.\boldsymbol{x_N} + \boldsymbol{b_{\tilde{N}}}) \end{bmatrix}_{\boldsymbol{N \times \tilde{N}}} \tag{9}$$

Since in most cases where $L \ll N$, we cannot compute $\boldsymbol{\beta}$ through direct matrix inversion. So, the smallest norm least squares solution is computed as:

$$\boldsymbol{\beta} = \boldsymbol{H}^{\dagger}\boldsymbol{T}, \tag{10}$$

where we define $\boldsymbol{H}^{\dagger}$ as Moore–Penrose generalized inverse for $\boldsymbol{H}$. The interested reader is referred to [34] for a detailed explanation.

3.3.3. Modified metacognition

We propose a modification on metacognition which is based on DCT. In this way, we can assay the power of each code (analogue or symbolic) in the training step and score each code based on the predefined metacognition rules. We believe that the proposed method helps us prevent Stroop effect happening during information recall process. In the test step, we can use this scoring scheme on deduction where we select the main feature (verbal or visual) to avoid cognitive inference. To prevent Stroop effect in the training phase, we give +1 or –1 scores to qualify the degree of importance for dual codes in each decision-making step (each frame classification).

We calculate these scores as **n** and **m** for each code and define a coefficient for each one as C_A (for analogue codes) as well as C_S (for symbolic codes), which will help us in the decision phase. Concerning the test, the metacognitive component decides about the prediction score of each code type and introduces each code based on the gained score (coefficient) in the training phase (Figure 5 (top-left)). The coefficients for both codes are explained in Equation 11. As samples in the training stage are presented one-by-one for each code type, the metacognitive component records scores from the cognitive component and chooses a suitable code in the test phase. As illustrated in Figure 5 (bottom-left), we assume **n** as the count of each **+1** score for analogue codes and **m** as the count of each **+1** score for symbolic codes. Both **n** and **m** are obtained from four metacognitive strategies defined in the previous sections. Then, we can define analogue code coefficient C_A and symbolic code coefficient C_S based on the gained scores by the codes as follows:

$$C_A = \frac{n}{n+m} \quad , \quad C_S = \frac{m}{n+m} \tag{11}$$

Each coefficient will be multiplied by the occurrence of the class label related to its code to be used in final decision making step (Figure 5 (top-right)). For example, assume that we have five frames to classify. We also assume that $C_A = 0.7$ and $C_S = 0.3$, which are obtained from the training step. Figure 5 (mid-right) and Figure 5 (bottom-right) illustrate a simple example for the voting scheme. In Figure 5 (mid-right), we can see that, first frame is categorized as class 5 using analogue codes (which is shown in green). The mentioned frame is also classified as class 1 using symbolic codes (which is shown in red). We have two frames classified as class 5, two frames as class 2, and one of the frames classified as class 3 using analogue codes. On the other side, all of the frames are classified as class 1 using symbolic codes. These values are used in Figure 5 (bottom-right) to be multiplied with the importance degrees (C_A and C_S) of each code. Finally, we use a 'winner takes all' method to decide about the final class based on the last row of Figure 5 (bottom-right). The main purpose of using this scheme is to prepare both codes to be recalled in the recognition step using their acquired scores.

