Representation Learning for Image Retrieval through 3D CNN and Manifold Ranking

Lucas Barbosa de Almeida[1], Vanessa Helena Pereira-Ferrero[1], Lucas Pascotti Valem[1], Jurandy Almeida[2], and Daniel Carlos Guimarães Pedronette[1]

[1]Department of Statistics, Applied Math. and Computing (DEMAC), São Paulo State University (UNESP), Rio Claro, Brazil
[2]Institute of Science and Technology, Federal University of São Paulo (UNIFESP), São José dos Campos, Brazil
E-mail: barbosa.almeida@unesp.br, nessahelena@gmail.com, lucas.valem@unesp.br, jurandy.almeida@unifesp.br, daniel.pedronette@unesp.br

***Abstract*—Despite of the substantial success of Convolutional Neural Networks (CNNs) on many recognition and representation tasks, such models are very reliant on huge amount of data to allow effective training. In order to improve the generalization ability of CNNs, several approaches have been proposed, including variations of data augmentation strategies. With the goal of achieving more effective retrieval results on unsupervised learning scenarios, we propose a representation learning approach which exploits a rank-based formulation to build a more comprehensive data representation. The proposed model uses 2D and 3D CNNs trained by transfer learning and fuse both representations through a rank-based formulation based on manifold learning algorithms. Our approach was evaluated on an unsupervised image retrieval scenario applied to action recognition datasets. The experimental results indicated that significant effectiveness gains can be obtained on various datasets, reaching +56.93% of relative gains on MAP scores.**

I. Introduction

In the past years, image-based human action recognition has become a very active topic of research, involving diverse machine learning and computer vision techniques [1]. In general, it focuses on identifying a person's action or behavior from images and is an important branch among studies of human perception and computer vision systems [1], [2]. The study and analysis of human action has evolved from earlier schemes, often limited to controlled environments, to recent advanced solutions that can learn from millions of videos and apply to almost every daily activity [3]. Research involving the most important issues in a human action recognition related topics, e.g., on how to create adequate data representations with a high level abstraction, have become increasingly relevant [4]. Recent solutions have applications in domains such as visual surveillance, human-computer interaction, image, and video retrieval, and are challenging due to variations in movement performance and interpersonal differences [5].

Recently, most of the successful results on many of these applications rely on deep learning techniques, which have gained a notable reputation on diverse domains, including image retrieval, object classification and detection [2]. In this scenario, the literature highlights the importance of machine learning decisive tools, such as neural networks, and the growing and constant need to improve its related techniques.

Currently,data-driven approaches often outperform handcrafted methods [6], [7]. The Convolutional Neural Networks (CNNs), mainly inspired by biological processes and the human vision system [6], have become popular particularly due to the ability in handling large amounts of data and the advances in hardware's technology [8]. Notably, CNNs tend to require a minimal level of pre-processing when compared to other image classification algorithms [8]. Other CNNs capacities include performing feature extraction jointly with classification in an end-to-end manner; learn to optimize the features during the training phase directly from the raw input; process large inputs with great computational efficiency; adapt to different input sizes; and are immune to small transformations in the input data, including translation, scaling, skewing and distortion [7].

However, despite such advantages, CNNs often require a huge amount of data for the training stage, which can lead to the limitation of their use in many situations. Therefore, CNNs have to be able to circumvent their inherent characteristics, including their data starving aspect, because of their large amount of learnable parameters to estimate [9]. Additionally, the CNN's performance may degrade sharply when training data are limited [10]. This is one of the reasons why data augmentation strategies can be applied to improve the generalization of CNNs [11], reducing the bottlenecks associated with high requirements on the amount of training data and, consequently, obtaining significant accuracy gains.

To overcome those shortcomings, in this paper, we propose a representation learning approach, that exploits a rank-based formulation to build more comprehensive and effective representations based on the same amount of data. Our approach is used on an unsupervised image retrieval scenario applied on action recognition datasets. A 2D CNN trained through transfer learning is employed to extract initial features, used to compute rankings. The rankings define a sequence of images, which is subsequently used as input to a 3D CNN in order to extract additional features. The rankings are computed based on both 2D and 3D CNNs features and fused with well established manifold learning methods.

An experimental evaluation was conducted to assess the effectiveness of the proposed representation learning approach.

The experiments were performed on three public image datasets for action recognition activities. A challenging unsupervised scenario is considered on image retrieval tasks. The obtained results show that our approach yields significant effectiveness gains, reaching +56.93% of relative gains on MAP scores for the Stanford-40 dataset. Visual analysis was also conducted to evaluate the impact of the proposed approach.

The remainder of this paper is organized as follows. Section II presents the proposed representation learning approach. Section III discusses the CNNs models and Section IV describes the manifold ranking methods. Section V discusses the experimental evaluation and Section VI the conclusions.

II. Rank-based Representation Learning

This section presents the proposed rank-based representation learning approach. Section II-A introduces the main ideas and provides a general view of our method. Section II-A defines the image retrieval model and 2D CNN representation. Section II-C details the 3D CNN representation and Section II-D discusses the fusion of 2D and 3D representations based on manifold learning.

A. Overview

Despite of the substantial success of CNNs on many recognition and representation tasks, such models are very reliant on huge amount of data to allow effective training and to avoid overfitting. In order to improve the generalization ability of these networks, several approaches have been proposed. Data augmentation, for instance, encompasses a suite of techniques that enhance the size and quality of training datasets, aiming at representing a more comprehensive set of possible data [12].

In an analogous direction, we propose the construction of a more comprehensive and effective data representation for image retrieval by exploiting transfer learning and manifold learning algorithms through a rank-based formulation. The main goal is to achieve more effective retrieval results based on representation, provided for unsupervised scenarios, where no labeled data is available.

