Neural Computing and Applications (2021) 33:1207–1220
https://doi.org/10.1007/s00521-020-05009-z

ORIGINAL ARTICLE

Deep metric learning for open-set human action recognition in videos

Matheus Gutoski[1] · André Eugênio Lazzaretti[1] · Heitor Silvério Lopes[1]

Received: 2 December 2019 / Accepted: 2 May 2020 / Published online: 3 June 2020

Abstract
Human action recognition (HAR) is a topic widely studied in computer vision and pattern recognition. Despite the success of recent models for this issue, most of them approach HAR from the closed-set perspective. The closed-set recognition works under the assumption that all classes are known a priori and they appear during the training and test phase. Unlike most previous works, we approach HAR from the open-set perspective, that is, previously unknown classes are considered in the model. Additionally, feature extraction for HAR in the context of open set is still underexplored in the recent literature, since one needs to represent known classes with a low intra-class variance to reject unknown examples. To achieve this task, we propose a deep metric learning model named triplet inflated 3D convolutional neural network (TI3D), which builds upon the well-known I3D model. TI3D is a representation learning model that takes as input video sequences and outputs 256-dimensional representations. We perform extensive experiments and statistical comparisons on the UCF-101 dataset using a 30-fold cross-validation procedure in 25 different scenarios with varying degrees of openness and a varying number of training and test classes. Results reveal that the proposed TI3D achieves better performance than non-metric learning models in terms of F_1 score and Youdens index, indicating a promising approach for open-set video action recognition.

Keywords Human action recognition · Open-set recognition · Metric learning · Extreme value machine

1 Introduction

Human action recognition (HAR) is a recurrent subject in the fields of computer vision and pattern recognition. Recently, many works have achieved impressive performance on HAR from the *closed-set* perspective [6, 46, 59]. Closed-set classification works under the assumption that all classes are known *a priori*. However, in real-world scenarios, this assumption is often false, particularly in HAR, where entirely new classes can be created at will. That is, individuals can create new actions, movements, or gestures that are not previously known by the model. For this reason, HAR is naturally an *open-set* problem.

Open-set recognition has subtle differences with some other recognition tasks. For instance, classification with a reject option is a similar, yet different, task. In this case, the classifier still works under the closed-set assumption, and the purpose of rejecting is not to identify new classes. Instead, the goal is to avoid errors in low-confidence predictions [12]. Another similar task to open-set recognition is anomaly detection [14, 29]. Models such as the one-class support vector machine (OC-SVM) [35] and the support vector data descriptor (SVDD) [43] separate known data from their surroundings in all directions in the feature space. However, these models have poor performance in open-set recognition tasks [19], mainly because all known classes are usually treated as a single class in the feature space, ignoring the possible differences between them [12].

The ability to deal with the unknown is a widely studied subject in the open-set recognition field [12]. Unlike traditional classifiers, an open-set classifier should correctly classify samples that belong to known classes and reject those that belong to unknown classes, as shown in Fig. 1.

✉ Matheus Gutoski
matheusgutoski@alunos.utfpr.edu.br

André Eugênio Lazzaretti
lazzaretti@utfpr.edu.br

Heitor Silvério Lopes
hslopes@utfpr.edu.br

[1] CPGEI, Federal University of Technology – Paraná, Av. Sete de Setembro, 3165, Curitiba, PR 80230-901, Brazil

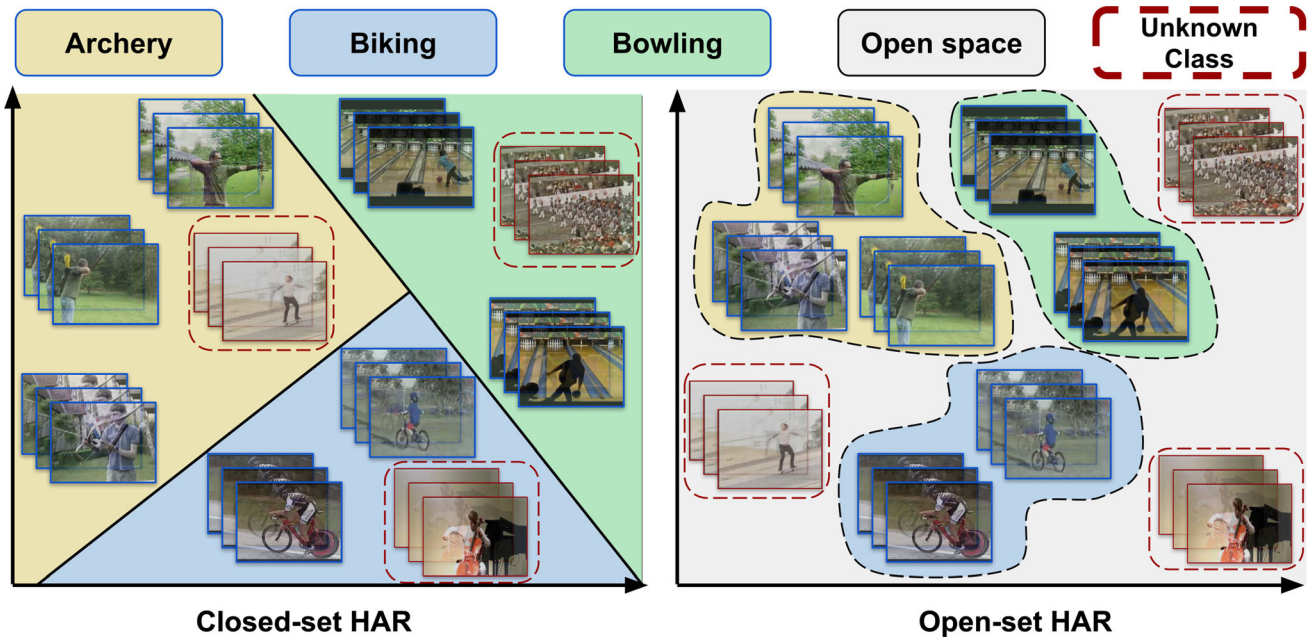

Fig. 1 Overview of the open-set human action recognition problem. In the closed-set scenario (left), new classes that appear during the test phase are wrongly classified as known. In the open-set scenario (right), new classes appear in what is called open space and are classified as unknown. Figure best viewed in color (color figure online)

To achieve open-set classification, the model needs to delimit the space occupied by known classes. The remaining feature space is defined as open space, where unknown classes may appear.

The open-set HAR problem is highly complex and still under-explored, with only a few works found in the recent literature [5, 30, 38, 61]. The challenges for performing open-set HAR are many:

- One must develop a feature extractor capable of generating robust representations of human actions using videos as input.
- It is necessary to have a classifier capable of rejecting inputs belonging to unknown classes.
- The evaluation protocol for open-set classification is often computationally expensive, in special for video-related tasks such as HAR. Moreover, evaluation protocols for open-set HAR are still ill-defined with no globally used protocol, which preclude direct comparison between methods in the literature.

For tackling the feature learning problem, we introduce the triplet inflated 3D convolutional neural network (TI3D), which is based on the I3D model [6]. TI3D performs metric learning (ML) to map input videos to a latent space where the cosine distance corresponds to a measure of semantic similarity between human actions.

In open-set problems, it is important to have compact and well-defined boundaries around known classes. In this case, compact boundaries that comprise small regions of open space are preferred, as opposed to large boundaries, which increase the risk of falsely accepting unknown samples. This concept is known as open-space risk [33]. Compact class boundaries are more easily achieved if intra-class distances are small and inter-class distances are large in the feature space, which is the core idea of ML. In this sense, TI3D indirectly contributes to minimizing the open space risk.

For solving the classification problem, we employ the extreme value machine (EVM) [31]. The EVM receives the representations generated by TI3D, classifies known samples, and rejects those of unknown classes. We show that the threshold parameter of the EVM dictates the classification performance and should be carefully chosen.

We also propose an evaluation protocol for open-set HAR. As suggested by early works on open-set recognition, our method is evaluated using a k-fold cross-validation procedure and varying degrees of openness [33], which is measured as a function of the number of classes in the training and test sets. Moreover, we ensure that similar degrees of openness appear through different numbers of training and test classes, showing that openness is not the only factor that affects the classification performance, and thus contributing to the evaluation protocol problem. We perform extensive experiments using the UCF-101 dataset [40] and show that deep metric learning with TI3D consistently improves open-set performance on HAR when compared to non-metric learning methods.