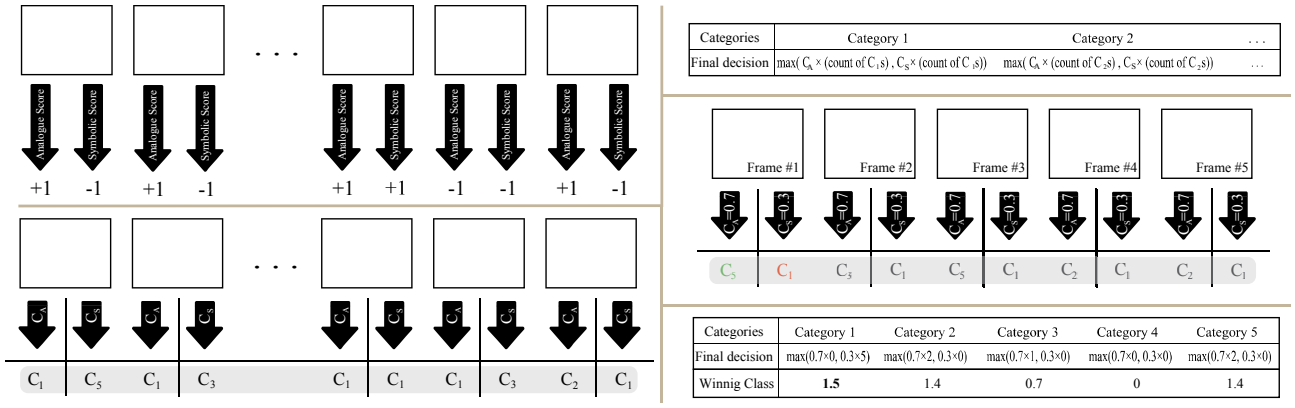

Figure 5. Proposed scoring function based on frame-level classification in modified metacognitive classifier. Two gray boxes indicate predicted class labels per frame.

4. Experimental results

We evaluated the recognition performance of the system using video sequences of SBU human interaction dataset (see Table 1) and the UT-interaction dataset (see Table 2). Our main goal is to demonstrate that

ELMs with metacognitive learning formulation can help us improve the generalizability of a classifier. Using the metacognitive framework, we can ensure that an algorithm prevents a network from learning repetitive information. To investigate the above-mentioned challenges on evaluation parameters, we have examined different values for the frame length and eigenvalue threshold of dense trajectories as well as three pairs of parameter settings for the metacognitive block of a classifier. Setting A: $(\beta_a = 1.3, \beta_c = 0.3, \beta_d = 0.9, \beta_u = 0.4)$, setting B: $(\beta_a = 1.5, \beta_c = 0.5, \beta_d = 0.92, \beta_u = 0.55)$ and setting C: $(\beta_a = 1.7, \beta_c = 0.7, \beta_d = 0.95, \beta_u = 0.7)$. Additionally, three different values of the dictionary size K are tested (500,1000,1500) and the effects of dictionary size are embedded in the result plots. We have also compared the proposed classifier performance with two studies using ELM with constrained bag of words [14] and ELM with 3D CNN [15] on UCF-101 using leave-one-out validation in Table 3.

Table 1. Correct recognition rates for different algorithms using SBU dataset (average rates are reported in %).

Method	Approach	Depart	Push	Kick	Punch	Exchange	Hug	Shake	Avg
GBSWC [35]	**1**	**1**	0.9	0.86	0.89	0.95	**1**	**0.91**	93.84%
RotClips + MTCNN [36]	-	-	-	-	-	-	-	-	94.17%
GCA-LSTM [37]	-	-	-	-	-	-	-	-	94.9%
Shape context [38]	0.92	0.92	**0.99**	**1**	**1**	0.91	**1**	0.87	95.05%
HGN + KNN [39]	-	-	-	-	-	-	-	-	90.8%
RV + HS + GCNs [40]	-	-	-	-	-	-	-	-	94.54%
Motion sequence [16]	0.95	0.9	0.91	0.92	0.83	**0.98**	0.95	0.83	90.98%
ELM (without DCMMC)	0.86	0.88	0.76	0.79	0.81	0.83	0.86	0.78	82.12%
Proposed	**1**	**1**	0.96	0.90	0.98	0.95	**1**	0.86	**95.6**%

Table 2. Correct recognition rates (in %) of different algorithms for UT-interaction dataset.

Method	Hug	Kick	Point	Punch	Push	Shake	Avg
DWT + Harris [41]	**1**	**1**	0.65	0.9	**1**	**1**	**91.5**%
Motion trajectory occurrences [42]	-	-	-	-	-	-	68.3%
LMDI [43]	0.75	0.75	**1**	0.87	0.87	**1**	87.5%
joint + AS [44]	-	-	-	-	-	-	90.3%
ELM (without DCMMC)	0.87	0.83	0.84	0.79	0.77	0.89	83.1%
Proposed	0.95	0.87	0.91	**0.92**	0.89	0.93	91.1%

Table 3. Correct recognition rates (in %) of different methods for UCF101 dataset.