Figure 1 illustrates the main steps of the proposed representation learning approach. Firstly, we use a transfer learning formulation based on a 2D CNN trained in another large-scale dataset. The 2D CNN features are used to rank the images from the analyzed dataset (step 1). The computed rankings define the sequence of images that are used as input to a 3D CNN. The 3D CNN is also trained by transfer learning and the features extracted are used to compute other sets of rankings (step 2). In the last step, both rankings defined by 2D and 3D CNNs are fused by manifold learning algorithms (step 3), in order to compute the final retrieval results. Each step is detailed and formally defined in the next sections.

B. Image Retrieval through 2D CNN Representation

This section introduces the notation used for image retrieval and ranking tasks, based on related work [13]–[15] and formally defines the 2D CNN representation. Let x denotes an image, it can be formally defined by a pair (D_x, I_x), where:

- D_x is a finite set of points (pixels) in $\mathbb{N}^2$, e.g., $D_x \subset \mathbb{N}^2$
- $I_x : D_x \rightarrow \mathbb{N}^3$ is a function that assigns to each pixel $p \in D_x$ a vector $I(p) \in \mathbb{N}^3$ (when a color in the RGB system is assigned to a pixel).

Let $\mathcal{X}$=$\{x_1, x_2, \ldots, x_n\}$ be an image collection, where n denotes the size of the collection. Ranking and retrieval tasks are performed based on features extracted from the images. Typically, 2D CNNs trained on large-scale datasets as Imagenet [16] are used to extract features for unsupervised tasks through transfer learning. The last fully connected layer are often exploited for feature extraction. In this way, a feature extractor can be formally defined as a function f_2, where the subscript notation refers to a 2D CNN feature. Formally, the function $f_2 : \mathcal{X} \rightarrow \mathbb{R}^d$ computes a d-dimensional vector for a given collection image, such that $\mathbf{v}_{2i} = f_2(x_i)$ and $\mathbf{v}_{2i} = [v_{2i1}, v_{2i2}, \ldots, v_{2id}]$.

A distance function that computes the distance between two images according to the distance between their corresponding feature vectors can be defined as ρ: $\mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}^+$. Therefore, a distance between two images x_i, x_j can be computed by $\rho(\mathbf{v}_{2i}, \mathbf{v}_{2j})$. A general image retrieval task based on extracted features can be modeled as the computation of a ranked list τ_{2q} in response to a query image x_q, according to the distance function ρ. The top positions of ranked lists are expected to contain the most relevant images with regard to the query image, such that the length L of ranked images are often considered, with $L \ll n$. We also refer to neighbors as a small set of similar images given by the top-k ranked images, such that $k \ll L \ll n$.

The ranked list τ_{2q} can be defined as a permutation $(x_1, x_2, \ldots, x_L)$ of the subset $\mathcal{X}_L \subset \mathcal{X}$, which contains the L most similar images to query image x_q, such that and $|\mathcal{X}_L| = L$. Formally, a permutation τ_{2q} is a bijection from the set $\mathcal{X}_L$ onto the set $[n_L] = \{1, 2, \ldots, L\}$. For a permutation τ_q, we interpret $\tau_{2q}(x_i)$ as the position (or rank) of image x_i in the ranked list τ_{2q}. If x_i is ranked before x_j in the ranked list of x_q (i.e., if $\tau_{2q}(x_i) < \tau_{2q}(x_j)$), then $\rho(\mathbf{v}_{2q}, \mathbf{v}_{2i}) \leq \rho(\mathbf{v}_{2q}, \mathbf{v}_{2j})$.

Taking each image $x_i \in \mathcal{X}$ as a query image x_q, a set of ranked lists $\mathcal{T}_2$ can be computed, containing one ranked list for each image in the collection. The computation of $\mathcal{T}_2$ can be accelerated through similarity search approaches [17], based on indexing or hashing structures. Each ranked list establishes a similarity relationship among the query image and all images in the collection $\mathcal{X}$. Therefore, the set $\mathcal{T}_2$ encodes a rich source of similarity/dissimilarity information about the collection $\mathcal{X}$.

C. 3D CNN Representation

While in 2D CNNs, convolutions are applied on the 2D feature maps to compute features from the spatial dimensions [18], 3D CNNs are applied to capture the motion information encoded in multiple contiguous frames [19]. In general, 3D convolutions are performed in stages of CNNs to compute features from both spatial and temporal dimensions [20].

The visual modality information contained in a video can be defined as a sequence of images (or frames), such that $\sigma = (x_{t1}, x_{t2}, \ldots, x_{tm})$, where the subscript t_i denotes the temporal dimension and m denotes the number of images/frames in the video. 3D CNNs are often employed to extract features

