An Instance-Based Learning Statistical Framework for One-Shot and Few-Shot Human Action Recognition

Mark Haddad

A Thesis

in the

Concordia Institute for Information Systems Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of

Master of Applied Science (Quality System Engineering) at

Concordia University

Montréal, Québec, Canada

November 2021

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Mark Haddad**

Entitled: **An Instance-Based Learning Statistical Framework for One-Shot and Few-Shot Human Action Recognition**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science (Quality System Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

_____ Chair
Dr. Abdessamad Ben Hamza

_____ External Examiner
Dr. Mohsem Ghafouri

_____ Examiner
Dr. Abdessamad Ben Hamza

_____ Supervisor
Dr. Nizar Bouguila

Approved by _____
Dr. Mohammed Mannan, Chair
Department of Concordia Institute for Information Systems Engineering

_____ 2021

_____
Dr. Mourad Debbabi, Dean
Faculty of Engineering and Computer Science

Abstract

An Instance-Based Learning Statistical Framework for One-Shot and Few-Shot Human Action Recognition

Mark Haddad

Along with the exponential growth of online video creation platforms such as TikTok and Instagram, state of the art research involving quick and effective human action/gesture recognition remains crucial. This thesis presents an instance-based statistical framework which addresses the challenge of classifying short human action video clips, using a domain-specific feature design approach, capable of performing significantly well using as little as one training example per action (one-shot learning). The method is based on Gunner Farneback's dense optical flow (GF-OF) estimation strategy, Gaussian mixture models, and information divergence. We first aim to obtain accurate representations of the human movements/actions by clustering the results given by GF-OF using K-means method of vector quantization. We then proceed by representing the result of one instance of each action by a Gaussian mixture model. Furthermore, using Kullback-Leibler divergence (KL-divergence), we estimate the information divergences in an attempt to find similarities between the trained actions and the ones in the test videos. Classification is then done by matching each test video to the trained action with the highest similarity (a.k.a lowest KL-divergence value). We have performed experiments on the KTH and Weizmann Human Action datasets using One-Shot and K-Shot learning approaches, and the results reveal the discriminative nature of our proposed methodology in comparison with state-of-the-art techniques.

Acknowledgments

I would like to express my sincere appreciation and gratitude to my supervisor, Prof. Nizar Bouguila, who supported me throughout my Master's program. He not only helped me make this work possible but encouraged and supported me in my professional career by offering me the tools and flexibility I needed to thrive. His teaching was of the highest standard and I am grateful for the opportunity to work with him.

To my parents, everything I have and everything I am, I owe it all to you. The sacrifices you've made for me are beyond any description. Thank you for making this possible.

To my brother, Dr. Alexandre, thank you for being a role model and an inspiration to me. The support and care you have given me cannot go unnoticed. I am proud to have you in my life.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this thesis, we propose a novel action recognition framework whose goal is to classify short action video clips to their respective actions by automatically matching their representations to trained ones. The trained representations are essentially labeled instances of each action, as shown in the upper left of Fig. 1.1, that are used in a few-shot learning setting to achieve a few-shot action recognition task. The framework is flexible enough to be extended in various ways according to the application and could for example be integrated in users' devices to classify and organize their videos using little training data.

Requests for new action (e.g. dance) challenges are emerging on a daily basis, and our framework is designed to be able to effectively learn each new action using as little as one instance of it, and classify new videos using the learned instances. An overview of the process flow is displayed in Fig. 1.2 and goes as follows: Initially, the input dataset is split into training and testing sets that both go through the same feature extraction process. This process initially tracks the actors in the videos and places a bounding box around them, computes the dense optical flow inside the box, and clusters the optical flow vectors using the KMeans algorithm. Subsequently, classification is achieved using a similarity check method which employs Gaussian Mixture Models and Kullback-Leibler divergence between the KMeans clusters of the training and testing videos. A visual representation of the similarity measurement is demonstrated in Fig. 1.1, in which the KMeans cluster centers of a trained "waving" action are used in an attempt to find similar movement patterns in a test video.

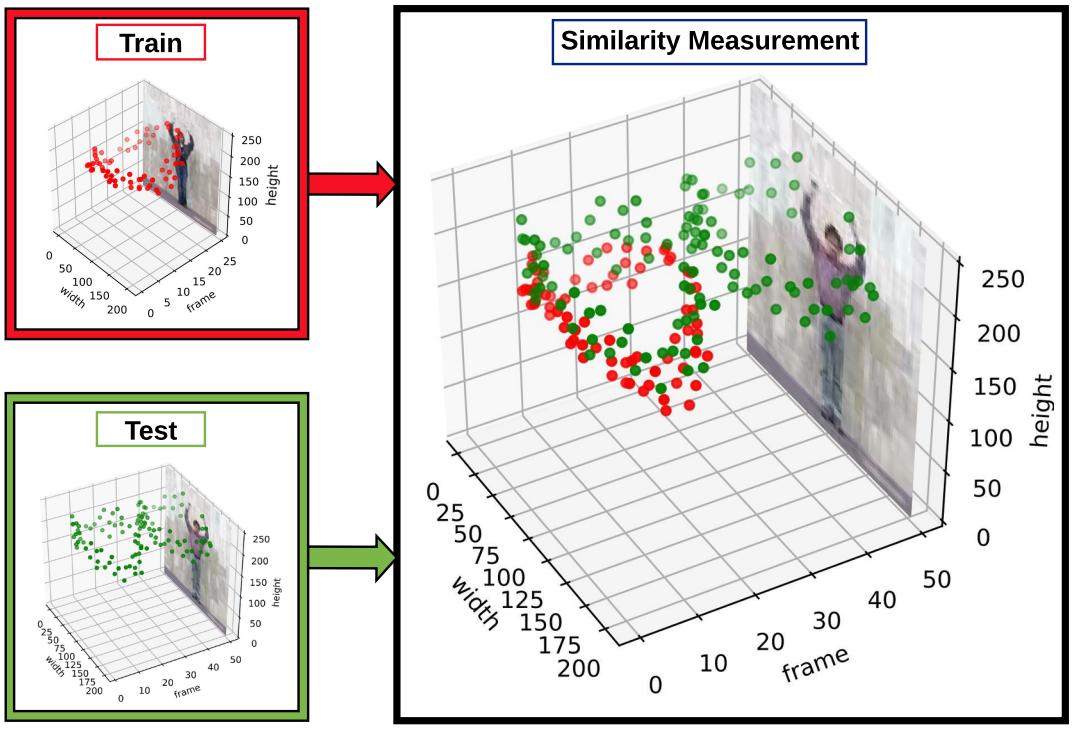

Figure 1.1: Similarity between clustered optical flow centers (KMeans) of a trained "Two-hands wave" action (Red) and a test one (Green) from the Weizmann Human Action dataset.