The main contributions of this paper are:

- The TI3D model, which performs deep metric learning to obtain high-quality feature representations for open-set HAR;
- A framework for performing open-set video HAR that combines TI3D and the extreme value machine;
- An evaluation protocol for open-set HAR;
- A performance analysis under different experimental settings and parameter values;
- A detailed statistical analysis of the results, which suggests that our feature learning model significantly improves the open-set HAR classification performance when compared to non-metric learning models.

This paper is organized as follows: Section 2 presents related works found in recent literature. Section 3 addresses the fundamental topics related to the method proposed for open-set HAR. Section 4 describes in detail the proposed method. Section 5 presents the experimental settings, their results, and a discussion. Finally, Sect. 6 reports the general conclusions drawn and suggests future research directions.

2 Related works

Unlike traditional image classification tasks, video classification depends upon complex spatiotemporal relationships among entities, making it a much more challenging task. Recently, convolutional neural networks have been used for solving several vision-related problems with great success [17, 21, 41]. However, traditional 2D CNNs are inherently fit for two-dimensional data and may be inefficient for solving video-related tasks [56].

A natural way to mitigate this shortcoming is to add recurrent layers on the top of the CNN, thus adding a "temporal-like" dimension to the classification model, as done in [8]. Although this approach seems to be more effective than a regular CNN, it raises problems. As pointed out by [6], recurrent models such as the long short-term memory (LSTM) network may fail at capturing fine low-level motion. Moreover, they demand a much higher computational power, require more data to train, and are more likely to overfit.

Another interesting approach to the video classification problem was proposed by [39]. The authors explicitly provide motion information to the CNN in the form of precomputed optical flow (OF). The authors also introduce the concept of two-stream convolutional networks, which has been widely used in succeeding works [51, 56, 63].

A different line of research has approached the video classification problem with a different idea. For instance, [45] employed 3D convolution filters to learn both spatial and temporal information from video sequences. This approach is commonly referred to as 3D convolutional network (or C3D, for short). These filters operate over both spatial and temporal dimensions, generating 3D feature maps. Actually, 3D convolutional networks have since become popular for video classification tasks [7, 16, 27, 46, 50, 52, 53, 59]. The main issue with this approach is that 3D CNNs have substantially more parameters than a regular CNN and, thus, require more data to be trained.

It was not until the work of [6] that a sufficiently large video dataset was introduced: Kinetics. With 400 classes and over 240K training videos, the dataset leveraged the performance of succeeding models. Another contribution of [6] was the inflated 3D convolutional neural network (I3D). The I3D model pre-trained on Kinetics has been shown to generalize well to other datasets and has inspired other research projects [46, 59].

Despite their impressive performance in video classification tasks, most of the current works were designed for a *closed-set* scenario [2, 10, 15, 22, 44, 47] and provide no means for rejecting data belonging to previously unseen classes. Some works approach this problem by introducing a reject mechanism in neural network models. Bendale and Boult [4] presents the OpenMax model that is capable of estimating the probability of an input belonging to an unknown class by using meta-recognition. Shu et al. [37] introduces the deep open classification (DOC) model for open set classification of text documents using a 1-vs-rest final layer of sigmoids.

Other works have approached the open-set problem with non-deep classifiers such as the 1-vs-set machine [33], W-SVM [32], nearest non-outlier (NNO) [4], POS-SVM [34] and, more recently, the extreme value machine (EVM) [31].

Unlike deep learning-based classifiers, non-deep classifiers do not learn representations automatically, and thus are sensitive to the quality of the features used as input. Ideally, features should be extracted such that similar inputs are mapped close from each other and dissimilar inputs are mapped far from each other. This kind of representation is often achieved through metric learning (ML) [20, 25].

ML consists of learning a distance function that measures the similarity between inputs. Classic ML algorithms, such as the large margin nearest neighbor (LMNN) [54], have inspired more recent deep ML models. Many of these models have been used for facial recognition [28], such as Deep Face [42], FaceNet [36], SphereFace [24], and CosFace [49]. Other works have also successfully applied deep ML for object classification [18, 58], one-shot learning [48], and video-related tasks [23, 57]. As a matter of fact, the relationship between ML and open-set recognition is still under-explored in the literature. Some very recent

works have accessed the impact of ML in open-set image and product classification [26, 60]. However, to the best of our knowledge, this is the first work to address open-set HAR in videos using deep ML.

Regarding open-set human action recognition, very few recent works can be found in the literature. Shu et al. [38] presents the Open Deep Network (ODN) model, which uses multi-class triplet thresholding to detect unknown classes. Yang et al. [61] employs micro-Doppler signatures for recognizing human activities. Busto et al. [5] investigates domain adaptation strategies, and [30] proposes a voting-based system for novelty detection in open-set action recognition. Despite tackling a similar problem, our work differs from the above-mentioned works in the sense that our model focuses on feature learning for open-set HAR, instead of novelty/unknown detection or classification mechanisms. The classification task is handled by the EVM in all cases to ensure a fair comparison between different feature learning models. In principle, TI3D could serve as the backbone network for other open-set classifiers as well, including some of the above-cited works.

3 Theoretical aspects

3.1 Inflated 3D convolutional neural networks

The inflated 3D convolutional neural network (I3D) was proposed by [6] for human action recognition in videos.

The model consists of two streams of the Inception V1 model with inflated kernels. The inflation consists of extending kernels learned on ImageNet from 2D to 3D to initialize the I3D model. As in previous models, one of the streams uses RGB images as input, while the other uses precomputed OF features. The optimization is performed using standard backpropagation with a softmax cross-entropy loss.

Because of its state-of-the-art performance and simplicity, we used the I3D model, pre-trained on ImageNet and Kinetics, as the backbone network for open set video classification. Since the goal of this work is not an in-depth exploration of the temporal aspect of the model, we discarded the Optical Flow stream and used only the RGB stream. For more details about the model, refer to [6].

3.2 Extreme value machine for open-set recognition

The Extreme Value Machine (EVM) was introduced by [31] as an open-set classifier. In the EVM, each class in the training set is represented by a set of extreme vectors, each of which associated with a Probability of Sample Inclusion Ψ.

The EVM uses the concept of margin distributions, which is the distribution of the half margin distances of the observed data. Let $\mathbf{x}_i$ be a training sample and y_i be its label, and given $\mathbf{x}_i$ and $\mathbf{x}_j$, where $\forall j, y_j \neq y_i$, and $\mathbf{x}_j$ is the nearest point to $\mathbf{x}_i$, the margin estimate for the pair $(\mathbf{x}_i, \mathbf{x}_j)$ is given by:

$$\mathbf{m}_{ij} = \frac{||\mathbf{x}_i - \mathbf{x}_j||}{2}. \tag{1}$$

Computing Eq. (1) for the τ nearest points allows for estimating a distribution of the margins. To estimate this distribution, the extreme value theorem (EVT) is used. The margin distribution for the minimum values of $\mathbf{x}_i$ is then given by a Weibull distribution [31]. The probability of inclusion Ψ for a point $\mathbf{x}'$ is given by

$$\Psi(\mathbf{x}_i, \mathbf{x}', \kappa_i, \lambda_i) = \exp\left(-\frac{||\mathbf{x}_i - \mathbf{x}'||}{\lambda_i}\right)^{\kappa_i}, \tag{2}$$

where $||\mathbf{x}_i - \mathbf{x}'||$ is the distance between $\mathbf{x}'$ and $\mathbf{x}_i$, λ_i and κ_i are the Weibull shape and scale parameters, respectively.

Ψ is an EVT rejection model where $\Psi(\mathbf{x}_i, \mathbf{x}', \kappa_i, \lambda_i)$ corresponds to the probability that a sample does not lie beyond the negative margin. Despite having zero probability around the margin, the model still supports a soft margin. The probability that a point $\mathbf{x}'$ belongs to class C_l is given by Eq. 3:

$$\hat{P}(C_l|\mathbf{x}') = \text{argmax}_{i:y_i=C_l}\, \Psi(\mathbf{x}_i, \mathbf{x}', \kappa_i, \lambda_i). \tag{3}$$

Then, the final classification function is:

$$y^* = \begin{cases} \text{argmax}_{i:y_i=C_l}\, \hat{P}(C_l|\mathbf{x}'), & \text{if } \hat{P}(C_l|\mathbf{x}') \geqslant \delta \\ \text{unknown}, & \text{otherwise} \end{cases}, \tag{4}$$

where δ is a threshold that defines the boundary between known and open space.