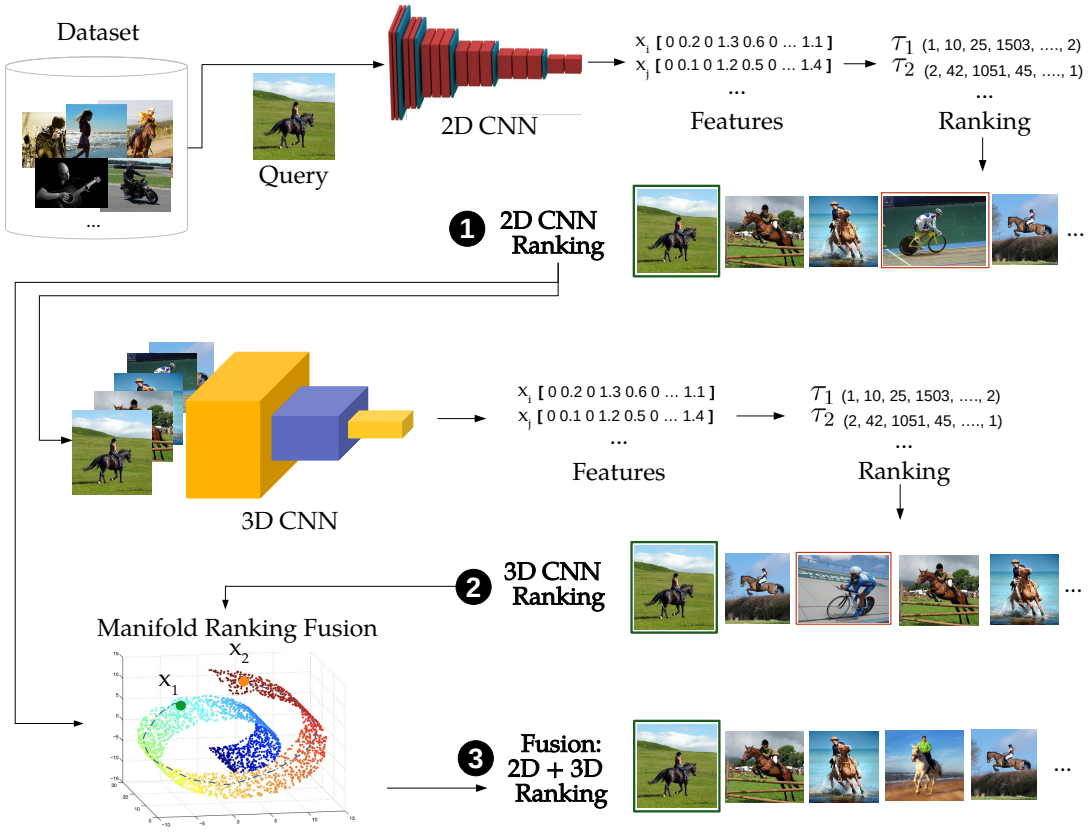


Fig. 1: Illustration of the proposed Representation Learning approach based on 3D CNN and Manifold Ranking.

from videos, e.g., represent the visual information encoded in a video on a d-dimensional vector representation which is used for retrieval and machine learning tasks.

Although the images sequences may be typically defined by a video, it can also be defined by a ranked list. Our hypothesis considers that a ranked list computed in response to a query image can encode relevant contextual similarity information about the image, in replacement to the temporal dimension. In fact, it often provides a diversified representation about the respective class, once top positions are expected to contain different relevant images of the same class of the query image.

A ranked list computed based on a 2D CNN representation is exploited in order to determine the sequence. Let k_r denotes the size of the sequence defined by the ranked list. Let $\mathcal{N}_2(x_q, k_r)$ be the set of k_r most similar to x_q according to the 2D CNN feature and ranked lists and defined as:

$$\mathcal{N}_2(x_q, k_r) = \{\mathcal{C} \subseteq \mathcal{X}, |\mathcal{C}| = k_r \wedge \forall x_i \in \mathcal{C}, \\ x_j \in \mathcal{X} - \mathcal{C} : \tau_{2q}(i) < \tau_{2q}(j)\}. \quad (1)$$

The sequence σ_q is a permutation defined as a bijection from the set $\mathcal{N}_2(x_q, k_r)$ onto the set $\{1, 2, \ldots, k_r\}$, which follows the order of the ranked list τ_{2q}. If x_i is ranked before x_j in the sequence σ_q (that is, if $\sigma_q(x_i) < \sigma_q(x_j)$), then $\tau_{2q}(x_i) \leq \tau_{2q}(x_j)$. A sequence can be defined for each image $x_i \in \mathcal{X}$ in order to compute a set of sequences $\mathcal{S} = \{\sigma_1, \sigma_2, \ldots, \sigma_n\}$. Each sequence can be analyzed by a 3D CNN aiming to extract novel features. Formally, a 3D CNN can be defined as a function $f_3 : \mathcal{S} \rightarrow \mathbb{R}^d$, that computes a d-dimensional vector for an image sequence, such that $\mathbf{v}_{3i} = f_3(\sigma_i)$ and $\mathbf{v}_{3i} = [v_{3i1}, v_{3i2}, \ldots, v_{3id}]$.

Analogously to 2D CNN features, ranked lists can be computed based on 3D CNN features. The ranked list τ_{3q} can be also defined as a bijection from the set $\mathcal{X}_L$ onto the set $[n_L] = \{1, 2, \ldots, L\}$, such that if $\tau_{3q}(x_i) < \tau_{3q}(x_j)$, then $\rho(\mathbf{v}_{3q}, \mathbf{v}_{3i}) \leq \rho(\mathbf{v}_{3q}, \mathbf{v}_{3j})$.

Every image $x_i \in \mathcal{X}$ can also be taken as a query image x_q, in order to obtain a set of ranked lists $\mathcal{T}_3$. In this way, the features extracted for each image x_i by a 3D CNN encode not only spatial information, but also contextual similarity information from its ranked list τ_{2i}. Once the sequences processed by 3D CNNs contain a more diversity representation for the images, the set of ranked lists $\mathcal{T}_3$, it is expected to improve the generalization provided by the representation and, consequently, improving the comprehensiveness often evaluated by the recall measure.

D. Manifold Ranking Fusion

The sets of ranked lists computed based on both 2D and 3D CNN features encodes relevant and complementary similarity information. While the set $\mathcal{T}_2$ provides the original and more precise similarity information, the set $\mathcal{T}_3$ provides a more diverse similarity representation. Therefore, such information can be combined to achieve a more effective similarity measure and ranking.

Manifold learning approaches have been recently exploited to improve and combine the set of ranked lists [14], [15], [21].

Ranking and retrieval tasks are often performed by pairwise comparisons of points in a high-dimensional feature space, using Euclidean-like distance functions. However, traditional pairwise measures ignore the complex similarity arrangements and the structural information of the dataset manifold. In order to address such limitations, manifold learning methods have been proposed based on more global measures, capable of taking into account the structure of datasets and providing a more effective similarity measurement.