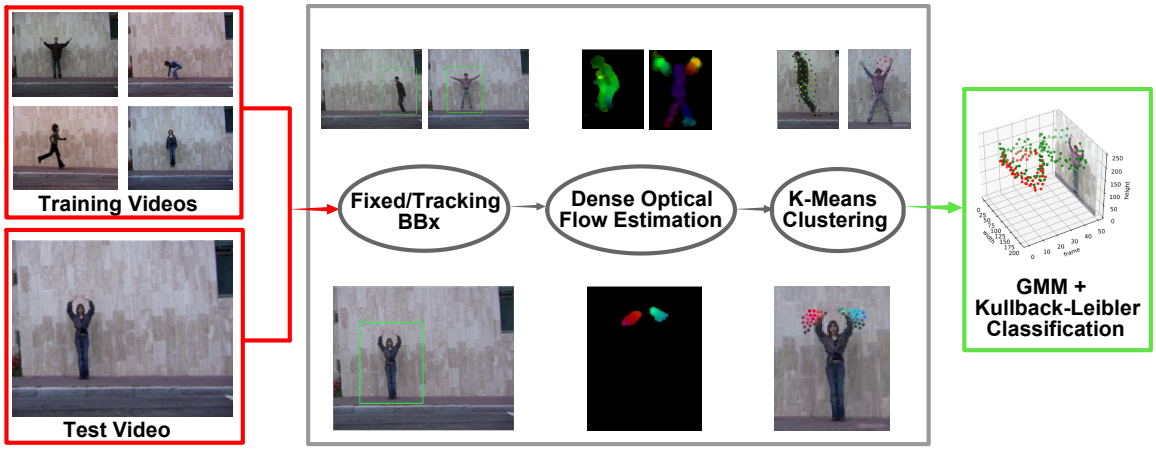

Figure 1.2: Overview of the process flow. The training and testing videos (shown in red) are used as inputs and go through a feature extraction process (shown in grey), followed by a similarity check process (shown in green) in which classification is achieved.

1.1 Background

Recognition is a field that concentrates on a classical problem in computer vision, which is determining whether the information on images or video frames contains a specific feature, object, or activity. Such field includes "object recognition", "Human action recognition", "identification" and "detection" [20, 21, 22, 23, 24, 25, 26, 27]. On the other hand, applying deep learning, image restoration and classification has facilitated the study over different sub-branches of recognition. Some of the novel deep learning, image restoration and classification applications which are recognized for this purpose can be found in [12, 13, 14, 15, 16, 17, 18, 19].

More specifically, the "Human action recognition" field is an active and important area in computer vision. Related comprehensive research works can be found in [7, 8, 45, 28]. In this regard, spatiotemporal interest points and feature descriptors for human action recognition have been researched in [1, 2, 3], which include a wide range of methodologies and described as "Bag of visual and video words". The strength of these methodologies is their robustness to occlusion, whereas their drawback is their locality and distribution of content understanding, and their sensitivity to several intermediate processes such as classifiers. There is another set of techniques that focuses on detecting a bounding box, which includes the person executing the action. These methods include spatiotemporal shapes using contours for body tracking [29], spatiotemporal volumes using silhouette images [9] and space-time gestures [31]. Such methodologies ignore the primitive human sub-actions, which are considered a drawback for their representations.

There is another set of methodologies which considers the location knowledge or body parts appearances. For instance, landmark trajectory features of body parts have been researched in [42]. Additionally, the learning of cascade of filters has been proposed by Ke *et al.* [38] for accurate spatiotemporal localization and detection purposes. Such approaches are challenging issues in the field of human action recognition since a completely supervised strategy is not ensured.

The problem of long-term visual tracking has been addressed in [33] where ascribable to deformation, abrupt motion, heavy occlusion and out-of-view, the target objects undergo significant appearance variation. Accordingly, the task of tracking into translation and scale estimation of objects has been decomposed. In another work, an adaptive region proposal scheme with feature

channel regularization has been provided for facilitating robust object tracking [11]. Correspondingly, the unsupervised video object segmentation task has been addressed in [34] where the method was denominated as CO-attention Siamese Network (COSNet). Recently, another video object segmentation (VOS) work has been proposed which unlike most existing methods which rely heavily on extensive annotated data, this method addresses object pattern learning from unlabeled videos [35].

1.2 Contributions

Despite the significant progress which has formerly been performed, there are several challenges in the field of human action recognition. For instance, the variation of the camera position relative to the subject may create confusions in the human action detection and classification. Moreover, similarities in different action categories may cause action misclassifications. In this work, we have overcome such challenges by presenting a human action representation and classification framework that automatically matches human action test videos to trained ones. The action representation is based off the repetitive nature of human actions and can be utilized effectively in one-shot or k-shot learning settings [4, 30]. The importance of our contributions can be described as follows:

- Representing human actions considering their repetitive nature using Gaussian Mixture Models coupled with K-Means method of vector quantization for instance-based learning of multidimensional data
- Classifying human actions by automatically matching their representations to trained ones in videos using Kullback-Leibler divergence for one-shot and few-shot human action classification

Our research work has been accepted and published as:

M. Haddad, V. K. Ghassab, F. Najar and N. Bouguila, "A statistical framework for few-shot action recognition", Multimed Tools Appl 80, 24303–24318 (2021)

M. Haddad, V. K. Ghassab, F. Najar and N. Bouguila, "Instance-Based Learning for Human Action Recognition", 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC),

2020, pp. 147-153

1.3 Thesis Overview

The rest of the thesis is arranged as follows. Chapter 2 describes the methodology and the mathematical background behind the Gaussian Mixture models and K-Means method of vector quantization that have been applied with the goal of achieving instance-based learning of multidimensional data. The section has been split into two sub-sections, one focused on Parametrized Displacement Fields 2.1, and one 2.2 regarding Gaussian Mixture Models for human action representation. Chapter 3 covers the Information Divergence Estimation method that has been employed in the proposed framework 3.1, as well as all the experimental settings, results, and an empirical examination of the experimental outcomes 3.2. The limitations and possible extensions have been displayed in 3.3, and the conclusion of the thesis has been presented in Section 4.

Chapter 2

Gaussian Mixture Models and K-Means Method of Vector Quantization for Instance-Based Learning of Multidimensional Data

The process in which the input video data is translated into features capable of being interpreted by our model is presented in this section. This includes the optical flow used to estimate the movement between each two consecutive frames, and the combination of KMeans and Gaussian Mixture Models on multidimensional data to accurately represent human actions.

2.1 Parameterized Displacement Fields

2.1.1 Gunnar Farneback Optical Flow

In this subsection, we are going to describe the parametrized displacement fields which we have applied in Farneback optical flow estimation considering two consecutive video frames using the eight-parameter model in a two dimensional space [40, 41]. For this purpose, we define the global parameterized displacements considering polynomials which represent the neighborhood of a pixel

in each of our video frames as follows

$$d_x(x,y) = a_1 + a_2x + a_3y + a_7x^2 + a_8xy,$$
$$d_y(x,y) = a_4 + a_5x + a_6y + a_7xy + a_8y^2, \tag{1}$$

where x and y are the horizontal and vertical coordinates of corresponding pixels in two consecutive video frames; and d_x and d_y illustrate the parametrized displacement polynomial with respect to x and y. Furthermore, $a_1, a_2, \cdots, a_8$ are expansion coefficients considering the polynomial expansions of both video frames.