In order to reduce the size of the model, many redundant $[\mathbf{x}_i, \Psi(\mathbf{x}_i, \mathbf{x}', \kappa_i, \lambda_i)]$ pairs can be discarded with minimal impact on performance. Let $\mathbf{x}_i$ be a point and $\Psi(\mathbf{x}_i, \mathbf{x}', \kappa_i, \lambda_i)$ be its corresponding model. Let $\mathbf{x}_j$ be a point in the same class and $\Psi(\mathbf{x}_j, \mathbf{x}', \kappa_j, \lambda_j)$ be its corresponding model. Let ς be the probability threshold above which the pair $[\mathbf{x}_j, \Psi(\mathbf{x}_j, \mathbf{x}', \kappa_j, \lambda_j)]$ is considered redundant with respect to $[\mathbf{x}_i, \Psi(\mathbf{x}_i, \mathbf{x}', \kappa_i, \lambda_i)]$. If $\Psi_i(\mathbf{x}_i, \mathbf{x}_j, \kappa_i, \lambda_i) \geq \varsigma$, then $[\mathbf{x}_j, \Psi(\mathbf{x}_j, \mathbf{x}', \kappa_j, \lambda_j)]$ is redundant. Finally, let $I(\mathbf{x}_i)$ be an indicator function that keeps or discards a pair:

$$\begin{cases} I(\mathbf{x}_i) = 1, & \text{if } \langle \mathbf{x}_i, \Psi(\mathbf{x}_i, \mathbf{x}', \kappa_i, \lambda_i) \rangle \text{ kept} \\ I(\mathbf{x}_i) = 0, & \text{otherwise.} \end{cases} \tag{5}$$

If the pair $[\mathbf{x}_i, \Psi(\mathbf{x}_i, \mathbf{x}', \kappa_i, \lambda_i)]$ is kept, it becomes an extreme vector. The model reduction is, then, defined by the following integer linear programming objective function:

$$\begin{aligned} &\text{minimize} \sum_{i=1}^{N_l} I(\mathbf{x}_i), \\ &\text{s.t. } \forall j, \exists i \mid I(\mathbf{x}_i)\Psi(\mathbf{x}_i, \mathbf{x}', \kappa_i, \lambda_i) \geqslant \varsigma, \end{aligned} \tag{6}$$

where N_l is the number of training points of class C_l. For a more detailed explanation about the EVM, refer to the original paper [31].

4 Method

An overview of the proposed method for evaluating the performance of TI3D, as well as baseline approaches, is shown in Fig. 2. Initially, known and unknown classes are randomly selected from the UCF-101 dataset. Next, videos are preprocessed and passed through one of the base networks, which are used to precompute video representations and initialize the TI3D model. The TI3D model is then trained to maximize inter-class and minimize intra-class cosine distances between videos. Finally, new representations are extracted from TI3D and classified by the EVM. This process is repeated 30 times for each experiment with different random seeds. The following sections will detail each step of the processes.

4.1 Data sampling

In this work we used the UCF-101 [40] dataset. This dataset has been used in other human action recognition works in the literature [6, 46, 59]. It contains 101 classes of human actions and over 27 hours of video data.

The fold selection strategy consists of randomly selecting C_k known classes from the pool of 101 classes. Once the known classes are selected, 70% of the videos belonging to these classes are used for training and 30% for testing. The test set is further incremented with C_u unknown classes, which are also selected at random. This data sampling protocol generates subsets of data under different degrees of openness. We compute openness as suggested by [33]:

$$\text{openness} = 1 - \sqrt{\frac{2 \times \text{Tr}}{\text{Te} + \text{Ta}}}, \tag{7}$$

where Tr, Te, and Ta stand for the number of training, test, and target classes. Since all training classes are included during the test time, we set Tr = Ta. This is equivalent to the redefined openness equation suggested by [12].

The UCF101 dataset demands additional attention concerning how the known classes are split into training and test sets. Within each class of the dataset, there are groups of videos that have been sampled from the same source video, rendering them very similar to each other. Hence, we ensure that all videos belonging to a given group remain in the same split.

4.2 Data preprocessing

The data preprocessing steps follow the guidelines proposed by [6]. RGB frames are normalized in the range $[-1 \ldots 1]$ and resized so that the smallest side contains 256 pixels, preserving the aspect ratio.

4.3 Base network models

Since TI3D requires a base network to build upon, we consider two variants of the I3D model. The first model was pre-trained on ImageNet and Kinetics, as provided by the original authors [6]. The second model consists of the original I3D model with an additional fine-tuning step using data from the known classes of the UCF-101 dataset.

For fine-tuning the I3D, we removed the original classification block used in kinetics and replaced it with a new classification block with C_k output neurons, softmax activation, and cross-entropy loss.

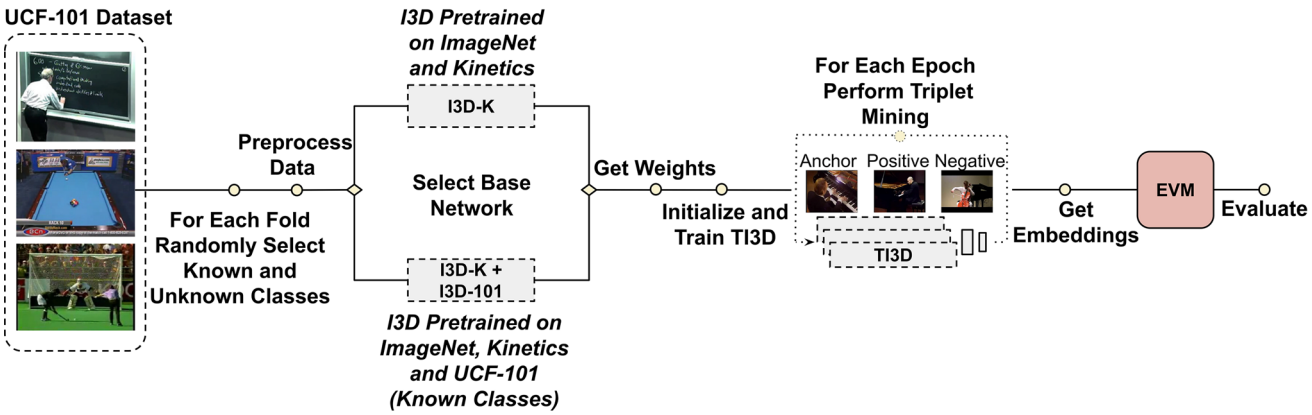

Fig. 2 Overview of the proposed method

The training steps follow those proposed in [6]. A package of 64 frames is used at a time. These frames are selected using a random temporal crop. Moreover, a random spatial crop selects a $224 \times 224 \times 64$ region of the cuboid. Last, the cuboid has a 50% chance to receive a horizontal flip (mirror). These steps are applied at each epoch of training for data augmentation purposes.

Once again the set of known classes is split into train and validation sets at a 70%/30% ratio for fine-tuning the I3D. We use the stochastic gradient descent (SGD) optimizer with a learning rate of 0.1, weight decay of 10^{-5}, and the Nesterov momentum of 0.9. The model was trained for 10 epochs or until the validation accuracy stagnated in 0.95 or higher. In most cases, the validation reached near-perfect accuracy since the reduced number of classes and closed-set scenario makes the problem trivial for I3D. The batch size was set to 6, due to hardware constraints.

4.4 Triplet inflated 3D convolutional neural network (TI3D)

Triplet networks were introduced by [18] and popularized in [36] for learning face embeddings. Its formulation was inspired by a classic metric learning approach called large margin nearest neighbors (LMNN) [54].

The main idea behind triplet networks is to map data to a space where distance corresponds to a measure of semantic similarity. The triplet network model takes three inputs: anchor, positive, and negative. For instance, in a human action recognition task, the Anchor (a) is a video of any given class, the positive (p) is a video of the same class, and the negative (n) is a video of a different class.