The main objective of the rank-based manifold learning method is to exploit the similarity information encoded in the set of ranked lists, being able to capture global similarity information encoded on the dataset manifold. Based on such analysis, a new and more effective set of ranked lists can be computed, improving the effectiveness of ranking and retrieval tasks. Considering the two sets of ranked lists $\mathcal{T}_2$ and $\mathcal{T}_3$, given by 2D and 3D CNNs, a manifold ranking fusion task can be defined as a function f_m:

$$\mathcal{T}_f = f_m(\mathcal{T}_2, \mathcal{T}_3) \tag{2}$$

The set $\mathcal{T}_f$ is expected to contain more effective ranked lists which can be used in retrieval tasks.

III. 2D and 3D CNN Models

This section discusses the 2D and 3D CNN features used to instantiate the proposed representation learning approach.

A. 2D CNN

The pre-trained model used for the extraction of initial features in order to generate a first set of rankings of the images is a variant of Residual Network (ResNet), proposed by He et al. [22], named ResNet-18. This model has 18 layers with learnable parameters and was trained for the classification task on the ImageNet [16] dataset.

The ResNet architecture is a special case of the CNN architecture which popularized the idea of "*skip connections*", also known as shortcut connections. According to He et al [22], "*with increasing network depth, accuracy becomes saturated and then degrades rapidly*". To solve this problem, they used residual blocks, whose underlying idea is to include a short-circuit mechanism between every two layers of the ordinary network, adding the input directly to the output. In this way, the more layers, the smaller the change of each layer to the input, making residual networks easy to optimize and achieve higher accuracy when the depth of the network increases, producing results that are better than non-residual networks.

This model was chosen due to its excellent results in many tasks. A similar network implemented by the same authors, but with greater depth, ranked first place in ImageNet detection, ImageNet localization, COCO detection, and COCO segmentation at ILSVRC COCO 2015 competitions. The implementation of the model used in our experiments was taken from a Github repository[1] and is coded in Python and Pytorch [23] framework.

[1]https://github.com/pytorch/vision/blob/master/torchvision/models

B. 3D CNN

For the video feature extraction, we used a pre-trained ResNet 3D Model proposed by Monfort et al. [24]. This model is an Inflated 3D ResNet, defined by the process proposed by Carreira et al. [25], where using a pre-trained 2D model, all the polling kernels and filters are inflated to a third dimension, to be able to deal with the temporal dimension (filters $N \times N$ become $N \times N \times N$). The weights are then replicated from the 2D kernel, over the temporal dimension. The reason behind this procedure is that 3D models contain much more parameters than their equivalent 2D, because of their third dimension. And also this procedure has proven to improve the learning efficiency and performance of 3D models, compared to a initialization from scratch.

The 3D model was inflated from the best 2D ResNet of Monfort et al. [24] and trained on the Moments in Time [24] dataset, which has more than 1 million videos, distributed in 339 distinct classes. This 3D model is based on ResNet-50, which has 50 layers with learnable parameters and is fed with 16 video frames sampled at 5 fps. Among the architectures and approaches executed by Monfort et al. [24], the inflated 3D Resnet was the one that stood out the most. The 3D model used in our experiments was also implemented in Python and Pytorch [23] framework and its code was taken from a public Github repository[2]

IV. Manifold Ranking Methods

This section discusses two manifold ranking methods used to instantiate the proposed approach. Both methods are used to fuse the ranked lists of 2D and 3D CNNs and are publicly available on the framework UDLF[3].

A. LHRR

The method called LHRR (Log-based Hypergraph of Ranking References) [21] uses a hypergraph model to explore the similarity information and transform it into ranking models. Graphs are commonly represented by sets of vertices (nodes) and their corresponding connections (edges or links). Hypergraphs, on the other hand, are a generalization of these graphs that allow the connection of any number of vertices and the representation of higher-order similarity relations.

Based on the ranking references, the representation of hypergraphs is constructed. To build a contextual representation of data samples, the hyperedges approach is used. Following a log-based function, weights are assigned to images in each hyperedge. Through it, it is possible to explore the encoded similarity information. Such similarity is obtained through the result of the product of the similarities with their respective hyperedges. The goal then is to obtain a more effective similarity function. The idea of this new unsupervised calculated set and a new computed similarity function is to use them to improve the effectiveness of the final ranking results. The LHRR method is used in manifold ranking tasks, to improve the effectiveness of retrieval results, since it is capable of identifying more reliable similarity relations and capturing the

[2]https://github.com/zhoubolei/moments_models/

[3]https://github.com/UDLF/UDLF

geometric structure of datasets. The LHRR [21] can also be used for rank fusion tasks, which is the objective that we used the method in this work.

B. BFS-Tree

Using a tree structure, the BFS-Tree Manifold Learning (Breadth-First Search Tree of Ranking References) algorithm [15] is applied to exploit the similarity information encoded in the ranking references. In order to obtain the top-k ranking results, the Breadth-First Search Tree (BFS) provides a hierarchical representation of the ranking results, by encoding the first and second-order neighborhood relationships obtained through ranking references. Calculated based on the rank correlation measures, the edge weights assigned to the elements of the tree represent the similarity.

To discover underlying similarity relationships, the BFS-Tree is exploited. Tree elements are represented based on their path to the root and their respective weights. Between the leaves, new connections are established. Such connections make it possible to discover new relationships of similarity. A tree structure also allows, in addition to new similarity connections, to analyze the frequency of elements in the tree. Commonly, a solid indication of similarity can be obtained by the co-occurrence of elements at different levels of the tree structure, while a low occurrence can be an indication of noise. Thanks to the consideration of similarity information extracted from all constructed trees, it is possible to compute a more global and effective similarity measure between pairs.