Eq. 1 can be rewritten as

$$D = PS, \tag{2}$$
$$P = \begin{pmatrix} 1 & x & y & 0 & 0 & 0 & x^2 & xy \\ 0 & 0 & 0 & 1 & x & y & xy & y^2 \end{pmatrix}, \tag{3}$$
$$S = (a_1 \;\; a_2 \;\; a_3 \;\; a_4 \;\; a_5 \;\; a_6 \;\; a_7 \;\; a_8)^T, \tag{4}$$

where $D = \langle d_x, d_y \rangle$ is the global displacement, and P and S stand for the polynomial matrix and the solution, respectively. In addition, the polynomial expansion is the neighborhood approximation of each pixel with a polynomial. Accordingly, the quadratic polynomial in a local coordinate system can be represented as

$$F(X) \sim X^T AX + BX + C, \tag{5}$$

where $X = \langle x, y \rangle$ is a pixel vector considering its direction in the video frame and F is the polynomial expansion of pixels neighborhood in the video frame. Furthermore, A, B, and C are the coefficients of such neighborhood polynomial expansion where A represents a symmetric matrix, B is a vector and C is considered a scalar. By applying the global displacement in Eq. 5, we end up

with

$$\begin{aligned} F(X-D) &\sim (X-D)^T A(X-D) + B^T(X-D) + C \\ &= X^T AX + (B-2AD)^T X + D^T AD \\ &\quad - B^T D + C. \end{aligned} \tag{6}$$

Accordingly, by defining $B' = B - 2AD$ and $\Delta B = \frac{B'-B}{2}$ and considering Eq. 3, we minimize the following weighted least square problem for calculating our desired solution

$$\sum_j \omega_j \| A_j P_j S - \Delta B_j \|, \tag{7}$$

where j is the pixel index and ω_j represents the weight of the corresponding pixel. Therefore, the solution is calculated as follows

$$S = (\sum_j \omega_j {P_j}^T {A_j}^T A_j P_j)^{-1} \sum_j \omega_j {P_j}^T {A_j}^T \Delta B_j. \tag{8}$$

2.1.2 Vector Quantization of Optical Flow using KMeans

The application of the parametrized displacement fields solution displayed in Eq. 8 is illustrated on the left side of Fig. 2.1, in which a "bending" action from the Weizmann Human Action dataset has been used as an input. Furthermore, the right side is the result of clustering of the optical flow vectors using KMeans. The following sub-section describes how this feature extraction method may be utilized on a spatiotemporal level to obtain a valuable representation tool for human action classification.

2.2 Gaussian Mixture Models and Mahalanobis Distance for Human Action Representation

As seen in Fig. 2.2, obtaining KMeans clusters gives an accurate 3D representation of the movement in a video. The idea behind our approach involves the use of just one (or k) example(s) of

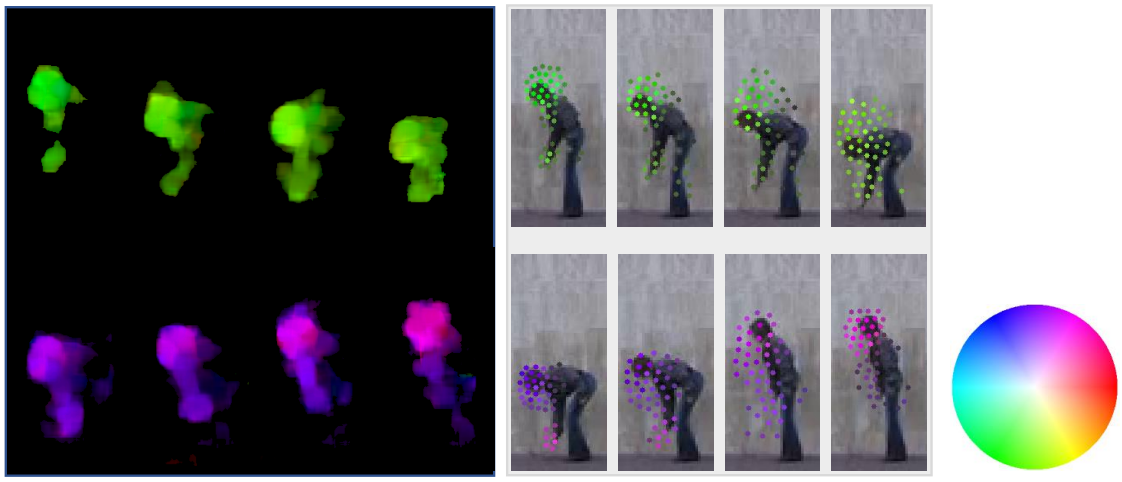

Figure 2.1: Application of dense optical flow (on left) and K-means clustering (on right) on a "Bending" action from the Weizmann Human Action dataset. Colors correspond to the flow magnitude and direction, as per the color wheel.

each action in the training phase. Since our work is instance-based oriented, we focus on obtaining a representation of a single instance/repetition of each action. In videos which contain many repetitions of the same action such as a video of a person performing numerous jumping jacks, we only focus on one of the occurrences, typically the one that looks the most representative or general for the action.

Once an instance for each action is obtained during the training phase, we employ the following method to compare those instances to groups of KMeans points found in test videos. In this regard, we propose that the KMeans clusters of each action instance be modeled by a mixture of Gaussian distributions, resulting in a set of Gaussian components defined as follows

$$p(x|\Theta) = \sum_{j=1}^{M} p_j \mathcal{N}(x; \mu_j; \Sigma_j), \tag{9}$$

where p_j is the mixing parameter of component j ($0 \leq p_j \leq 1, \sum_{j=1}^{M} p_j = 1$), Θ is the set of all the parameters $(p_1, \ldots, p_M, \mu_1, \ldots, \mu_M, \Sigma_1, \ldots, \Sigma_M)$ and $\mathcal{N}(x; \mu_j; \Sigma_j)$ is the j-th Gaussian

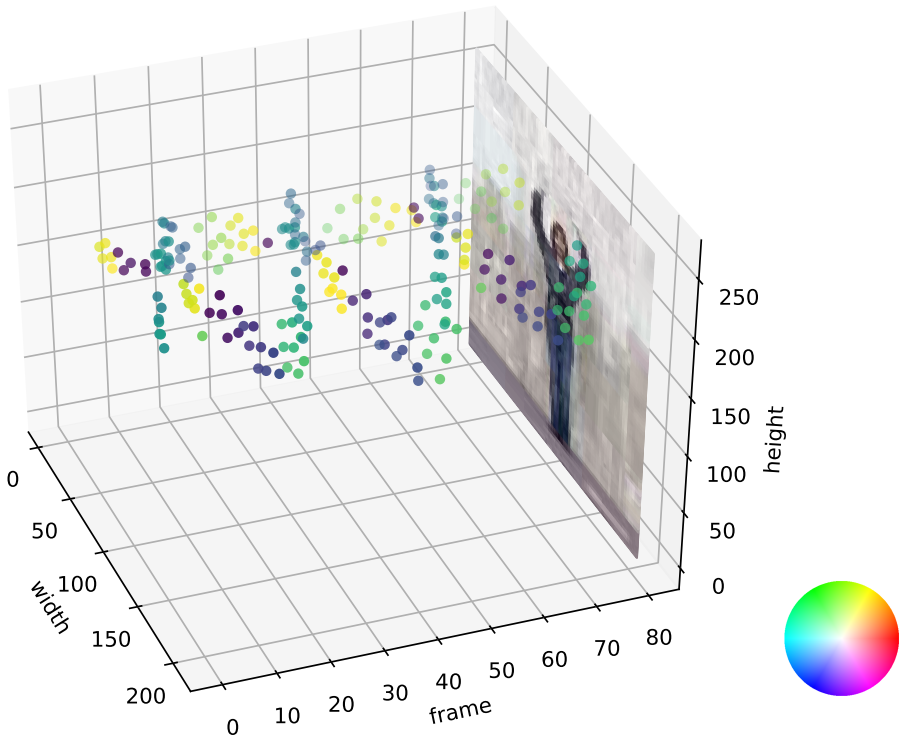

Figure 2.2: 3D Representation of K-means clusters of optical flow for the "Wave 2" action of Daria from Weizmann Human Action dataset. Colors correspond to the mean flow magnitude and direction of each cluster, as per the color wheel. Three full repetitions of the action are clearly discernable.

distribution given by the mean μ_j and the covariance matrix parameter Σ_j

$$\mathcal{N}(x;\mu_j;\Sigma_j) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma_j|^{1/2}} \times \exp\{-\frac{1}{2}(x-\mu_j)^T \Sigma_j^{-1}(x-\mu_j)\}. \tag{10}$$

Each Gaussian component represents part of an action, meaning that the mean μ of each component is a 5D vector (x, y, u, v, t) representing the position (x, y, t) and magnitude (u, v) of a group of KMeans clusters that constitute a lower-level action, or sub-action (e.g. left arm moving up, right leg moving to the left).