Given $N(a, p, n)$ triplets, the triplet loss function L is defined as shown in Eq. 8, where i is the triplet index, $f(\mathbf{x}^a)$, $f(\mathbf{x}^p)$, $f(\mathbf{x}^n)$ are the anchor, positive, and negative embeddings, α is a margin parameter, and $_+$ indicates the loss is ≥ 0.

$$L = \sum_{i=1}^{N} \left[\| f(\mathbf{x}_i^a) - f(\mathbf{x}_i^p) \|_2^2 - \| f(\mathbf{x}_i^a) - f(\mathbf{x}_i^n) \|_2^2 + \alpha \right]_+ . \tag{8}$$

In principle, optimizing the model using the triplet loss function produces discriminant features and well-separated classes by a margin of at least α in the Euclidean space. However, based on previous experiments, we find that the Euclidean distance is not the most appropriate for the task of video HAR. Therefore, we turn to the cosine distance.

The cosine distance Θ between two feature vectors $\mathbf{a}$ and $\mathbf{b}$ is given by:

$$\Theta(\mathbf{a}, \mathbf{b}) = 1 - \frac{\mathbf{a} \cdot \mathbf{b}}{\| \mathbf{a} \| \| \mathbf{b} \|} . \tag{9}$$

The updated cosine triplet loss function becomes:

$$L_\Theta = \sum_{i=1}^{N} \left[\Theta(f(\mathbf{x}_i^a), f(\mathbf{x}_i^p)) - \Theta(f(\mathbf{x}_i^a), f(\mathbf{x}_i^n)) + \alpha \right]_+ . \tag{10}$$

Our TI3D network is built by discarding the softmax layer of the pre-trained I3D and appending two additional fully connected layers to the end of the model, with 512 and 256 output neurons. The weights in new layers are initialized with the Glorot uniform method [13]. During the optimization, we freeze the weights of the I3D network and optimize only the newly added fully connected layers. This way, we can use precomputed features and greatly speed up the training process. The main drawback of this approach is that it requires a backbone network to build upon. However, it has been shown that starting from a pre-trained model can lead to much better results than training a model from scratch [6].

The TI3D is optimized using the triplet loss (Eq. 10) with the cosine distance (Eq. 9) as the distance function. We chose this function empirically, based on previous experiments that have shown a significant performance gain compared to the Euclidean distance. This difference in performance with respect to high-dimensional feature vectors was also observed in [31] and elaborated in [1].

4.5 Triplet mining

Once the base network was defined, we used its weights to initialize TI3D. An important aspect of our TI3D model is the formation of training triplets. Since using all possible combinations of triplets is both unfeasible and does not lead to good performance (see [36]), we turned to a triplet mining strategy. For each training epoch, semi-hard and hard triplets were mined. Semi-hard triplets are those in which the distance between the anchor and positive videos is smaller than the distance between the anchor and negative videos, and such distance is smaller than the desired margin, i.e.,

$$\begin{aligned} \Theta(f(\mathbf{x}^a), f(\mathbf{x}^p)) &< \Theta(f(\mathbf{x}^a), f(\mathbf{x}^n)) \\ &< \Theta(f(\mathbf{x}^a), f(\mathbf{x}^p)) + \alpha. \end{aligned} \tag{11}$$

Hard triplets are those in which the distance between the anchor and positive videos is larger than the distance between the anchor and negative videos, i.e.,

$$\Theta(f(\mathbf{x}^a), f(\mathbf{x}^p)) > \Theta(f(\mathbf{x}^a), f(\mathbf{x}^n)). \tag{12}$$

In our experiments, the margin parameter α was set to 0.2. The network was trained for 50 epochs or until no more semi-hard or hard triplets could be mined. In all experiments, we used SGD with learning rate of 0.001 and a batch size of 128.

4.6 Video feature extraction

During the feature extraction step, a package of 250 frames was fed to the trained network. These frames were not selected at random. Instead, a central temporal crop was employed. In cases where the video did not contain 250 frames, we loop the video to obtain the remaining frames. The spatial crop was also taken from the center of the image with a resolution of 224×224 pixels. This process resulted in a feature vector of 1024 dimensions (I3D) or 256 dimensions (TI3D) per video.

4.7 Classification with the extreme value machine

The EVM parameters were adjusted as follows. The tail size τ for fitting the Weibull distribution was set to 10, since the original authors state that there is well-established for selecting this parameter [31]. The cover threshold ς was set to 0.1, so as to simulate a more realistic scenario where smaller models are preferred because they require less computational resources. These parameters were kept fixed in all experiments, since performing parameter optimization would require a prohibitively large number of experiments. Moreover, a more in depth investigation regarding the parameters was presented in [31].

However, we investigated the impact of varying the probability of inclusion (δ). On one hand, [31] searched for values of δ in the range $[1 \times 10^{-1}, 1.5 \times 10^{-1}, \ldots, 3 \times 10^{-1}]$. On the other hand, we found that this range was not enough to obtain the best classification performance. Hence, our search for δ was expanded to smaller and larger probability values, in the range $[1 \times 10^{-8}, 1 \times 10^{-7}, \ldots, 1 \times 10^{-1}]$, and $[4 \times 10^{-1}, 5 \times 10^{-1}, \ldots, 9 \times 10^{-1}, 9.9 \times 10^{-1}]$, respectively.

4.8 Evaluation protocol

The evaluation protocol proposed in this work used the macro-averaged F_1-score, as suggested by previous works in the context of open-class classification [4, 31, 33]. We also compute the Youdens index [62] suggested by [34].

The macro-averaged F_1-score is the harmonic mean between macro-averaged precision and recall. In the open set recognition, only the known classes are used to compute the F_1-score. When a sample that belongs to a known class is predicted as unknown, it is considered a false negative.

The macro-averaged precision P and recall R are presented in Eqs. 13 and 14, respectively:

$$P = \frac{1}{C_k} \sum_{i=1}^{C_k} \frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FP}_i}, \tag{13}$$

$$R = \frac{1}{C_k} \sum_{i=1}^{C_k} \frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FN}_i}, \tag{14}$$

where C_k is the number of known classes during test time; TP, FP, and FN stand for true positives, false positives, and false negatives, respectively. Then, the macro-F_1-score is computed as shown in Eq. 15:

$$F_1 = 2 \times \frac{P \times R}{P + R}. \tag{15}$$

Many previous works consider the macro-F_1-score, as described above, to be an *open metric* [4, 31, 33]. However, [34] classifies it as a *closed metric*. According to the authors, *closed metrics* should measure the classifier's capability to discriminate known classes from one another, and *open metrics* should measure the capability to discriminate known from unknown. In our view, the macro-F_1-score measures the potential to distinguish between known classes, while also measuring the capability of rejection in a weak way. We consider it weak because it only accounts for incorrectly rejected data, but does not directly account for the data that belongs to unknown classes.

For measuring the capability to distinguish known from unknown more directly, we used the Youdens index. It combines the recall R and the specificity S, as defined in Eq. 16:

$$S = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}. \tag{16}$$

The Youdens index J is given by $J = R + S - 1$, and it is computed as a binary metric, where all samples from known classes are assigned the *known* label and unknown samples are assigned the *unknown* label. J is defined in the range $[-1 \ldots 1]$, where a score of -1 is achieved by a classifier that incorrectly classifies all samples, 0 by an uninformative classifier, and 1 by a perfect classifier.

We compared our method with two variants of the I3D model, namely I3D trained on ImageNet and Kinetics (I3D-K) and I3D-K fine-tuned using the known classes with standard softmax cross-entropy loss (I3D-K + I3D-101). The TI3D builds upon these models by performing an additional fine-tuning step using the cosine triplet loss (Eq. 10).

In current open-set recognition literature, some works use evaluation protocols with fixed training, validation and test sets (no cross-validation) [4, 34, 38, 60], while some works use 10-fold or less cross-validation procedures [3, 11]. In this work, we consider that cross-validation is quite important to obtain a robust performance

measure, since some combinations of classes may be "easier" than others, thus leading to a non-realistic performance evaluation. Actually, in our experiments, the number of possible known/unknown class combinations is quite large and, therefore, we report the average results of 30-fold to reduce the possible result bias generated by random class selection. Although a thorough evaluation using all possible combinations of classes would be desired, it is computationally unfeasible to perform such a large number of experiments with our computational resources. However, we expect that a 30-fold cross-validation provide a good approximation to the actual performance of our model.

5 Experiments and results

For clarity, this section is divided as follows: Sect. 5.1 reports the experimental parameters, Sect. 5.2 presents the results obtained in terms of the F_1-score and Youdens index. Section 5.3 provides an analysis of the results with respect to *openness* and the classification threshold δ.