V. Experimental Evaluation

This section describes the experimental evaluation conducted to assess the effectiveness of the proposed approach. Section V-A describes the datasets and the experimental protocol. Section V-B discusses the results and Section V-C presents a visual analysis.

A. Datasets and Experimental Protocol

Three public datasets were used in our experimental evaluation, described in the following:

1) Willow Actions Dataset: The Willow Actions [26] is composed of 911 static images distributed in seven classes of actions. Its images were extracted from Flickr and have only one of seven actions (Interact with Computer, Photograph, Playing Instrument, Riding a Bike, Horseback Riding, Running, and Walking). In general, they have a simple background, without many elements beyond the action.

2) Ikizler Dataset: The second dataset, named Ikizler Dataset [27], is a collection of 1972 images, divided into ten classes (boxing-punching, dining, handshaking, high-five, hugging, kicking, kissing, partying, speech, and talking), where each class has at least 150 images. This dataset is considerably more complex than the Willow Actions, as it has classes with very similar actions, such as handshaking and high-five.

3) Stanford 40 Actions Dataset: Finally, the third dataset was the Stanford 40 Actions Dataset [28], composed of 9532 images and has 40 different action classes with at least 180 images for each category. Due to the higher number of classes, compared to the others datasets, it becomes an even more significant challenge to obtain an efficient retrieval in the dataset. The images that form the dataset were extracted from Bing, Google, and Flickr.

For all three datasets, the same parameters settings were used. The parameter values of manifold learning algorithms followed the default values available through the UDLF framework [29]. Regarding the parameter k_r, which defines the neighborhood size of the proposed representation learning method, we evaluate three different configurations $\{5, 10, 15\}$. As for effectiveness measures, we considered Precision and Mean Average Precision (MAP).

B. Results

Tables I, II, and III present the results of the proposed approach on the Stanford 40 Actions, Ikizler, and Willow Actions datasets, respectively. We can observe that fusion based on manifold ranking methods (τ_f) showed higher effectiveness gains compared to the models isolation (τ_2,τ_3). We can also notice that the scenario with neighborhood size $k_r = 5$ achieved the best results in most of the scenarios considered Although, in general, the use of the BFS-Tree method yielded better results than LHRR, it is possible to notice that for the Stanford 40 Actions and Ikizler datasets in the precision($P@x$) metric the LHRR outperforms the BFS-Tree at some depths x (for values of x equals or lower than 10 for the Ikizler and lower or equals than 15 for the Stanford).

The relative gains obtained by the proposed approach based on manifold rank fusion are significant in all scenarios. Particularly, for the MAP metric, on the Stanford-40 dataset, the absolute gains reach up to +12.73% in relation to the 2D model (22.36% to 35.09%) and +9.72% comparing with the 3D model (25.37% to 35.09%). Considering the Ikizler dataset and MAP metric, the absolute gain is up to +12.61% in contrast to the 2D model (33.65% to 46.26%) and 1.98% in regard to the 3D model (44.28% to 46.26%). Finally, on the Willow Actions dataset, the absolute gains reach up to +14.9% comparing with the 2D model (47.47% to 62.37%) and +9.68% in relation to the 3D model (52.69% to 62.37%). Considering relative gains, the results are even more impressive, with gains up to +56.93%, +37.47% and +31.38% on Stanford-40, Ikizler and Willow datasets, respectively.

C. Visualization Analysis

With the intention of enriching the discussion about the proposed approach, we employed dimensionality reduction methods to represent the impact of the method on a 2-D projection of feature space. The analysis was performed on the three aforementioned datasets, using the t-SNE [30] algorithm.

Figure 2, shows the visualizations of the application of t-SNE on the datasets Stanford-40, Ikizler, and Willow Actions. For each dataset, it presents respectively in a 2D plane, the distance obtained from the features of the 2D model, followed by the fusion of rankings of the 2D and 3D model by the LHRR and BFS-TRee algorithms. As we can notice on the representations, both approaches based on manifold ranking fusion resulted in better separability of classes scenarios. The same behavior can be observed for all datasets compared to the initial ranking, obtained from the 2D CNN model in isolation.

TABLE I: Results of representation learning based on manifold ranking and 3D CNN on Stanford 40 Dataset.

	Fusion	**P@5**	**P@10**	**P@15**	**P@20**	**P@30**	**P@50**	**P@100**	**MAP**
2D CNN	-	0.6170	0.5422	0.5049	0.4805	0.4476	0.4052	0.3431	0.2236
3D CNN ($k_r = 5$)	-	0.5653	0.4936	0.4626	0.4440	0.4203	0.3911	0.3490	0.2537
3D CNN ($k_r = 10$)	-	0.5431	0.4706	0.4400	0.4218	0.3979	0.3693	0.3283	0.2364
3D CNN ($k_r = 15$)	-	0.5246	0.4503	0.4204	0.4028	0.3797	0.3532	0.3162	0.2297
2D + 3D CNN ($k_r = 5$)	LHRR	**0.6280**	**0.5655**	**0.5374**	**0.5193**	0.4953	0.4635	0.4164	0.3234
2D + 3D CNN ($k_r = 10$)	LHRR	0.6191	0.5557	0.5286	0.5108	0.4850	0.4533	0.4072	0.3129
2D + 3D CNN ($k_r = 15$)	LHRR	0.6051	0.5429	0.5167	0.4982	0.4740	0.4430	0.3983	0.3078
2D + 3D CNN ($k_r = 5$)	BFS-Tree	0.6202	0.5565	0.5306	0.5161	**0.4965**	**0.4730**	**0.4371**	**0.3509**
2D + 3D CNN ($k_r = 10$)	BFS-Tree	0.6088	0.5466	0.5213	0.5062	0.4866	0.4623	0.4275	0.3416
2D + 3D CNN ($k_r = 15$)	BFS-Tree	0.6046	0.5389	0.5115	0.4951	0.4747	0.4522	0.4186	0.3352

TABLE II: Results of representation learning based on manifold ranking and 3D CNN on Ikizler Dataset.