Method	UCF101
DC-BoW (DC-ELM) [14]	88.9%
CNN-ELM [15]	92.7%
Proposed	93.9%

4.1. SBU interaction dataset

Figure 6 (left) displays the resulting recognition accuracy for different parameter sets, where the effect of dictionary size is plotted. As illustrated in the figure, the descriptor with the eigenvalue threshold value of

0.001 and dictionary size of 1000 presents the highest results (95.6%) using the parameter setting. We choose a 5-fold evaluation protocol to compare the results of our method with those of the state-of-the-art methods. From the experiments carried out, it is evident that the correct recognition rate of interactions is over 95% (except two categories), while three interactions are classified with an accuracy of 100%. By analyzing results for separate parameter sets, the DCMMC extreme learning machine classifier with the parameter set B shows surprisingly good outcomes for different eigenvalue thresholds. Additionally, to quantitatively evaluate the recognition and retrieval of interactions, we test the mentioned ELM without using the dual coded metacognition modification. This evaluation can illustrate the efficacy of the proposed method on the common dataset. Table 1 presents a detailed comparison with seven different algorithms implemented on SBU-interaction dataset.

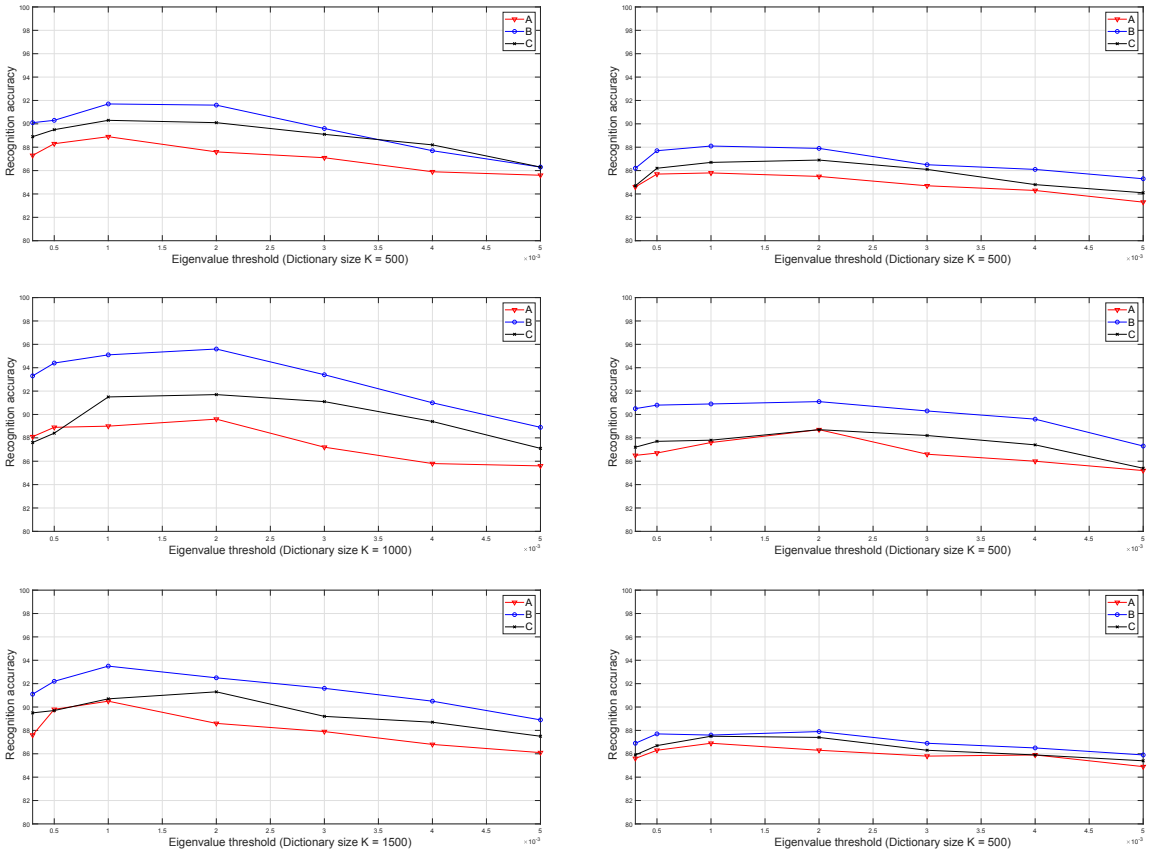

Figure 6. Correct recognition rates using proposed method with different parameters with 500 words (top row), 1000 words (middle row) and 1500 words in dictionary (bottom row) for SBU (left column) and UT (right column) datasets.