The parameters of each Gaussian mixture model are then estimated using the Expectation-Maximization estimation algorithm (EM) where the log-likelihood is derived with respect to the mean, the covariance matrix, and the mixing weight. Starting with the expected value of the posterior probabilities, all the parameters are updated until convergence of the likelihood. Subsequently, Kullback-Leibler (KL) divergence measure is employed in an attempt to find similarities between the GMM representations of the trained action instances, and the ones being generated in different sections of each test video. The aforementioned is detailed and discussed in the following section.

Lower level features within a human action [7] may also be represented more accurately by employing a larger number of Gaussian components during the modeling of the Gaussian mixture, then computing the sum of bidirectional Mahalanobis distances d_{ij} between each consecutive Gaussian components i and j on the temporal axis t:

$$d_{ij} = (\hat{t_i} - \mu_j)^T \Sigma_j^{-1} (\hat{t_i} - \mu_j) + (\bar{t_j} - \mu_i)^T \Sigma_i^{-1} (\bar{t_j} - \mu_i) \tag{11}$$

where t_i and t_j are the forward and backward transition predictions of Gaussian components i and j, respectively:

$$\hat{t_i} = (x_i + fu_i, y_i + fv_i, u_i, v_i) \tag{12}$$

$$\bar{t_j} = (x_j + fu_j, y_j - fv_j, u_j, v_j) \tag{13}$$

x, y, u and v stand for the first 4 dimensions of the mean μ of a Gaussian component, as described earlier in section 2.2, and f stands for the temporal difference between the Gaussian components which is determined in section 3.1. Distances with values below a certain threshold are then removed and different sub-actions are separated from each other.

Following in-depth experimentation, we have concluded that the application of Mahalanobis distance is extremely sensitive to the location of the Gaussian components, which have a direct relation to the feature extraction method being employed. After careful tuning of the feature extraction parameters, the computed distance between Gaussian components of different actions may still be significant enough to surpass the specified threshold, causing an unwanted connection between unrelated sub-actions. For this reason, we refrained from adding this step, and used the previously proposed method for representing human actions in a much more computationally-efficient and flexible manner.

Chapter 3

Information Divergence Estimation using Kullback-Leibler Divergence for One-Shot and Few-Shot Human Action Classification

Following the representation of human actions using KMeans and Gaussian Mixture Models, it is now possible to find similarities between different actions by employing a method based on Kullback-Leibler Divergence. This can be conducted in a one-shot or few-shot setting. In this section, the similarity check process, as well as all the conducted experiments that prove the effectiveness of the proposed framework will be presented.

3.1 Kullback-Leibler Divergence

Considering the single Gaussians $p(x) = \mathcal{N}(x; \mu_p; \Sigma_p)$ and $q(x) = \mathcal{N}(x; \mu_q; \Sigma_q)$, the KL-divergence is represented as follows [44]

$$KL_{GMM}(p||q) = \frac{1}{2}\Big[\log\frac{|\Sigma_p|}{|\Sigma_q|} + \text{tr}(\Sigma_q^{-1}\Sigma_p) - k + (\mu_q - \mu_p)^T\Sigma_q^{-1}(\mu_q - \mu_p)\Big], \tag{14}$$

where k is the dimension of both distributions, μ_p and μ_q stand for the mean values of the Gaussians, Σ_p and Σ_q represent the covariance values and $\text{tr}(\cdot)$ the trace of a matrix.

In order to compute the KL-divergence between two GMMs, we consider the approximation proposed by Goldeberger *et al.* [43] as follows

$$KL_{GMM}(f||g) = \sum_{i=1}^{m}\omega_{f,i}(KL_G(f_i||g_{\pi(i)}) + \log\frac{\omega_{f,i}}{\omega_{g,\pi(i)}}), \tag{15}$$

where $\pi(i) = \arg\min_j(KL_G(f_i||g_j) - \log\omega_{g,j})$, f and g are two GMMs including f_i and g_i for $i \in \{1, \cdots, m\}$ as their Gaussian distributions. Moreover, $\omega_{f,i}$ and $\omega_{g,i}$ are the corresponding weights and m is the total number of Gaussian components.

Since each trained action has its own GMM representation, the classification process is done by matching each test video to the trained action with which the KL-divergence value was the lowest. The proposed classification framework is described in Alg. 1.

3.2 Experimental Results

The conducted experiments involved training the model using a set of actions and classifying test ones using one-shot and k-shot learning approaches. Trials were conducted using several assumptions in an attempt to increase the representation quality of each action, hence maximizing classification accuracy.

Algorithm 1: The proposed classification framework using similarity measurement

function Classification (KM, n, GMM);
Input : KMeans parameters KM of test videos $\mathcal{X}$
Number of frames n of training $block_t$
Gaussian Mixture Model parameters GMM of training $block_t$
Output: Classification labels t of videos $\mathcal{X}$
foreach *Video* X_i *in* $\mathcal{X}$ **do**
  foreach $block_j$ *in the set of* $blocks_n$ **do**
    GMM_j = GMM($block_j$), Eq. 9;
    $KL_j = KL(block_j, block_t)$, Eq. 15;
  end
  min_i = Min KL_j
  Label X_i as t where $min_i = KL(block_j, block_t)$
end
return labels t

3.2.1 Datasets

Our experiments were conducted on the Weizmann Human Action [37] and the KTH [47] datasets, which contain actions that resemble the ones seen on online platforms (Short, contain one or more repetitions of the action and recorded using a fixed camera). The Weizmann dataset contains 90 low-resolution videos consisting of 10 natural actions, as seen in Fig. 3.1 (bend, jumping jack, jump forward, jump in place, gallop sideways, run, skip, walk, wave one hand, wave two hands). As for the KTH dataset, it contains 600 action videos involving 25 subjects performing 6 different actions, as seen in Fig. 3.2 (walking, jogging, running, boxing, hand waving, hand clapping) in 4 dissimilar scenarios.