5.1 Control parameters

For a thorough evaluation of the proposed method in different scenarios, experiments in multiple settings (regarding the number of training and test classes) were performed. A total of 25 combinations were done, as shown in Table 1. For instance, in the first experiment 2 random *known* classes and 2 random *unknown* classes (totaling 4 test classes, where 2 are known and 2 are unknown) were selected for training and testing, respectively. This sampling strategy allows the evaluation of the models under different degrees of openness.

Each experiment was evaluated under a set of δ values, as mentioned in Sect. 4.7. Moreover, all experiments reported in this section were performed in a 30-fold cross-validation procedure, except for the experiment with 30 train classes and 101 test classes, which was performed on a 10-fold trial for reasons of computational resources.

5.2 Results

The experimental results are shown in Table 2, and it includes the mean and standard deviation obtained in the k-fold cross-validation procedure. Only the results using the best value of δ are presented.

Several observations can be made with respect to the experimental results. First, it can be observed that on average, fine-tuning the I3D-K model on the known classes using a standard softmax cross-entropy loss (I3D-K + I3D-101) boosts the open-set performance in terms of F_1-score and Youdens index.

Table 1 Number of training classes, test classes, and openness of each experiment

# Train classes	# Test classes	Openness
2	4	0.18
	6	0.29
	8	0.36
	10	0.42
3	6	0.18
	9	0.29
	12	0.36
	15	0.42
4	8	0.18
	11	0.27
	15	0.35
	18	0.4
5	10	0.18
	13	0.25
	17	0.32
	20	0.37
6	12	0.18
	15	0.24
	18	0.29
	22	0.35
7	14	0.18
	18	0.25
	22	0.31
	25	0.34
30	101	0.32

Second, fine-tuning the models with TI3D yields better results in almost all cases, with a few exceptions where the models presented the same performance. None of the experiments have shown a decrease in performance when using TI3D. It can also be observed that TI3D, on average, obtained a lower standard deviation than the other methods, showing that it is more robust regarding the possible combinations of training and test classes. Finally, the combination of the three models (I3D-K + I3D-101 + TI3D), followed by I3D-K + TI3D, which are the main contributions of this work, presented the best overall performance.

Figure 3 shows the F_1-score in the form of boxplots for the experiments with 3 known classes (Tr) and 6, 9, 12, and 15 test classes (Te). The line inside the box represents the mean, the lower and upper limits of the box represent the first and third quartiles, and the lower and upper lines outside the box represent the minimum and maximum values. The boxplot shows that TI3D increased the mean

Table 2 Mean and standard deviations of the *k*-fold cross-validation procedure obtained with different number of classes used during the training and test phases

	F_1 score				Youdens index			
# Train/test classes	I3D-K	I3D-K + **TI3D**	I3D-K + I3D-101	I3D-K + I3D-101 + **TI3D**	I3D-K	I3D-K + **TI3D**	I3D-K + I3D-101	I3D-K + I3D-101 + **TI3D**
2/4	0.87 (0.13)	0.88 (0.11)	0.82 (0.15)	**0.91 (0.09)**	0.90 (0.10)	0.91 (0.09)	0.87 (0.11)	**0.93 (0.07)**
2/6	0.87 (0.11)	**0.88 (0.09)**	0.84 (0.16)	0.88 (0.10)	0.87 (0.12)	**0.88 (0.10)**	0.86 (0.12)	0.87 (0.10)
2/8	0.87 (0.11)	**0.89 (0.08)**	0.83 (0.12)	0.85 (0.09)	**0.88 (0.10)**	**0.88 (0.10)**	0.81 (0.14)	0.83 (0.11)
2/10	0.90 (0.09)	**0.91 (0.07)**	0.85 (0.08)	0.89 (0.07)	0.89 (0.10)	**0.89 (0.09)**	0.85 (0.10)	0.87 (0.09)
3/6	0.85 (0.10)	**0.89 (0.07)**	0.82 (0.14)	0.87 (0.10)	0.88 (0.08)	**0.92 (0.05)**	0.88 (0.09)	0.90 (0.08)
3/9	0.84 (0.09)	0.88 (0.06)	0.87 (0.08)	**0.89 (0.07)**	0.86 (0.08)	0.88 (0.07)	0.88 (0.08)	**0.89 (0.07)**
3/12	0.85 (0.09)	**0.89 (0.07)**	0.86 (0.09)	**0.89 (0.07)**	0.83 (0.10)	0.87 (0.08)	0.85 (0.11)	**0.87 (0.07)**
3/15	0.84 (0.11)	**0.89 (0.04)**	0.86 (0.08)	0.88 (0.07)	0.82 (0.09)	**0.86 (0.06)**	0.82 (0.10)	0.84 (0.08)
4/8	0.83 (0.15)	**0.89 (0.07)**	0.83 (0.11)	0.87 (0.08)	0.88 (0.07)	**0.91 (0.06)**	0.87 (0.08)	0.90 (0.07)
4/11	0.84 (0.13)	0.87 (0.07)	0.85 (0.10)	**0.89 (0.07)**	0.86 (0.08)	**0.89 (0.06)**	0.87 (0.08)	0.89 (0.07)
4/15	0.83 (0.09)	**0.87 (0.06)**	0.85 (0.06)	**0.87 (0.06)**	0.82 (0.08)	**0.86 (0.07)**	0.81 (0.07)	0.83 (0.08)
4/18	0.86 (0.08)	0.89 (0.07)	0.88 (0.07)	**0.89 (0.06)**	0.82 (0.10)	0.86 (0.09)	**0.86 (0.08)**	**0.86 (0.08)**
5/10	0.79 (0.14)	0.85 (0.13)	0.88 (0.07)	**0.89 (0.06)**	0.86 (0.09)	0.90 (0.08)	0.91 (0.06)	**0.92 (0.05)**
5/13	0.83 (0.07)	0.86 (0.06)	0.85 (0.09)	**0.87 (0.08)**	0.85 (0.07)	**0.88 (0.06)**	0.87 (0.08)	0.88 (0.07)
5/17	0.82 (0.09)	0.86 (0.06)	0.86 (0.06)	**0.88 (0.06)**	0.81 (0.10)	0.84 (0.07)	0.85 (0.07)	**0.87 (0.06)**
5/20	0.80 (0.07)	0.85 (0.04)	0.88 (0.05)	**0.88 (0.04)**	0.78 (0.08)	0.81 (0.07)	0.85 (0.06)	**0.86 (0.07)**
6/12	0.81 (0.10)	**0.88 (0.09)**	0.85 (0.08)	0.86 (0.07)	0.85 (0.08)	**0.91 (0.06)**	0.89 (0.06)	0.90 (0.05)
6/15	0.83 (0.07)	**0.87 (0.06)**	0.85 (0.07)	0.86 (0.07)	0.84 (0.07)	**0.88 (0.06)**	**0.88 (0.06)**	**0.88 (0.06)**
6/18	0.80 (0.10)	0.85 (0.06)	0.85 (0.06)	**0.87 (0.06)**	0.82 (0.07)	0.85 (0.07)	0.85 (0.07)	**0.86 (0.07)**
6/22	0.81 (0.08)	**0.87 (0.05)**	0.87 (0.06)	0.87 (0.06)	0.80 (0.07)	**0.85 (0.05)**	0.84 (0.07)	0.85 (0.07)
7/14	0.79 (0.08)	0.84 (0.06)	**0.88 (0.05)**	**0.88 (0.05)**	0.84 (0.06)	0.88 (0.05)	0.90 (0.05)	**0.91 (0.04)**
7/18	0.82 (0.06)	0.86 (0.06)	0.87 (0.06)	**0.87 (0.05)**	0.84 (0.06)	**0.88 (0.04)**	0.88 (0.06)	0.88 (0.05)
7/22	0.79 (0.07)	0.83 (0.07)	0.88 (0.05)	**0.88 (0.04)**	0.80 (0.07)	0.83 (0.06)	**0.88 (0.06)**	**0.88 (0.06)**
7/25	0.79 (0.07)	0.84 (0.07)	0.85 (0.06)	**0.86 (0.05)**	0.77 (0.09)	0.81 (0.09)	0.83 (0.07)	**0.83 (0.06)**
30/101	0.64 (0.03)	0.70 (0.05)	0.74 (0.03)	**0.75 (0.03)**	0.63 (0.03)	0.68 (0.04)	0.72 (0.03)	**0.73 (0.03)**
Mean	0.82 (0.09)	0.86 (0.07)	0.85 (0.08)	**0.87 (0.07)**	0.83 (0.08)	0.86 (0.07)	0.86 (0.08)	**0.87 (0.07)**

Best results are highlighted

F_1-score and decreased the deviation in relation to their non-metric learning baseline models. This suggests that the ML performed by our model produces feature representations that are more suitable for open-set tasks. We omitted the boxplots that measure the Youdens index because they showed very similar behavior.