4.2. UT-interaction dataset

Figure 6 (right) depicts the recognition results for the UT-interaction dataset using the DCMMC extreme learning machine classification with a 5-fold setup which gives an average of 91.1%. At this point, we should note that this study aims at classifying two-person interactions, while there is a category in the UT dataset (pointing) performed by one person. We handle this by eliminating the mentioned action. Thus, the pointing class has not been taken into account in our evaluations. There is a detailed comparison in Table 2 for six different

algorithms implemented on UT-interaction dataset. Possible explanations for classification errors are similar to reasons explained in the former section. We have adopted four state-of-the-art methods for comparison on the UT-interaction dataset. The results obtained using our algorithm indicated a very competitive categorization accuracy (91.1%) on the UT dataset, which is among the best recognition rates acquired by the state-of-the-art algorithms on the UT-interaction dataset (see Table 2). We believe this decrement in classification performance is due to two main factors: moving background and camera jitter. UT-interaction dataset is made from two video sets. The first set has contained sequences with a slightly different zoom rate, static background, and a little camera jitter. The second set sequences have been taken on a lawn in a windy day with moving background and more camera jitter. Since we have selected train and test samples randomly from both sets, we expect lower recognition accuracy in comparison with SBU dataset. Similar to the SBU dataset, we have tested a strategy eliminating DCMMC on UT-interaction to assure effectiveness of the proposed method.

5. Conclusion and future work

As with the majority of studies, the algorithm proposed in this paper is subject to limitations which persuade us to continue the research and attempt to improve the method performance. The first is the lack of diverse, large, and all-inclusive interaction recognition datasets. Most publicly available datasets are designed in nonnatural situations to impact the reliability of existing research results. They are created in a laboratory environment in which the actors and their performing actions are limited in number, using constant camera angles with no changes in lighting conditions. There are multiple solutions to this problem, some of which may be addressed by the benchmarks creating datasets. One of them is to increase number of subjects in a dataset to evaluate the consistency of results across subjects. Since each subject acts differently from the other one, the diversity of subjects should be increased to create more comprehensive datasets. Recording videos with different angles is another solution that helps us evaluate perspective effect. In this way, we can also solve selfocclusion and partial occlusion issues on human body parts which is a fundamental concern in human interaction recognition. The inclusion of lightning variations with datasets is a simple but effective method to evaluate the proposed method with fully natural situations which is an important limitation in state-of-the-art studies. The second limitation is the variety of gestures which results in large intraclass variations and interclass similarities. This limitation exists due to constraints on research design and may be solved by using auxiliary information such as using multiple cameras or sensors. Another limitation to the generalization of the results is the variety of background scenes in natural videos which is often not addressed in current datasets. This is why most proposed methods may encounter difficulties with unstructured scenes or complex moving backgrounds. Some of these problems may be solved by using naturally created datasets with longer video duration. It means that the dataset must be longer in duration, containing cluttered complex background. The ground truth of the dataset should be extracted carefully to avoid training issues. Similar to any other study, the abovementioned limitations may force us to make unrealistic assumptions such as the limited number of subjects, constant lighting conditions, and partly clear background which would affect the accuracy of the proposed methodology in real situations. These assumptions may not influence the comparisons with other studies while they use the same dataset. In other words, the mentioned assumptions are dependent on the dataset, not the methodology. However, there are limitations such as process time and high computational complexity that result from the proposed methodology. Although we aimed at using a simple preprocessing and learning algorithm (ELM), we had a computation bottleneck at feature extraction/classification stages. Most of the processing time used for interaction learning and classification is consumed in the dual feature extraction step. Additionally, we have a

metacognition rule checker as one of the most computationally expensive steps in this study. These limitations will convince us that the feature extraction/classification stages should be done as offline processes. Thus, our future work involves studying alternative ways to extract different features with lower computational complexity while deciding based on more comprehensive metacognitive rules.