3.2.2 Data Preprocessing

The goal is to pre-process all the data in a way that will maximize the quality of the extracted features while discarding unmeaningful ones. Initially, the input videos were resized according to their initial resolution using similar scale factors for both heights and widths. This was achieved by resizing each frame using either an area interpolation when resizing down, or a bicubic interpolation when resizing up, to cover up for lost information. Videos with lower resolutions can benefit from

Figure 3.1: Actions from the Weizmann Human Action dataset [37]

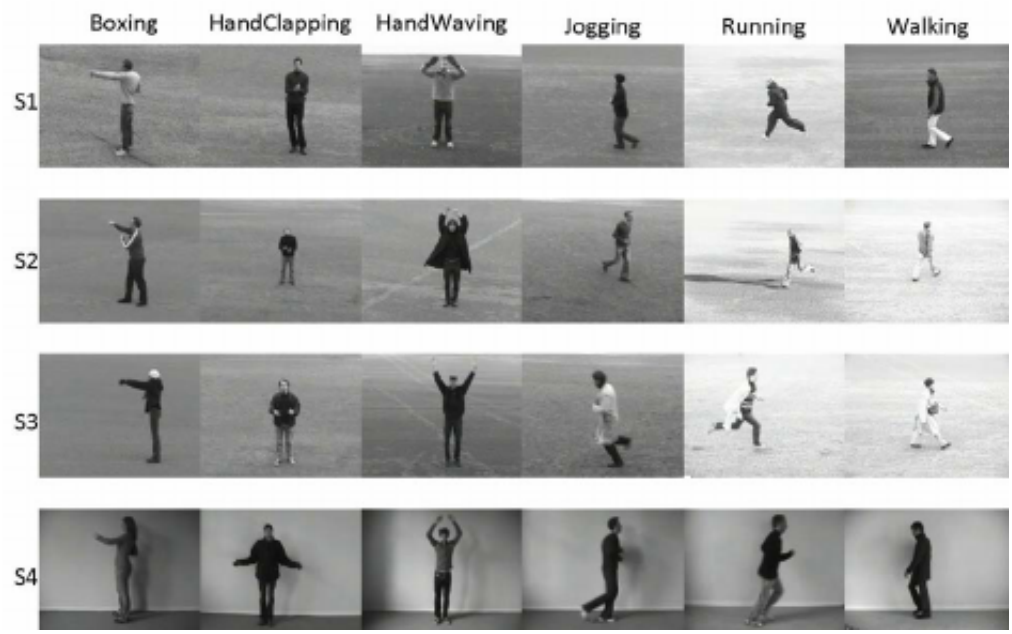

Figure 3.2: Actions from the KTH dataset [47]

being resized to a larger size prior to feature extraction, especially when followed by optical flow-based methods as the one that has been described in section 2.1. In contrast, higher resolution videos are typically heavy in terms of processing and may be problematic when it comes to storage space. Furthermore, the creation and the use of additional and often unnecessary features in the model also affect the performance of the model itself. Therefore, the clips can benefit from being resized to a lower specified size.

Followingly, a fixed size bounding box *(BBx)* was employed around the actors in each video to maximize classification accuracy while minimizing background noise. For actions that involve

significant lateral movement, a high-speed tracking method based on kernelized correlation filters [46] was employed on the actors to automatically keep the *BBx* around them. Additionally, the velocity of the *BBx* was substracted from the optical flow vectors within it. This enables us to focus solely on the articulation of the limb movements of the actors.

Both aforementioned steps can be altered according to different factors such as the quality of the clips in the dataset, the desired processing time, and the expected output from the user.

3.2.3 Training

Our goal was to obtain a representation of a single instance for each action. In this regard, after computing GF-OF between two consecutive frames, KMeans was applied using 50 components $(K = 50)$ on sub-clips $[f, f + 2]$ consisting of three consecutive frames. A moderately larger number of frames per sub-clip may have also been employed for faster computation without having a considerable impact on classification results. Sub-clips in which little to no movement was present (e.g. transition from bending down to going back up) employed a lower value of K clusters, to prevent them from holding or being concentrated upon few optical flow points.

During the training process for one-shot learning, one video of each action was used (k videos were used for k-shot learning). Training videos in which the action was only executed once had all their KMeans clusters, which represent that single instance, saved. On the other hand, training videos in which more than one repetition of the action was completed, had the KMeans clusters of only one of those instances saved. This process was completed by setting a range of frames which contain only one instance for each action. The length f (number of frames) of each action occurrence was also stored and used in the testing process described in Section 3.2.4. Finally, the KMeans clusters of each action were represented by a Gaussian mixture model consisting of n Gaussian components.

Gaussian Components Experiments were conducted to determine the ideal number n of Gaussian components per mixture. The results illustrated in Fig. 3.3 demonstrate that the highest average classification accuracies were achieved when $n = 9$. This signifies that 9 Gaussian components are sufficient to accurately represent a fully executed action.

Additionally, Gaussian components of similar action instances may resemble each other in the

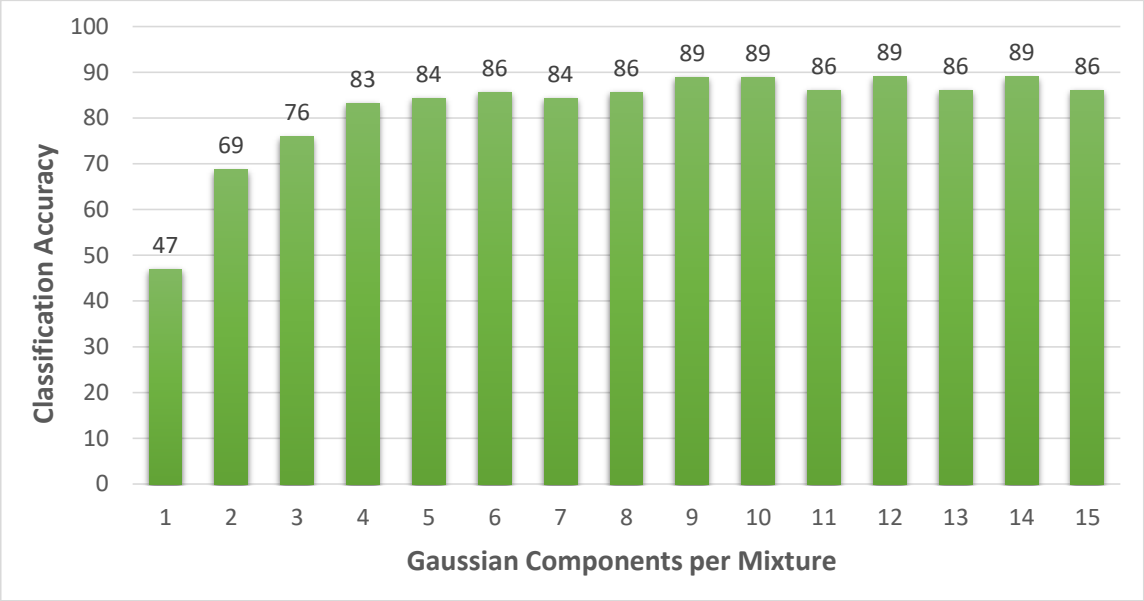

Figure 3.3: Classification accuracy with different numbers of Gaussian components n per mixture using the Weizmann dataset. The highest accuracies were achieved when at least 9 components were used $(n = 9)$.

(x, y, u, v) dimensions, however, they often do not occur in the same range on the temporal axis (t). To deal with this drawback, we employed a simple method in which the Gaussian mixture of each action instance was assumed to begin at $t = 0$ on the temporal axis. This was done by subtracting the lowest t value of all the KMeans points, which are within the range of frames of interest, from the t values of all the other points within that same range. This process was employed prior to each Gaussian mixture generation in both training and testing steps. For example, selecting the second "Wave2" instance displayed in Fig. 2.2 for training would involve the employment of this process on the KMeans points within the range of that instance so that they end up being within the frame range $t \in [0, 25]$ instead of $t \in [30, 55]$.