	Fusion	**P@5**	**P@10**	**P@15**	**P@20**	**P@30**	P@50	P@100	MAP
2D CNN	-	0.6578	0.5878	0.5539	0.5331	0.5073	0.4704	0.4136	0.3365
3D CNN ($k_r = 5$)	-	0.6859	0.6272	0.6045	0.5906	0.5698	0.5459	0.5048	0.4428
3D CNN ($k_r = 10$)	-	0.6629	0.6090	0.5871	0.5716	0.5530	0.5271	0.4855	0.4211
3D CNN ($k_r = 15$)	-	0.6258	0.5649	0.5377	0.5235	0.5062	0.4828	0.4437	0.3856
2D + 3D CNN ($k_r = 5$)	LHRR	**0.6902**	**0.6334**	0.6094	0.5947	0.5722	0.5413	0.4962	0.4445
2D + 3D CNN ($k_r = 10$)	LHRR	0.6835	0.6240	0.5980	0.5823	0.5597	0.5297	0.4835	0.4310
2D + 3D CNN ($k_r = 15$)	LHRR	0.6534	0.5982	0.5688	0.5545	0.5362	0.5092	0.4643	0.4115
2D + 3D CNN ($k_r = 5$)	BFS-Tree	0.6798	0.6328	**0.6111**	**0.5961**	**0.5784**	**0.5546**	**0.5198**	**0.4626**
2D + 3D CNN ($k_r = 10$)	BFS-Tree	0.6802	0.6222	0.5997	0.5877	0.5698	0.5454	0.5083	0.4519
2D + 3D CNN ($k_r = 15$)	BFS-Tree	0.6581	0.5978	0.5744	0.5608	0.5422	0.5185	0.4836	0.4275

TABLE III: Results of representation learning based on manifold ranking and 3D CNN on Willow Dataset.

	Fusion	**P@5**	**P@10**	**P@15**	**P@20**	**P@30**	**P@50**	**P@100**	**MAP**
2D CNN	-	0.7745	0.7259	0.6959	0.6776	0.6479	0.5964	0.5011	0.4747
3D CNN ($k_r = 5$)	-	0.7524	0.7069	0.6859	0.6712	0.6483	0.6172	0.5463	0.5269
3D CNN ($k_r = 10$)	-	0.7324	0.6835	0.6591	0.6444	0.6207	0.5877	0.5197	0.5035
3D CNN ($k_r = 15$)	-	0.7183	0.6748	0.6553	0.6435	0.6261	0.6048	0.5434	0.5254
2D + 3D CNN ($k_r = 5$)	LHRR	0.7842	0.7423	0.7240	0.7106	0.6915	0.6628	0.5942	0.5924
2D + 3D CNN ($k_r = 10$)	LHRR	0.7748	0.7325	0.7172	0.7003	0.6785	0.6473	0.5786	0.5775
2D + 3D CNN ($k_r = 15$)	LHRR	0.7701	0.7268	0.7096	0.7003	0.6774	0.6434	0.5846	0.5801
2D + 3D CNN ($k_r = 5$)	BFS-Tree	**0.7897**	**0.7447**	**0.7301**	**0.7168**	**0.7037**	**0.6773**	**0.6229**	**0.6237**
2D + 3D CNN ($k_r = 10$)	BFS-Tree	0.7719	0.7360	0.7198	0.7059	0.6873	0.6589	0.6044	0.6048
2D + 3D CNN ($k_r = 15$)	BFS-Tree	0.7706	0.7338	0.7169	0.7058	0.6878	0.6588	0.6109	0.6086

In another visual analysis to assess the effectiveness of the proposed approach, Figures 3, 4, and 5 illustrates the ranked lists computed by the 2D CNN model and by the manifold ranking fusion of 2D+3D by the BFS-Tree method. The red borders indicate images that do not belong to the same class of the query image. In this set of representations, it is possible to visualize that the impact of our approach is especially remarkable for certain instances of each dataset.

VI. Conclusion

In this paper, we proposed a representation learning approach, aiming to improve the comprehensiveness of representations and effectiveness of image retrieval. It uses rankings generated by a pre-trained 2D model and builds a sequence analyzed by a pre-trained 3D model. Both representations have their rankings fused by manifold learning algorithms.

In the experimental evaluation, our approach achieved significant effectiveness gains on retrieval tasks conducted on action recognition datasets. In all scenarios and datasets, the proposed approach achieved better results in comparison with the 2D CNN model in isolation. Our results are promising for representation learning. As future works, we intend to investigate the use of the proposed approach in multimodal scenarios, aiming to fuse information from multiple modalities in unsupervised multimedia retrieval scenarios.

VII. Acknowledgements

The authors are grateful to São Paulo Research Foundation - FAPESP (grants #2020/02183-9, #2020/03311-0, #2018/15597-6, and #2017/25908-6), Brazilian National Council for Scientific and Technological Development - CNPq (grant #309439/2020-5) and Microsoft Research.

References

[1] G. Guo and A. Lai, "A survey on still image based human action recognition," *Pattern Recognition*, vol. 47, no. 10, pp. 3343–3361, 2014.

[2] F. Zhu, L. Shao, J. Xie, and Y. Fang, "From handcrafted to learned representations for human action recognition: A survey," *Image and Vision Computing*, vol. 55, pp. 42–52, 2016.

[3] S. Herath, M. Harandi, and F. Porikli, "Going deeper into action recognition: A survey," *Image and vision computing*, vol. 60, pp. 4–21, 2017.