5.3 Result Analysis

In open-set recognition, it is a good practice to measure the performance of the model at increasing degrees of openness [33]. Figure 4a, b present the mean F_1-score of each experiment ordered by degree of openness. The first issue that comes to attention is the decrease in performance at the 0.32 openness mark. This decrease corresponds to the largest experiment, which uses 30 classes for training and 101 for testing. It is also observed that fine-tuning with TI3D was beneficial in almost every case, emphasizing the robustness of the technique proposed in this work.

To further analyze the performance of TI3D as compared to non-metric learning models, we perform two statistical tests: the Friedman test for repeated measures [9] and the Wilcoxon signed-rank test [55]. We observed a statistically significant difference in F_1-score between models presented in Table 2 when applying the Friedman test. The test resulted in a x_r^2 of 40.404 and a p value $<1 \times 10^{-5}$, indicating that there is a significant difference among the feature extraction models (columns of the Table). We also performed an analysis using a two-tailed Wilcoxon signed-rank test, which compares pairs of models to verify whether they are significantly different. Statistically significant results were found when comparing I3D-K and I3D-K + TI3D ($z = -4.37$, p value

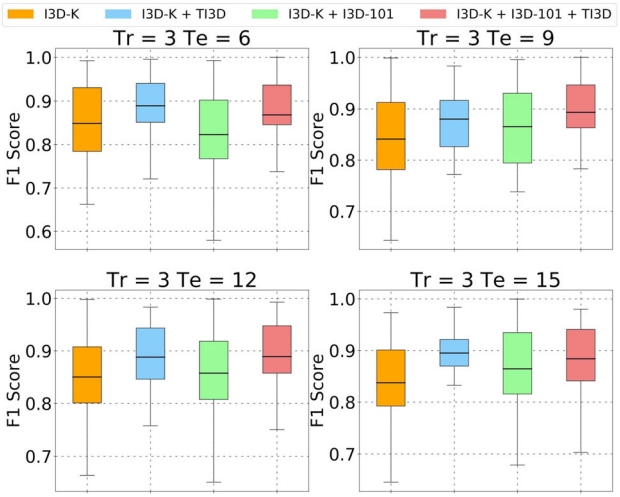

Fig. 3 F_1 Score boxplots of four experiments with 3 known classes. From top to bottom, the lines represent maximum value, third quartile, mean, first quartile, and minimum value

$< 1 \times 10^{-5}$), I3D-101 and I3D-101 + TI3D ($z = -3.91$, p value $= 8 \times 10^{-5}$), I3D-K and I3D-101 ($z = -2.54$, p value $= 1.1 \times 10^{-2}$). These results suggest that fine-tuning with TI3D increases open-set HAR performance. Results also suggest that fine-tuning the model using the known classes with a standard softmax cross-entropy loss also improves the performance, however, at a lower significance level when compared to TI3D.

Another interesting factor is that there is no clear decrease in performance as openness increases, unlike many other works in open-set recognition [4, 31–33]. This suggests that in our case, the difficulty of the problem increases as a function of the raw number of classes, as opposed to the openness. This is shown in Fig. 4c, d, where the experiments are sorted by the number of training and test classes. Notice that the experiment with 30 training classes and 101 test classes produced a clear drop in performance, despite of having an openness of 0.32 (Table 1). This observation was only possible because our experiments allow the same degree of openness to be produced by different numbers of training and test classes, which is often neglected in other works.

Finally, we analyzed the parameter choice related to the EVM classifier. As mentioned in Sect. 4, different values of δ were tested. Finding the optimal value of δ for the base models and the for TI3D is quite different. As shown in Fig. 5a, the I3D-K model reached its optimal F_1 score at $\delta = 1 \times 10^{-5}$, the I3D-K + I3D-101 model at $\delta = 1 \times 10^{-4}$ and the TI3D model at $\delta = 1 \times 10^{-2}$.

It is also noticed that the optimum values of δ are different between F_1 score and Youdens index, as shown in Fig. 5b. This suggests that the value of δ should be chosen according to the interest of the user, where the F_1 score gives more emphasis on correct classification of known classes, and the Youdens index favors the rate of correct acceptance or rejection, that is, the rate in which the classifier successfully discriminates knowns from unknowns.

Overall, the plots show that it is important to perform a search over the EVM δ since very different outcomes can be produced, depending on this parameter. Moreover, it becomes clear that the range of δ values must be wide. A search for δ only in the range of $[1 \times 10^{-1}, 1.5 \times 10^{-1}, \ldots, 3 \times 10^{-1}]$, as proposed by [31], would have missed the optimal values by a large margin in the cases of I3D-K and I3D-K + I3D-101.

6 Conclusions and future works

Human action recognition is a naturally open problem that is often approached as a closed-set scenario. Despite some recent works exploring this issue, open-set HAR is still in its early stages of research. The combination of HAR and open-set recognition inherits the complexities of recognizing intricate spatiotemporal patterns and rejecting those that are unknown, making it a difficult problem to solve.

This work presented a method for performing open-set HAR in videos using a deep metric learning approach named TI3D coupled with the extreme value machine (EVM). The TI3D is built upon the well-known I3D model using a cosine triplet loss function, which maps complex videos to a 256-dimensional feature space.

We have shown that in several experimental settings, the features generated by TI3D consistently outperform the original I3D trained with the standard softmax cross-entropy loss. In this sense, TI3D can be viewed as an extension of the original I3D for open-set classification tasks. Overall, the results suggest that the feature learning process should not be overlooked, even when using an open-set classifier such as the EVM. In other words, more attention should be given to the feature learning part of the process.

This work also introduced an evaluation protocol for open-set HAR using the UCF-101 dataset. Our evaluation was performed using a 30-fold cross-validation procedure. This was done to minimize the chance of randomly selecting "easy" class combinations and produce arguable results. Overall, results revealed that in our case, it was not the degree of openness that dictated the difficulty of the problem, but instead, it was the raw number of training and test classes. It is important to remark that such result was observed because our evaluation protocol allows similar degrees of openness to be achieved by different numbers of training and test classes. This evaluation protocol can be