3.2.4 Testing

Once one, or k, representations of each action were carried out, the testing process went as follows: Each video in the dataset, except the ones used in training, went through a similarity measurement process in which an attempt to find similarities between the trained actions and the data of the test video was conducted. This was done by calculating the KL-divergence between the Gaussian mixture of each trained action and different Gaussian mixtures in the testing video. Those Gaussian mixtures were generated on different f frames long blocks. One of the assumptions we made was that all similar actions have instances executed over the same number of frames f. For instance, a trained "bending" action consisting of 50 total frames had an f value set to 50. Therefore, when an attempt was made to find similarities between this "bending" action and the action in the test video, 50 frames long blocks of KMeans points were represented as Gaussian mixtures. After each cluster was generated, KL-Divergence was applied to obtain a measure of similarity between the training and the generated testing probability distributions. Finally, after obtaining a measure of similarity using each trained action, the test video was classified by matching it to the trained video with which the KL-divergence value was the lowest. All classification accuracies demonstrated are averages of 5 runs.

3.2.5 One-Shot Learning

In some cases, two instances of a same action executed by two different actors have a significant resemblance between each other from a temporal perspective. An example of such case is demonstrated in Fig. 1.1, in which the actors "Daria" and "Denis" from the Weizmann dataset have very similar "Two-hands wave" actions. However, in general cases, a clear difference was noticed in the number of frames f required to represent one same action. This is due to the presence of variance in the execution time of an action from person to person. Due to this observation, we conducted some experiments, using one-shot learning, to check how our assumption regarding fixing the value of f according to the training data would affect the classification results. The experiments involved replacing f by $f + \Delta$ with $\Delta \in [1, 15]$, in which Δ represents a fixed number of additional

frames ranging from 1 to 15. The results showed only a slight fluctuation of $\pm 2\%$ in accuracy as Δ increased, confirming the validity of our assumption.

Seo and Milanfar [6]	75%
Yang [7]	80%
FSHMM [39]	81.5%
MAP+SHMM [10]	81.88%
MAP+SHMM (Relaxed) [10]	87.12%
Proposed	**89.4%**

Table 3.1: Classification accuracies for one-shot learning of proposed work and similar works using Weizmann dataset.

Seo and Milanfar [6]	65%
SHMM [10]	70.4%
FSHMM [39]	71.8%
Proposed	**73.1%**

Table 3.2: Classification accuracies for one-shot learning of proposed work and similar works using KTH dataset.

Experiments conducted using one-shot learning lead to an average classification accuracy of **89.4%** for the Weizmann dataset and **73.1%** for the KTH one. The accuracies of our work have been compared with methods from other works which were used in one-shot/k-shot learning settings on the same datasets. The results displayed in Tab. 3.1 and Tab. 3.2 show that our work has the highest accuracy compared to other works. The confusion matrix in Fig. 3.4, shows that when only one example per action is used during training in the Weizmann dataset, the main source of misclassification happens in the "skip" action, in which 50% of the test videos were wrongly classified, and received prediction labels of "side" or "walk" instead. In the case of the KTH dataset, the main source of misclassification was between the "jogging" and "running" actions, which are highly similar in nature. A solution we will be implementing in an attempt to fix such problem in future works is to automate the hyperparameters adjustment, as discussed in Section 3.3.

Tab. 3.3 demonstrates the KL-Divergence values between a set of trained actions using one-shot learning and the ones of actor "Daria" from the Weizmann Human Action dataset. The classification

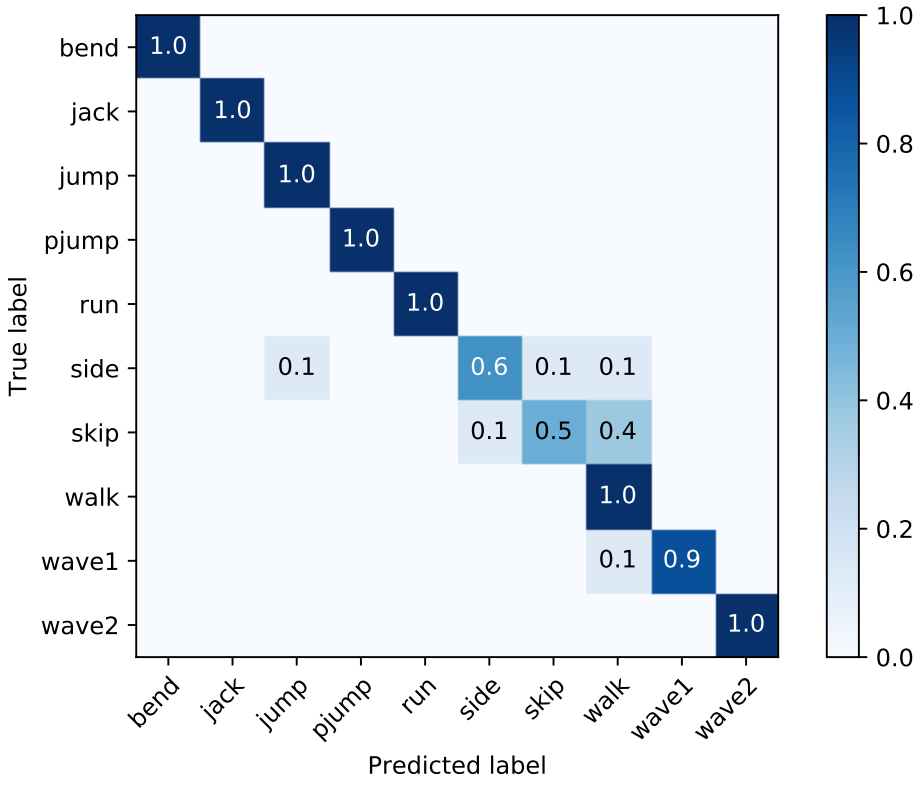

Figure 3.4: Normalized confusion matrix for the classification of 10 actions of Weizmann Human Action dataset using one-shot learning.

		bend	**jack**	**jump**	**pjump**	**run**	**side**	**skip**	**walk**	**wave1**	**wave2**
Test Actions	**bend**	**2.14**	31.32	10.51	28.39	6.85	6.81	5.92	7.68	7.07	9.30
	jack	3.77	**0.29**	3.64	1.35	2.90	3.09	4.20	4.15	4.00	3.02
	jump	5.81	31.55	**1.94**	23.08	2.82	3.50	10.91	13.05	4.77	24.36
	pjump	0.69	2.30	2.45	**0.69**	3.95	5.39	9.45	10.83	5.39	5.90
	run	2.59	25.40	1.75	18.35	**0.50**	2.09	5.29	6.28	2.66	14.90
	side	2.19	34.76	5.68	27.54	3.36	1.03	**0.86**	1.21	4.13	31.50
	skip	2.27	23.00	11.77	13.55	5.29	3.31	**0.82**	0.85	8.10	25.57
	walk	6.32	19.03	21.86	18.02	14.54	13.86	4.78	**4.15**	22.34	8.33
	wave1	3.99	6.57	6.35	7.22	5.09	5.25	5.98	7.04	**1.62**	4.71
	wave2	3.43	5.14	6.78	5.81	3.59	4.54	6.96	8.52	3.19	**1.74**

Table 3.3: Min. KL-Divergence values between training and testing actions using one-shot learning. All testing actions are executed by "Daria" from the Weizmann Human Action dataset. Values highlighted in green correspond to correct classifications, whereas the one highlighted in red is an example of misclassification, in which the "side" action of Daria had higher similarity with the trained "skip" action than the trained "side" one.

process was done by matching each action executed by "Daria" to the trained one with which the KL-Divergence value is the lowest. It is perceivable, that actions which share some similarities with

each other, have lower divergence values between each other compared to ones that do not share ample similitude.