References

[1] Fadl S, Han Q, Li Q. CNN spatiotemporal features and fusion for surveillance video forgery detection. Signal Processing: Image Communication 2021; 90: 116066. doi: 10.1016/j.image.2020.116066

[2] Ramezani M, Yaghmaee F. Motion pattern based representation for improving human action retrieval. Multimedia Tools and Applications 2018; 77: 26009–26032. doi: 10.1007/s11042-018-5835-6

[3] Vogiatzidakis P, Koutsabasis P. 'Address and command': Two-handed mid-air interactions with multiple home devices. International Journal of Human-Computer Studies 2022; 159: 102755. doi: 10.1016/j.ijhcs.2021.102755

[4] Shen Z, Elibol A, Chong NY. Multi-modal feature fusion for better understanding of human personality traits in social human–robot interaction. Robotics and Autonomous Systems 2021; 146: 103874. doi: 10.1016/j.robot.2021.103874

[5] Shen Z, Elibol A, Chong NY. Multi-modal feature fusion for better understanding of human personality traits in social human–robot interaction. Robotics and Autonomous Systems 2021; 146: 103874. doi: 10.1016/j.robot.2021.103874

[6] Islam N, Faheem Y, Din IU, Talha M, Guizani M et al. A blockchain-based fog computing framework for activity recognition as an application to e-Healthcare services. Future Generation Computer Systems 2019; 100: 569-78. doi: 10.1016/j.future.2019.05.059

[7] Gao Y, Xiang X, Xiong N, Huang B, Lee HJ et al. Human action monitoring for healthcare based on deep learning. IEEE Access 2018; 6: 52277-85. doi: 10.1109/ACCESS.2018.2869790

[8] Chen J, Samuel RD, Poovendran P. LSTM with bio inspired algorithm for action recognition in sports videos. Image and Vision Computing 2021; 112 :104214. doi: 10.1016/j.imavis.2021.104214

[9] Akila K, Chitrakala S. Highly refined human action recognition model to handle intraclass variability and interclass similarity. Multimedia Tools and Applications 2019; 78: 20877–20894. doi: 10.1007/s11042-019-7392-z

[10] Yun K, Honorio J, Chattopadhyay D, Berg TL, Samaras D. Two-person interaction detection using body-pose features and multiple instance learning. In: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops; Providence, RI, USA; 2012. pp. 28-35. doi: 10.1109/CVPRW.2012.6239234

[11] Ryoo MS, Aggarwal JK. UT-interaction dataset, ICPR contest on semantic description of human activities (SDHA). In: IEEE International Conference on Pattern Recognition Workshops; 2010. Vol. 2, p. 4.

[12] Soomro K, Zamir A, Shah M. UCF101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv 2012: 1212.0402.

[13] Al-Faris M, Chiverton J, Ndzi D, Ahmed AI. A Review on Computer Vision-Based Methods for Human Action Recognition. Journal of Imaging 2020; 6 (6): 46. doi: 10.3390/jimaging606004

[14] Chao Wu, Yaqian Li, Yaru Zhang, Bin Liu. Double constrained bag of words for human action recognition. Signal Processing: Image Communication 2021; 98: 116399. doi: 10.1016/j.image.2021.116399

[15] Zou Y, Ren X. An Efficient Action Recognition Framework Based on ELM and 3D CNN. In: Chinese Intelligent Systems Conference; Springer, Singapore; 2020. pp. 641-648.

[16] Ijjina EP, Chalavadi KM. Human action recognition in RGB-D videos using motion sequence information and deep learning. Pattern Recognition 2017; 72: 504-516. doi: 10.1016/j.patcog.2017.07.013

[17] Patel CI, Labana D, Pandya S, Modi K, Ghayvat H et al. Histogram of oriented gradient-based fusion of features for human action recognition in action video sequences. Sensors 2020. 20 (24): 7299. doi: 10.3390/s20247299

[18] Wang H, Kläser A, Schmid C, Liu CL. Dense trajectories and motion boundary descriptors for action recognition. International journal of computer vision 2013; 103 (1): 60-79. doi: 10.1007/s11263-012-0594-8

[19] Haoyuan Z, Yonghong H, Pichao W, Zihui G, Wanqing L. SAR-NAS: Skeleton-based action recognition via neural architecture searching. Journal of Visual Communication and Image Representation 2020; 73: 102942. doi:10.1016/j.jvcir.2020.102942.

[20] Scarpina F, Tagini S. The stroop color and word test. Frontiers in psychology 2017; 8: 557.

[21] Clark JM, Paivio A. Dual coding theory and education. Educational psychology review 1991; 3 (3): 149-210.

[22] Reed SK. Cognition: Theories and applications. CA, USA: CENGAGE learning, 2012.

[23] Sternberg RJ. Cognitive theory. CA, USA: Thomson Wadsworth, 2003.