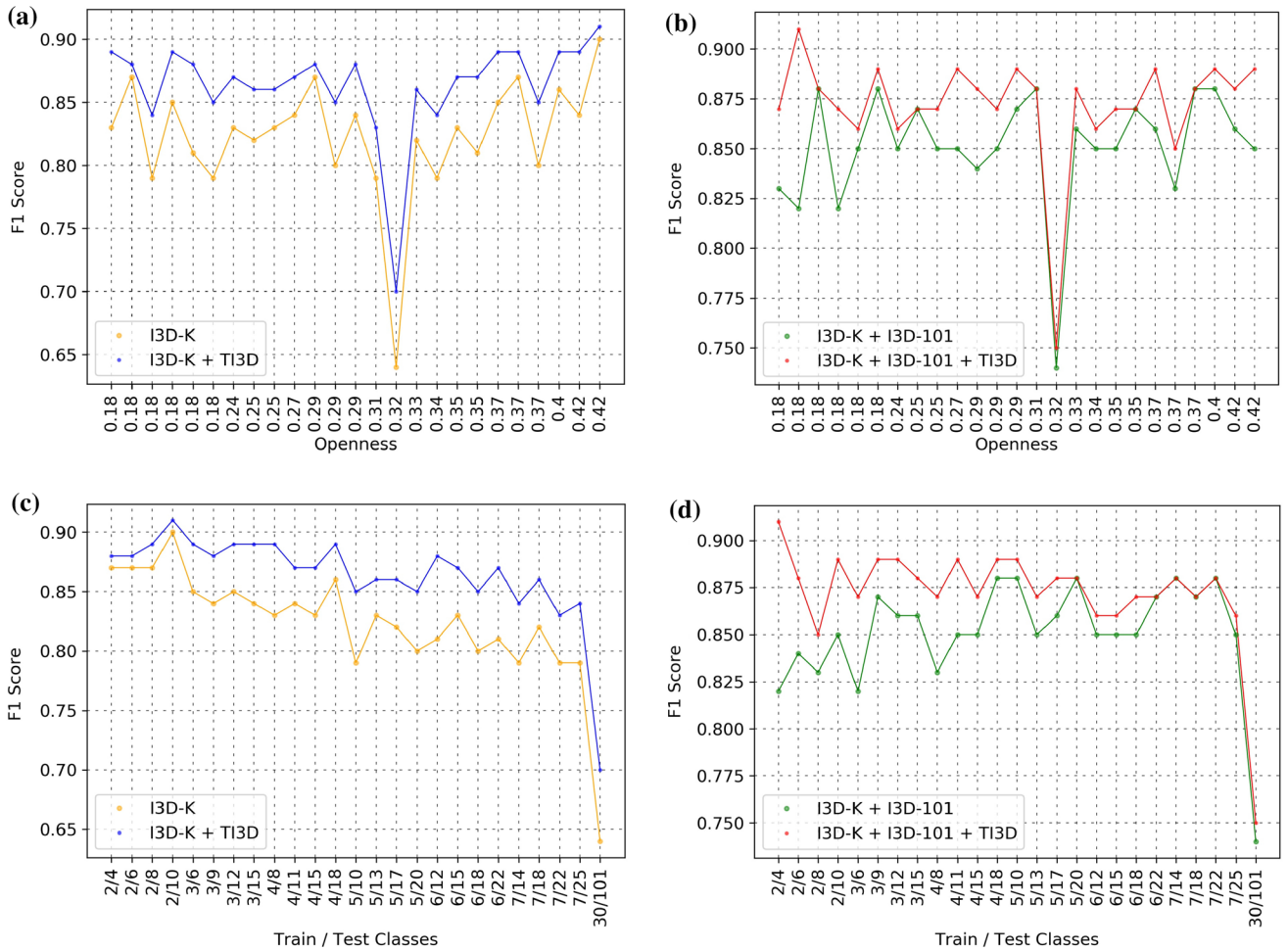

Fig. 4 Comparison between TI3D and the baseline models of experiments ordered by degree of openness (**a**, **b**) and by number of training and test classes (**c**, **d**)

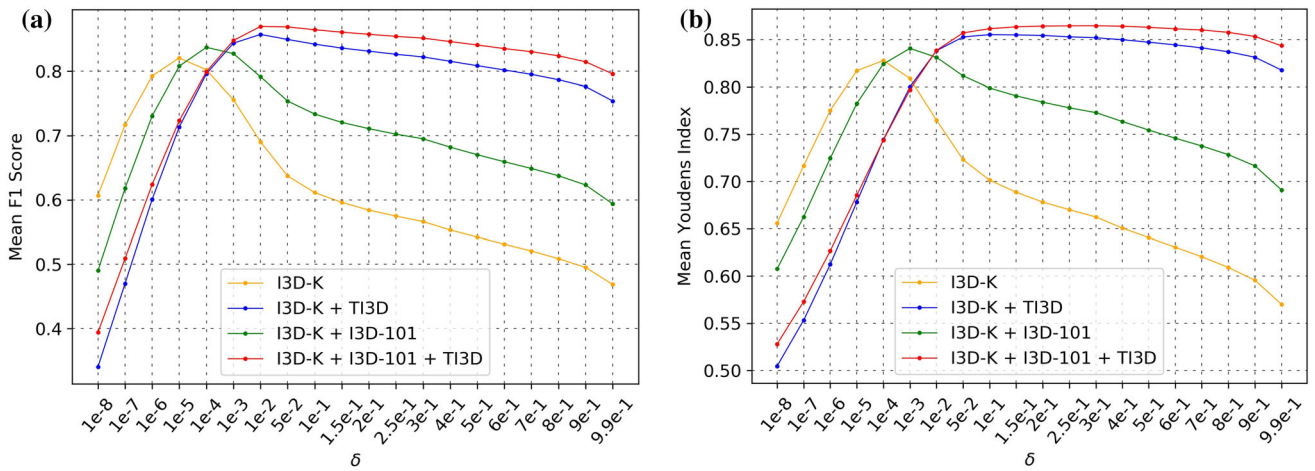

Fig. 5 Mean F_1-Score and Youdens Index averaged across all experiments for each value of δ

extended to other datasets, and higher degrees of openness can be achieved by including experiments with a more substantial portion of unknown classes.

Regarding the evaluation metrics, we observed that both F_1-score and Youdens Index presented similar (but not equal) results. This is expected since both metrics use

recall. The small difference comes from the fact that the F_1-score uses precision, emphasizing true positives, while the Youdens index uses the sensitivity, emphasizing true negatives.

Finally, we investigated the impact of δ on the classification performance of each model. The results suggested that a wide range of δ values should be tested since the EVM seems to be very sensitive to the choice of this parameter. Also, it became clear that there is no universal value for δ, since each model achieved its best performance with different values.

Several research directions can be pointed out for future works. The simpler, yet computationally expensive, is searching for other network architectures, classifiers, and parameters for boosting the classification performance. However, challenges such as new class detection and inclusion of knowledge in existing models are more interesting paths that should be explored. Achieving these goals in HAR is a challenging and under-explored task and, therefore, future work will focus on developing new methods to address HAR as an open-set problem.

Acknowledgements Author M. Gutoski would like to thank CNPq for the Scholarship Number 141983/2018-3; Author H. S. Lopes would like to thank to CNPq for the research Grants 440977/2015-0 and 311785/2019-0. All authors would like to thank NVIDIA Corp. for the donation of the Titan-Xp GPUs used in the experiments.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Aggarwal CC, Hinneburg A, Keim DA (2001) On the surprising behavior of distance metrics in high dimensional space. In: Proceedings of the 8th international conference on database theory (ICDT). Springer, Berlin, pp 420–434
2. Aslan MF, Durdu A, Sabanci K (2019) Human action recognition with bag of visual words using different machine learning methods and hyperparameter optimization. Neural Comput Appl. https://doi.org/10.1007/s00521-019-04365-9
3. Bendale A, Boult T (2015) Towards open world recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). IEEE Press, Piscataway, pp 1893–1902
4. Bendale A, Boult TE (2016) Towards open set deep networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). IEEE press, Piscataway, pp 1563–1572
5. Busto PP, Iqbal A, Gall J (2020) Open set domain adaptation for image and action recognition. IEEE Trans Pattern Anal Mach Intell 42(2):1–15
6. Carreira J, Zisserman A (2017) Quo vadis, action recognition? A new model and the kinetics dataset. In: Proceedings of the 30th IEEE conference on computer vision and pattern recognition (CVPR). IEEE press, Piscataway, pp 4724–4733
7. Chen Y, Kalantidis Y, Li J, Yan S, Feng J (2018) Multi-fiber networks for video recognition. In: Proceedings of the European conference on computer vision (ECCV). Springer International Publishing, pp 352–367
8. Donahue J, Anne Hendricks L, Guadarrama S, Rohrbach M, Venugopalan S, Saenko K, Darrell T (2015) Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). IEEE press, Piscataway, pp 2625–2634
9. Friedman M (1937) The use of ranks to avoid the assumption of normality implicit in the analysis of variance. J Am Stat Assoc 32(200):675–701
10. Gao Z, Zhang H, Liu AA, Xu G, Xue Y (2016) Human action recognition on depth dataset. Neural Comput Appl 27(7):2047–2054
11. Geng C, Chen S (2020) Collective decision for open set recognition. arXiv preprint arXiv:1806.11258
12. Geng C, Huang Sj, Chen S (2018) Recent advances in open set recognition: a survey. arXiv preprint arXiv:1811.08581
13. Glorot X, Bengio Y (2010) Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the 13th international conference on artificial intelligence and statistics. Microtome Publishing, Brookline, pp 249–256
14. Gutoski M, Ribeiro M, Aquino NMR, Lazzaretti AE, Lopes HS (2017) A clustering-based deep autoencoder for one-class image classification. In: Proceedings of the IEEE Latin American conference on computational intelligence. IEEE press, Piscataway, pp 1–6
15. Han D, Li J, Zeng Z, Yuan X, Li W (2017) RegFrame: fast recognition of simple human actions on a stand-alone mobile device. Neural Comput Appl 30(9):2787–2793
16. Hara K, Kataoka H, Satoh Y (2018) Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and imagenet? In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). IEEE press, Piscataway, pp 6546–6555
17. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). IEEE press, Piscataway, pp 770–778
18. Hoffer E, Ailon N (2015) Deep metric learning using triplet network. In: Proceedings of the international workshop on similarity-based pattern recognition. Springer, Heidelberg, pp 84–92
19. Jain LP, Scheirer WJ, Boult TE (2014) Multi-class open set recognition using probability of inclusion. In: European conference on computer vision. Springer, Heidelberg, pp 393–409
20. Kaya M, Bilge HŞ (2019) Deep metric learning: a survey. Symmetry 11(9):1066–1092
21. Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. In: Proceedings of the 25th international conference on neural information processing systems, vol 1. Curran Associates, Red Hook, pp 1097–1105
22. Ladjailia A, Bouchrika I, Merouani HF et al (2019) Human activity recognition via optical flow: decomposing activities into basic actions. Neural Comput Appl. https://doi.org/10.1007/s00521-018-3951-x
23. Lee J, Abu-El-Haija S, Varadarajan B, Natsev AP (2018) Collaborative deep metric learning for video understanding. In: Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery and data mining. ACM, New York, pp 481–490
24. Liu W, Wen Y, Yu Z, Li M, Raj B, Song L (2017) Sphereface: deep hypersphere embedding for face recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). IEEE press, Piscataway, pp 212–220