3.2.6 k-Shot Learning

Following one-shot learning, each experiment involved incrementally training *one* additional example of each action prior to going through the classification process. Each additional action video used in training was removed from the test dataset. The results shown in Fig. 3.5 compare our classification accuracies using different values of k with methods from different works. The graph shows that using as little as one training example per action (k = 1), a classification accuracies of **73.1%** and **89.4%** were achieved for the KTH and Weizmann datasets, respectively, compared to 80% for Yang [7], and 30% for BoVW [32]. As the number of training examples k increases, the

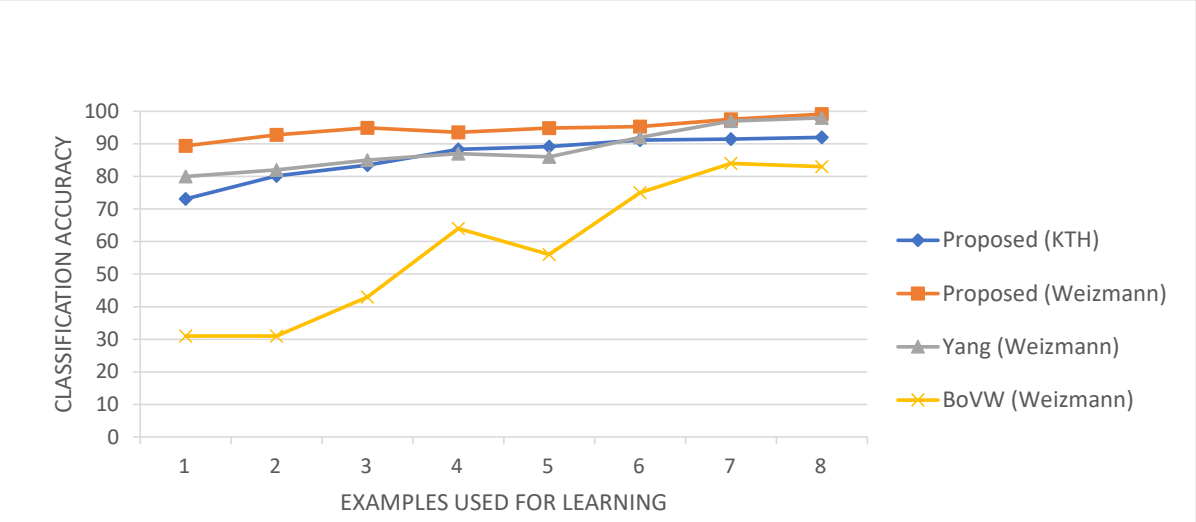

Figure 3.5: Classification accuracy comparison between proposed method and others.

3.3 Limitations and Possible Extensions

This novel occurrence-based representation method that we have presented has been proven to be robust, even when only one training example is used. The use of KL-divergence values as a measure of similarity may also be combined with threshold values and used to discard outliers in human action datasets (e.g. Datasets which do not solely contain videos of human actions). For instance, a

test video which has high KL-Divergence value with all trained actions would be labelled as an outlier. Regarding the time complexity of our method, since it is essentially based off an instance-based learning approach, it holds the same advantages and drawbacks as other lazy learning methods. The training phase is considerably efficient but is coupled with a slow evaluation phase. Needless to say, the computation time highly depends on the nature of the application in which the method is being used and comes at the expense of some classification accuracy. For instance, a one-shot learning setting combined with minimal frame rescaling, high value of k (number of frames per sub-clip prior to application of KMeans) and low number of blocks in the KL-Divergence process, leads to lower computation time than a few-shot learning setting using opposite settings to maximize classification accuracy. Moreover, although the classification accuracies have been effective, there is room for improvement in the following sections:

Hyperparameters Different hyperparameters such as the GF-OF threshold t, the number of K-means components K and the number of Gaussian components n used per mixture, were set after conducting experiments to find their ideal values. Our next objectives include the automation of the adjustment of those hyperparameters, by designing both a feature extraction and a training model which can automatically adjust the hyperparameters according to the input data. For example, the training model would set the ideal number of Gaussian components n to represent a specific action and proceed through the "similarity measurement" process using that same number of components to find actions similar to the trained ones.

Unsupervised Action Recognition The training model that we have implemented was done in a supervised manner. Our next goal is to create a completely unsupervised human action recognition model which is capable of automatically finding action instances/repetitions within a same video and use one of those repetitions in the classification process. Additionally, we plan on utilizing the effectiveness of frameworks such as GAN and R-CNN to expand the flexibility of our work and enable its application in a wider range of datasets, including ones with multiple actors per video.

Chapter 4

Conclusion

In this thesis, an instance-based learning approach for human actions classification was proposed. The method employed Gunnar Farneback Optical Flow and K-means clustering to obtain accurate spatiotemporal features of an action, represented those features by a Gaussian mixture model, classified test videos using KL-divergence between two Gaussian mixtures and matched ones with the lowest divergence values. The conducted experiments involved validating an assumption made regarding the temporal perspective of each action instance, pinpointing the ideal number of Gaussian components to use per Gaussian mixture and running experiments using one-shot and k-shot learning. As displayed in the results section, the application of KL-Divergence as a similarity measure is demonstrated. Its computed values validate the usefulness of using such measure in our framework to not only achieve action classification, but to also give us a sense of how similar the actions in the dataset are. Similar actions exhibit low divergence values between each other, whereas dissimilar ones exhibit considerably higher values. The meaningful representation of human action instances, combined with the instance-based learning approach used, demonstrated that using as little as one training video per action yielded considerably high accuracies in comparison with state-of-the-art works. The flexibility of our work enables its application to other fields such as detection of outliers in datasets according to their KL-Divergence values (or similarity) with respect to the rest of the dataset. Additionally, various extensions could also be used on the proposed framework depending on the application, such as automating the hyperparameter tuning process

used in the KMeans and GMM processes according to the input dataset to achieve higher classification accuracies while optimizing overall performance. Future works will be devoted to applying the proposed framework to anomaly detection in crowded areas by integrating it with different topic modeling-based approaches.

Bibliography

[1] D. Avola, M. Bernardi, G.L. Foresti (2019) Fusing depth and colour information for human action recognition. Multimed Tools Appl 78:5919–5939

[2] I. Kapsouras, N. Nikolaidis (2019) Action recognition by fusing depth video and skeletal data information. Multimed Tools Appl 78:1971–1998

[3] F. Najar, S. Bourouis, N. Bouguila, S. Belghith (2019) Unsupervised learning of finite full covariance multivariate generalized Gaussian mixture models for human activity recognition. Multimed Tools Appl 78:18669–18691

[4] L. Fei-Fei and R. Fergus and P. Perona (2006) One-shot learning of object categories. IEEE Trans. on Pattern Analysis and Machine Intelligence vol. 28(4):594–611

[5] S. Ji and W. Xu and M. Yang and K. Yu (2013) 3D convolutional neural networks for human action recognition. IEEE Trans. on Pattern Analysis and Machine Intelligence 35(1):221–231

[6] H. J. Seo and P. Milanfar (2011) Action recognition from one example. IEEE Trans. on Pattern Analysis and Machine Intelligence 33(5):867–882

[7] Y. Yang and I. Saleemi and M. Shah (2012) Discovering motion primitives for unsupervised grouping and one-shot learning of human actions, gestures, and expressions. IEEE Trans. on Pattern Analysis and Machine Intelligence 35(7):1635–1648

[8] A. F. Bobick and J. W. Davis (2001) The recognition of human movement using temporal templates. IEEE Trans. on Pattern Analysis and Machine Intelligence 23(3):257–267

[9] E. Shechtman and L. Gorelick and M. Blank and M. Irani and R. Basri (2007) Actions as space-time shapes. IEEE Trans. on Pattern Analysis and Machine Intelligence 29(12):2247–2253

[10] M. Rodriguez and C. Orrite and C. Medrano and D. Makris (2016) Oneshot learning of human activity with an map adapted gmm and simplex-hmm. IEEE Trans. on Cybernetics 47(7):1769–1780

[11] X. Lu, C. Ma, B. Ni and X. Yang (2019) Adaptive Region Proposal with Channel Regularization for Robust Object Tracking. IEEE Trans. on Circuits and Systems for Video Technology.