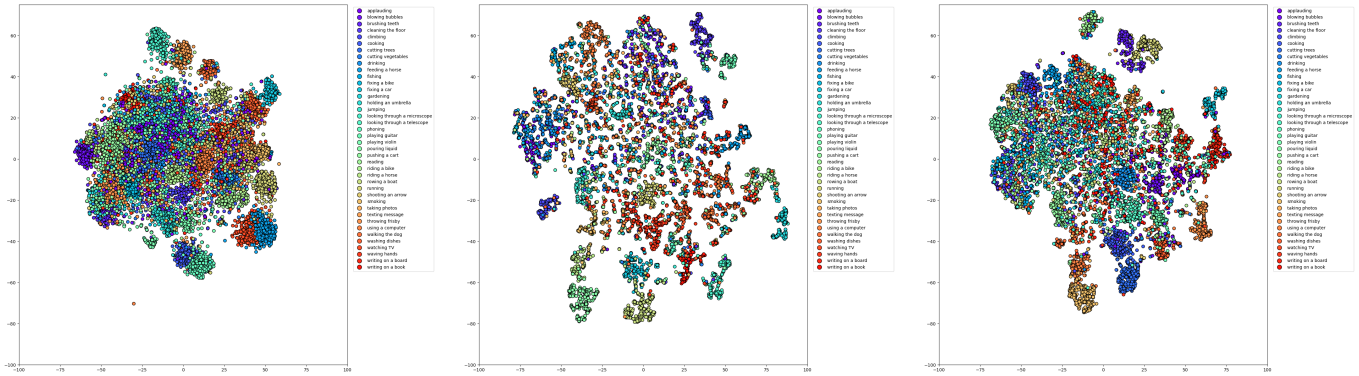
[4] M. Koohzadi and N. M. Charkari, "Survey on deep learning methods in human action recognition," *IET Computer Vision*, vol. 11, no. 8, pp. 623–632, 2017.

[5] Y. Kong and Y. Fu, "Human action recognition and prediction: A survey," *arXiv preprint arXiv:1806.11230*, 2018.

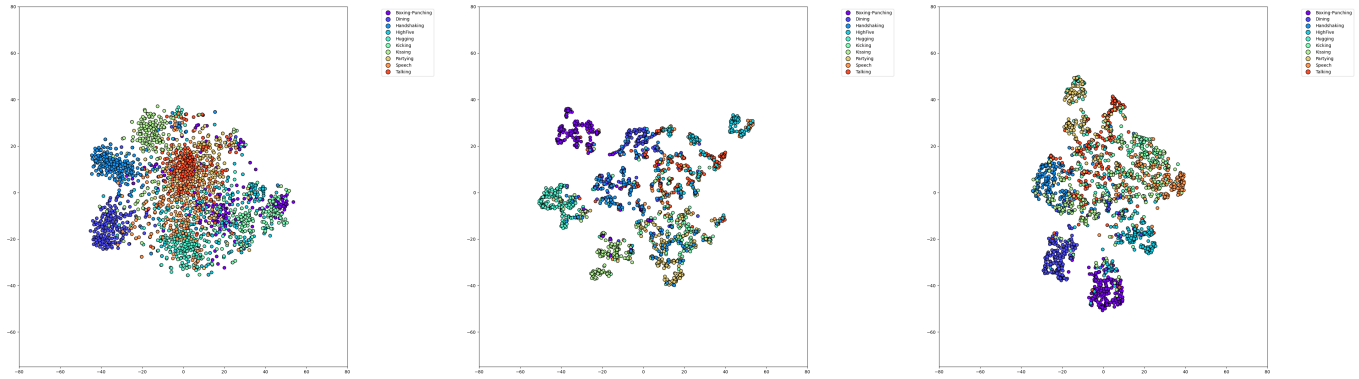
[6] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *Int. Conf. Engineering and Technology (ICET'17)*, 2017, pp. 1–6.

[7] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, "1d convolutional neural networks and applications: A survey," *Mechanical systems and signal processing*, vol. 151, p. 107398, 2021.

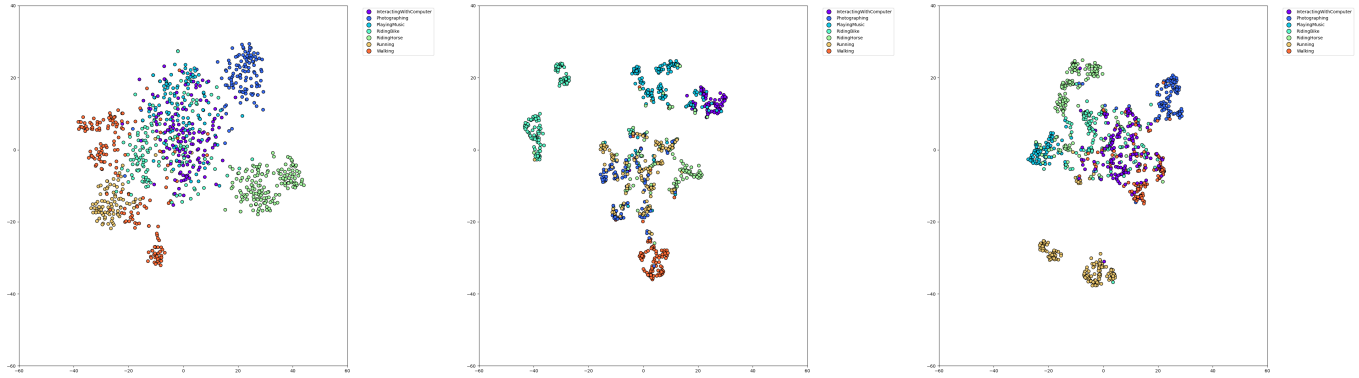
[8] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," *Artificial Intelligence Review*, vol. 53, no. 8, pp. 5455–5516, 2020.



(a) Stanford by original ResNet2D. (b) Stanford by ResNet2D+3D (LHRR). (c) Stanford by ResNet2D+3D (BFS-Tree).