25. Lu J, Hu J, Zhou J (2017) Deep metric learning for visual understanding: an overview of recent advances. IEEE Signal Process Mag 34(6):76–84
26. Meyer B, Drummond T (2019) The importance of metric learning for robotic vision: open set recognition and active learning. In: Proceedings of the IEEE international conference on robotics and automation (ICRA). IEEE press, Piscataway, pp 2924–2931
27. Ng JYH, Choi J, Neumann J, Davis LS (2018) Actionflownet: learning motion representation for action recognition. In: Proceedings of the IEEE winter conference on applications of computer vision (WACV). IEEE press, Piscataway, pp 1616–1624
28. Ranjan R, Sankaranarayanan S, Bansal A, Bodla N, Chen JC, Patel VM, Castillo CD, Chellappa R (2018) Deep learning for understanding faces: machines may be just as good, or better, than humans. IEEE Signal Process Mag 35(1):66–83
29. Ribeiro M, Lazzaretti AE, Lopes HS (2018) A study of deep convolutional auto-encoders for anomaly detection in videos. Pattern Recognit Lett 105:13–22
30. Roitberg A, Al-Halah Z, Stiefelhagen R (2018) Informed democracy: voting-based novelty detection for action recognition. In: Proceedings of the British machine vision conference. BMVA, Durham
31. Rudd EM, Jain LP, Scheirer WJ, Boult TE (2018) The extreme value machine. IEEE Trans Pattern Anal Mach Intell 40(3):762–768
32. Scheirer WJ, Jain LP, Boult TE (2014) Probability models for open set recognition. IEEE Trans Pattern Anal Mach Intell 36(11):2317–2324
33. Scheirer WJ, Rocha A, Sapkota A, Boult TE (2013) Towards open set recognition. IEEE Trans Pattern Anal Mach Intell 35:1757–1772
34. Scherreik MD, Rigling BD (2016) Open set recognition for automatic target classification with rejection. IEEE Trans Aerosp Electron Syst 52(2):632–642
35. Schölkopf B, Platt JC, Shawe-Taylor J, Smola AJ, Williamson RC (2001) Estimating the support of a high-dimensional distribution. Neural Comput 13(7):1443–1471
36. Schroff F, Kalenichenko D, Philbin J (2015) Facenet: a unified embedding for face recognition and clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). IEEE press, Piscataway, pp 815–823
37. Shu L, Xu H, Liu B (2017) Doc: deep open classification of text documents. In: Proceedings of the conference on empirical methods in natural language processing (EMNLP). Association for Computational Linguistics, Stroudsburg
38. Shu Y, Shi Y, Wang Y, Zou Y, Yuan Q, Tian Y (2018) Odn: opening the deep network for open-set action recognition. In: Proceedings of the IEEE international conference on multimedia and expo (ICME). IEEE press, Piscataway, pp 1–6
39. Simonyan K, Zisserman A (2014) Two-stream convolutional networks for action recognition in videos. In: Proceedings of the advances in neural information processing systems. MIT Press, Cambridge, pp 568–576
40. Soomro K, Zamir AR, Shah M (2012) UCF101: a dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402
41. Szegedy C, Liu W, Jia Y, SermarXivanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). IEEE press, Piscataway, pp 1–9
42. Taigman Y, Yang M, Ranzato M, Wolf L (2014) DeepFace: closing the gap to human-level performance in face verification. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). IEEE press, Piscataway, pp 1701–1708
43. Tax DMJ, Duin RPW (2004) Support vector data description. Mach Learn 54(1):45–66
44. Tong M, Li M, Bai H, Ma L, Zhao M (2019) DKD-DAD: a novel framework with discriminative kinematic descriptor and deep attention-pooled descriptor for action recognition. Neural Comput Appl
45. Tran D, Bourdev L, Fergus R, Torresani L, Paluri M (2015) Learning spatiotemporal features with 3D convolutional networks. In: Proceedings of the IEEE international conference on computer vision (CVPR). IEEE Press, Piscataway, pp 4489–4497
46. Tran D, Wang H, Torresani L, Ray J, LeCun Y, Paluri M (2018) A closer look at spatiotemporal convolutions for action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). IEEE Press, Piscataway, pp 6450–6459
47. Vandersmissen B, Knudde N, Jalalvand A et al (2019) Indoor human activity recognition using high-dimensional sensors and deep neural networks. Neural Comput Appl. https://doi.org/10.1007/s00521-019-04408-1
48. Vinyals O, Blundell C, Lillicrap T, Wierstra D et al (2016) Matching networks for one shot learning. In: Proceedings of the advances in neural information processing systems (NIPS). MIT Press, Cambridge, pp 3630–3638
49. Wang H, Wang Y, Zhou Z, Ji X, Gong D, Zhou J, Li Z, Liu W (2018) Cosface: large margin cosine loss for deep face recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). IEEE Press, Piscataway, pp 5265–5274
50. Wang L, Li W, Li W, van Gool L (2018) Appearance-and-relation networks for video classification. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). IEEE Press, Piscataway, pp 1430–1439
51. Wang L, Xiong Y, Wang Z, Qiao Y, Lin D, Tang X, Van Gool L (2016) Temporal segment networks: towards good practices for deep action recognition. In: Proceedings of the European conference on computer vision (ECCV). Springer, Heidelberg, pp 20–36
52. Wang X, Girshick R, Gupta A, He K (2018) Non-local neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). IEEE Press, Piscataway, pp 7794–7803
53. Wang Y, Zhou W, Zhang Q, Zhu X, Li H (2018) Low-latency human action recognition with weighted multi-region convolutional neural network. arXiv preprint arXiv:1805.02877
54. Weinberger KQ, Saul LK (2009) Distance metric learning for large margin nearest neighbor classification. J Mach Learn Res 10(1):207–244
55. Wilcoxon F (1945) Individual comparisons by ranking methods. Biom Bull 1(6):80–83
56. Wu CY, Zaheer M, Hu H, Manmatha R, Smola AJ, Krähenbühl P (2018) Compressed video action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). IEEE Press, Piscataway, pp 6026–6035
57. Wu L, Wang Y, Gao J, Li X (2018) Where-and-when to look: deep siamese attention networks for video-based person re-identification. IEEE Trans Multimed 21(6):1412–1424
58. Xia P, Zhang L, Li F (2015) Learning similarity with cosine similarity ensemble. Inf Sci 307:39–52
59. Xie S, Sun C, Huang J, Tu Z, Murphy K (2018) Rethinking spatiotemporal feature learning: speed-accuracy trade-offs in video classification. In: Proceedings of the European conference on computer vision (ECCV). Springer, Heidelberg, pp 305–321

60. Xu H, Liu B, Shu L, Yu P (2019) Open-world learning and application to product classification. In: Proceedings of the world wide web conference. ACM, New York, pp 3413–3419
61. Yang Y, Hou C, Lang Y, Guan D, Huang D, Xu J (2019) Open-set human activity recognition based on micro-Doppler signatures. Pattern Recogni 85:60–69
62. Youden WJ (1950) Index for rating diagnostic tests. Cancer 3(1):32–35
63. Zhu Y, Lan Z, Newsam S, Hauptmann A (2018) Hidden two-stream convolutional networks for action recognition. In: Proceedings of the Asian conference on computer vision. Springer, Heidelberg, pp 363–378