[24] Shi J. Good features to track. In: 1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition; Seattle, WA, USA; 1994. pp. 593-600. doi: 10.1109/CVPR.1994.323794

[25] Babu GS, Suresh S. Meta-cognitive RBF network and its projection based learning algorithm for classification problems. Applied Soft Computing 2013; 13 (1): 654-66. doi: 10.1016/j.asoc.2012.08.047

[26] Metcalfe J, Shimamura AP. Metacognition: Knowing about knowing. MIT press, 1994.

[27] Fleming SM, Frith CD. The cognitive neuroscience of metacognition. London, UK: Springer, 2014.

[28] Cox MT. Metacognition in computation: A selected research review. Artificial intelligence 2005; 169 (2): 104-141.

[29] Nelson TO. Metamemory: A theoretical framework and new findings. In: Psychology of Learning and Motivation. Academic Press, 1990, Vol. 26, pp. 125-173. doi: 10.1016/S0079-7421(08)60053-5

[30] Cheng S, Wu Y, Li Y, Yao F, Min F. TWD-SFNN: Three-way decisions with a single hidden layer feedforward neural network. Information Sciences 2021; 579: 15-32. doi :10.1016/j.ins.2021.07.091

[31] Guliyev NJ, Ismailov VE. On the approximation by single hidden layer feedforward neural networks with fixed weights. Neural Networks 2018; 98: 296-304. doi: 10.1016/j.neunet.2017.12.007

[32] Cheng X, Feng Z, Niu W. Forecasting Monthly Runoff Time Series by Single-Layer Feedforward Artificial Neural Network and Grey Wolf Optimizer. IEEE Access 2020; 8: 157346-157355. doi: 10.1109/ACCESS.2020.3019574

[33] Huang GB, Zhu QY, Siew CK. Extreme learning machine: theory and applications. Neurocomputing 2006; 70 (1-3): 489-501. doi: 10.1016/j.neucom.2005.12.126

[34] Huang GB, Zhu QY, Siew CK. Extreme learning machine: a new learning scheme of feedforward neural networks. In: 2004 IEEE International Joint Conference on Neural Networks; Budapest, Hungary; 2004. Vol. 2, pp. 985-990.

[35] Liu B, Ju Z, Liu H. A structured multi-feature representation for recognizing human action and interaction. Neurocomputing 2018; 318: 287-96. doi: 10.1016/j.neucom.2018.08.066

[36] Ke Q, Bennamoun M, An S, Sohel F, Boussaid F. Learning clip representations for skeleton-based 3d action recognition. IEEE Transactions on Image Processing 2018; 27 (6): 2842-55. doi: 10.1109/TIP.2018.2812099

[37] Liu J, Wang G, Duan LY, Abdiyeva K, Kot AC. Skeleton-based human action recognition with global context-aware attention LSTM networks. IEEE Transactions on Image Processing 2018; 27 (4): 1586-99.

[38] Nikzad S, Ebrahimnezhad H. Two-person interaction recognition from bilateral silhouette of key poses. Journal of Ambient Intelligence and Smart Environments 2017; 9 (4): 483-499. doi: 10.3233/AIS-170442

[39] Mottaghi A, Soryani M, Seifi H. Action recognition in freestyle wrestling using silhouette-skeleton features. Engineering Science and Technology 2020; 23 (4): 921-30. doi: 10.1016/j.jestch.2019.10.008

[40] Liu X, Li Y, Guo T, Xia R. Relative view based holistic-separate representations for two-person interaction recognition using multiple graph convolutional networks. Journal of Visual Communication and Image Representation 2020; 70: 102833. doi: 10.1016/j.jvcir.2020.102833

[41] Berlin SJ, John M. Particle swarm optimization with deep learning for human action recognition. Multimedia Tools and Applications 2020; 79: 17349-17371.

[42] Garzón G, Martínez F. A Fast Action Recognition Strategy Based on Motion Trajectory Occurrences. Pattern Recognition and Image Analysis 2019; 29 (3): 447-56. doi: 10.1134/S1054661819030039

[43] Sahoo SP, Ari S. On an algorithm for human action recognition. Expert Systems with Applications 2019; 115: 524-34. doi: 10.1016/j.eswa.2018.08.014

[44] Wang Z, Jin J, Liu T, Liu S, Zhang J et al. Understanding human activities in videos: A joint action and interaction learning approach. Neurocomputing 2018; 321: 216-26. doi: 10.1016/j.neucom.2018.09.031