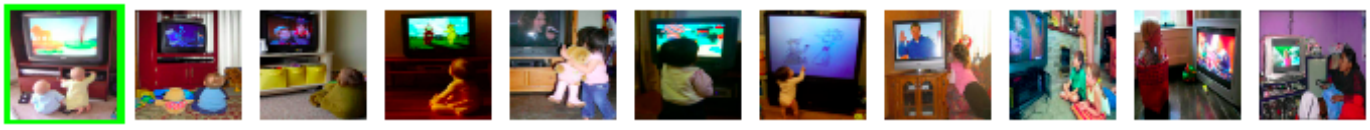
(d) Ikizler by original ResNet2D. (e) Ikizler by ResNet2D+3D (LHRR). (f) Ikizler by ResNet2D+3D (BFS-Tree).



(g) Willow by original ResNet2D. (h) Willow by ResNet2D+3D (LHRR). (i) Willow by ResNet2D+3D (BFS-Tree).

Fig. 2: Visualization of 2-D projections generated by t-SNE for different datasets (Stanford, Ikizler, and Willow), considering the original features (ResNet 2D) and the proposed representation learning approach (ResNet2D+3D by LHRR or BFS-Tree).

(a) Retrieval results by ResNet 2D features.



(b) Retrieval results by ResNet 2D + 3D fused by BFS-Tree manifold ranking.

Fig. 3: Visual retrieval results on Stanford dataset using the original ResNet 2D representation and the proposed approach. Query image in green borders and wrong results (from other classes) in red borders.

(a) Retrieval results by ResNet 2D features.

(b) Retrieval results by ResNet 2D + 3D fused by BFS-Tree manifold ranking.

Fig. 4: Visual retrieval results on Ikizler dataset using the original ResNet 2D representation and the proposed approach. Query image in green borders and wrong results (from other classes) in red borders.

(a) Retrieval results by ResNet 2D features.

(b) Retrieval results by ResNet 2D + 3D fused by BFS-Tree manifold ranking.

Fig. 5: Visual retrieval results on Willow dataset using the original ResNet 2D representation and the proposed approach. Query image in green borders and wrong results (from other classes) in red borders.

[9] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into imaging*, vol. 9, no. 4, pp. 611–629, 2018.

[10] H. S. Mousavi, T. Guo, and V. Monga, "Deep image super resolution via natural image priors," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'18)*, 2018, pp. 1483–1487.

[11] T. Agarwal, N. Sugavanam, and E. Ertin, "Sparse signal models for data augmentation in deep learning ATR," in *IEEE Radar Conference (RadarConf'20)*, 2020, pp. 1–6.

[12] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 60, 2019.

[13] R. da S. Torres and A. X. Falcão, "Content-Based Image Retrieval: Theory and Applications," *Revista de Informática Teórica e Aplicada*, vol. 13, no. 2, pp. 161–185, 2006.

[14] D. C. G. Pedronette, F. M. F. Gonçalves, and I. R. Guilherme, "Unsupervised manifold learning through reciprocal knn graph and connected components for image retrieval tasks," *Pattern Recognition*, vol. 75, pp. 161 – 174, 2018.

[15] D. Carlos Guimarães Pedronette, L. P. Valem, and R. da S. Torres, "A bfs-tree of ranking references for unsupervised manifold learning," *Pattern Recognition*, vol. 111, p. 107666, 2021.

[16] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *Int. J. Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[17] L. C. Shimomura, R. S. Oyamada, M. R. Vieira, and D. S. Kaster, "A survey on graph-based methods for similarity searches in metric spaces," *Information Systems*, vol. 95, p. 101507, 2021.

[18] S. F. Santos and J. Almeida, "Faster and accurate compressed video action recognition straight from the frequency domain," in *Conference on Graphics, Patterns and Images (SIBGRAPI'20)*, 2020, pp. 62–68.

[19] S. F. Santos, N. Sebe, and J. Almeida, "CV-C3D: action recognition on compressed videos with convolutional 3d networks," in *Conference on Graphics, Patterns and Images (SIBGRAPI'19)*, 2019, pp. 24–30.

[20] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221–231, 2013.

[21] D. C. G. Pedronette, L. P. Valem, J. Almeida, and R. d. S. Torres, "Multimedia retrieval through unsupervised hypergraph-based manifold ranking," *IEEE Trans. Image Processing*, vol. 28, no. 12, pp. 5824–5838, 2019.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR'16)*, 2016, pp. 770–778.

[23] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*.

[24] M. Monfort, B. Zhou, S. Bargal, A. Andonian, T. Yan, K. Ramakrishnan, L. Brown, Q. Fan, D. Gutfruend, C. Vondrick, and A. Oliva, "Moments in time dataset: One million videos for event understanding," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PP, 01 2018.

[25] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR'17)*, 2017, pp. 4724–4733.

[26] V. Delaitre, I. Laptev, and J. Sivic, "Recognizing human actions in still images: a study of bag-of-features and part-based representations," 2010, updated version, available at http://www.di.ens.fr/willow/research/stillactions/.

[27] G. Tanisik, C. Zalluhoglu, and N. Ikizler-Cinbis, "Facial descriptors for human interaction recognition in still images," *Pattern Recognition Letters*, vol. 73, pp. 44–51, 2016.

[28] B. Yao, X. Jiang, A. Khosla, A. L. Lin, L. Guibas, and L. Fei-Fei, "Human action recognition by learning bases of action attributes and parts," in *Int. Conf. Computer Vision (ICCV'11)*, 2011, pp. 1331–1338.

[29] L. P. Valem and D. C. G. Pedronette, "An unsupervised distance learning framework for multimedia retrieval," in *ACM International Conference on Multimedia Retrieval, ICMR*, 2017, pp. 107–111.

[30] L. van der Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 11 2008.