[12] Y. Fei, L. Li, X. Lin et al (2019) A robust and fixed-time zeroing neural dynamics for computing time-variant nonlinear equation using a novel nonlinear activation function. Neurocomputing 350:108–116

[13] Y. Fei, L. Li, H. Binyong et al (2019) Analysis and FPGA realization of a novel 5D hyperchaotic four-wing memristive system, active control synchronization, and secure communication application. Complexity

[14] Z. Jianming, X. Zhipeng, S. Juan et al (2020) A cascaded R-CNN with multiscale attention and imbalanced samples for traffic sign detection. IEEE Access 8:29742–29754

[15] Z. Jianming, L. Chaoquan, W. Jin and et al (2020) Training Convolutional Neural Networks with Multi-Size Images and Triplet Loss for Remote Sensing Scene Classification. Sensors 20(4):1188

[16] C. Yuantao, W. Jin, C. Xi et al (2019) Image super-resolution algorithm based on dual-channel convolutional neural networks. Applied Sciences 9(11):2316

[17] L. Yuanjing, Q. Jiaohua, X. Xuyu et al (2020) Coverless real-time image information hiding based on image block matching and dense convolutional network. Journal of Real-Time Image Processing 17(1):125–135

[18] D. Lin, X. Weihong, C. Yuantao (2020) Density Peaks Clustering by Zero-Pointed Samples of Regional Group Borders. Computational Intelligence and Neuroscience

[19] Z. Luoyu, Z. Tao, T. Yumeng, H. Hu (2020) Fraction-Order Total Variation Image Blind Restoration Based on Self-Similarity Features. IEEE Access 8:30436–30444

[20] Y. Chen and W. Xu and J. Zuo and K. Yang (2019) The fire recognition algorithm using dynamic feature fusion and IV-SVM classifier. Cluster Computing 22(3):7665–7675

[21] Y. Chen and J. Tao and L. Liu and J. Xiong and R. Xia and J. Xie and Q. Zhang and K. Yang (2020) Research of improving semantic image segmentation based on a feature fusion mode. Journal of ambient intelligence and humanized computing

[22] C. Yuantao, W. Jin, L. Songjie et al (2019) Multiscale fast correlation filtering tracking algorithm based on a feature fusion model. Concurrency and Computation: Practice and Experience

[23] C. Yuantao, T. Jiajun, Z. Qian et al (2020) Saliency Detection via the Improved Hierarchical Principal Component Analysis Method. Wireless Communications and Mobile Computing

[24] C. Yuantao, L. Linwu, T. Jiajun et al (2020) The improved image inpainting algorithm via encoder and similarity constraint. The Visual Computer

[25] C. Yuantao, W. Jin, X. Runlong et al (2019) The visual object tracking algorithm research based on adaptive combination kernel. Journal of Ambient Intelligence and Humanized Computing 10(12):4855–4867

[26] C. Yuantao, X. Jie, X. Weihong, Z. Jingwen (2019) A novel online incremental and decremental learning algorithm based on variable support vector machine. Cluster Computing 22(3):7435–7445

[27] C. Yuantao, W. Jin, C. Xi et al (2019) Single-image super-resolution algorithm based on structural self-similarity and deformation block features. IEEE Access 7:58791–58801

[28] H. B. Zhang, Y. X. Zhang, B. Zhong et al (2019) A comprehensive survey of vision-based human action recognition methods. Sensors 19(1005):1–20

[29] A. Yilmaz and M. Shah, Actions sketch: a novel action representation (2005) IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) 1:984–989

[30] E.G. Miller and N.E. Matsakis and P.A. Viola (2000) Learning from one example through shared densities on transforms. Proceedings IEEE Conference on Computer Vision and Pattern Recognition 1:464–471

[31] T. Darrell and A. Pentland (1993) Space-time gestures. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition pp. 335–340

[32] H. Wang and A. Klaser and C. Schmid and C. Liu (2011) Action recognition by dense trajectories. CVPR 2011 pp. 3169–3176

[33] M. Chao, Y. Xiaokang, Z. Chongyang, Y. Ming-Hsuan (2015) Long-term correlation tracking. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp. 5388–5396

[34] L. Xiankai, W. Wenguan, M. Chao et al (2019) See more, know more: Unsupervised video object segmentation with co-attention siamese networks. 2019 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp. 3623–3632

[35] L. Xiankai, W. Wenguan, S. Jianbing et al (2020) Learning Video Object Segmentation from Unlabeled Videos. 2020 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp. 8960–8970

[36] C. Feichtenhofer and A. Pinz and A. Zisserman (2016) Convolutional two-stream network fusion for video action recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp. 1933-1941

[37] M. Blank and L. Gorelick and E. Shechtman et al (2005) Actions as Space-Time Shapes. Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1 2:1395–1402

[38] Y. Ke, R. Sukthankar and M. Hebert (2005) Efficient visual event detection using volumetric features. Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1 1:166–173

[39] M. Rodriguez and C. Orrite and C. Medrano and D. Makris (2017) Fast simplex-HMM for one-shot learning activity recognition 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) pp. 1259–1266

[40] G. Farneback (2000) Fast and accurate motion estimation using orientation tensors and parametric motion models. Proceedings 15th International Conference on Pattern Recognition (ICPR-2000) 1:135–139

[41] G. Farneback (2001) Very high accuracy velocity estimation using orientation tensors, parametric motion, and simultaneous segmentation of the motion field. Proceedings Eighth IEEE International Conference on Computer Vision (ICCV 2001) 1:171–177

[42] A. Yilmaz and M. Shah (2005) Recognizing human actions in videos acquired by uncalibrated moving cameras. ICCV

[43] J. Goldberger and S. Gordon and H. Greenspan (2003) An efficient image similarity measure based on approximations of KL-divergence between two gaussian mixtures. Proceedings Ninth IEEE International Conference on Computer Vision 1:487–493

[44] J. R. Hershey and P. A. Olsen (2007) Approximating the kullback leibler divergence between gaussian mixture models. 2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07 4:IV-317-IV-320

[45] Y. Kong and Y. Fu (2018) Human Action Recognition and Prediction: A Survey. arXiv:1806.11230v2 [cs.CV] pp. 1–20

[46] H. João F, C. Rui, M. Pedro, B. Jorge (2014) High-speed tracking with kernelized correlation filters. IEEE transactions on pattern analysis and machine intelligence 37(3):583–596

[47] S. Christian, L. Ivan, C. Barbara (2004) Recognizing human actions: A local SVM approach. Proceedings - International Conference on Pattern Recognition 3:32–36