MDL-based Development of Ensembles with Active Learning over Evolving Data Streams

Samaneh Khoshrou
Eindhoven University of Technology
Eindhoven, Netherlands
s.khoshrou@tue.nl

Mykola Pechenizkiy
Eindhoven University of Technology
Eindhoven, Netherlands
m.pechenizkiy@tue.nl

ABSTRACT

Learning from multiple unbounded time-series has received less attention despite the key applications (such as video analysis) generating this data. Inspired by never-ending approaches, this paper presents an algorithm to continuously learn from multiple unregulated time-series, in a framework based on ensembles of GMM-UBM (Universal Background Models). The Minimum Description Length (MDL) method, as a powerful inductive inference, is exploited to predict the quality of current knowledge on arrival observations in an unsupervised manner in order to control the complexity while maintaining the accuracy of the framework in such evolving environment. Extensive experiments demonstrate the advantages of the proposed framework in terms of accuracy and complexity over several baseline approaches on multiple datasets.

CCS CONCEPTS

• **Data science** **Data stream mining**; • **Big data** *Long-term learning*; • **Visual data** Intelligent surveillance;

KEYWORDS

Long-term learning, Ensembles, Mdl-based development

ACM Reference Format:
Samaneh Khoshrou and Mykola Pechenizkiy. 2018. MDL-based Development of Ensembles with Active Learning over Evolving Data Streams. In *MiLeTS '18, August 2018, London, United Kingdom.* ACM, New York, NY, USA, Article 4, 8 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Time series are present in many key real world problem such as audio and video processing. It is expected that using time-series learning techniques leads to effective and hands-on solutions for such scenarios. In this paper, one of the central problems related to video analysis is approached from a time-series perspective.

Networks of video cameras are commonly employed to monitor large areas for a variety of applications. A central issue in such networks is the tracking and recognition of individuals of interest across multiple cameras. These individuals must be recognized when leaving the Field of View (FoV) of one camera and re-identified when entering the FoV of another camera. In such environments, the underlying distribution of data changes over time - often referred to as *concept drift* [14] – either due to intrinsic changes (pose change, movement, etc.), or extrinsic changes (lighting condition, dynamic background, complex object background, changes in camera angle, etc.). Thus, models need to be continually updated to represent the latest concepts. Moreover, when new objects enter the scene - referred to as *class evolution* – new models need to be trained for the novel classes. Additionally, it is likely to have multiple streams, recorded at different starting points with various lengths, for the same Region of Interest (RoI) of individuals, since the objects move and cross in the FoV of multiple cameras (see Figure 1a). The problem gets further complex when the system is faced with *unbounded streams* of data [1]. It is desirable, the surveillance system tracks that person across all cameras whose FoV overlap the person's path over an unlimited time frame. Thus, a suitable outcome for this system could be a time-line graph assigning streams from each camera to an identity for the indicated presence period, as illustrated in Figure 1b. Learning in such scenario can be characterized as follows:

Let υ be a set of unregulated time-series υ_i. Streams are potentially with concept drift as well as concept evolution. Each observation x within each stream is in a d-dimensional space, $x \in R^d$. Recording is not limited to a bounded period. An effective and appropriate one-pass algorithm to fit in our scenario is required to:

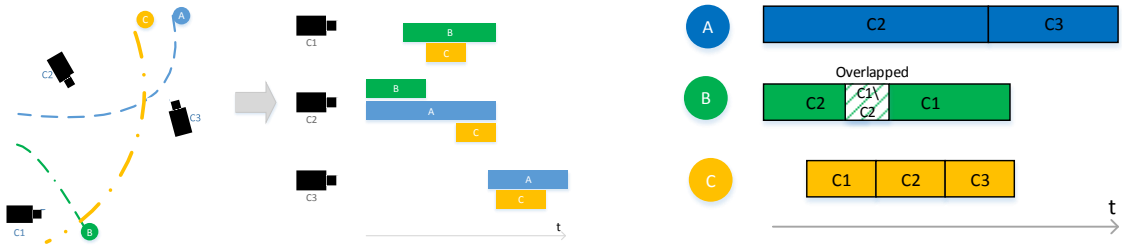
- learn from multiple unregulated streams;
- handle multiple high-dimensional data streams;
- handle concept drift;
- accommodate new evolving classes;
- deal with massive unlabelled data;
- be of limited space and time complexity.

Main Contributions: We propose a strategy for persistent learning of multiple time-series over an unbounded time frame. Inspired by never-ending learning approaches, we employ active (detect & re-act) techniques to control the complexity of the most popular group of passive approaches, ensemble based models [11], in a time evolving environment. The active approach is based on an information theoretic criterion that triggers an adaptation with respect to the models' quality by updating or building a classifier. The key insight is that the "good" models can describe incoming observations as efficiently as possible, thus, we adopt a Minimum Description Length (MDL) criterion to predict how well the current knowledge can represent new observations in an unsupervised nature.

The rest of the paper is organized as follows. In Sections 2 and 3 we review the employment of learning methods for evolving environments and some background on the NEVIL.ubm approach respectively. Section 4 provides an overview of the leaning framework. In Section 5 we discuss the experimental methodology. In Section 6, we

MiLeTS '18, August 2018, London, United Kingdom

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn



(a) Typical surveillance scenario

(b) An example of time-line graph

Figure 1: A surveillance scenario including three persons A, B, and C, moving in the scene, crossing the FoV of 3 cameras: c_1, c_2, and c_3 .

experimentally investigate the effectiveness of proposed long-term strategy on several real-world videos.

2 RELATED WORK

In this paper, we look at the problem as learning from multiple data streams in wild environments, that views segments of a stream as a unique element to classify, thus single stream mining methods cannot be employed. With a few exceptions [21, 22], most of the methods proposed for parallel stream mining [7] require equal-length streams coming from a fixed number of sources. Thus, they would fail to leverage information from time-varying video tracks. Despite the success of NEVIL.gmm and NEVIL.ubm to mine multiple unregulated streams, long-term learning is still a major issue.

Never-ending learning systems have been one of the latest interest in the field of learning as they are able to learn many concepts "in a cumulative nature". The Never-Ending Language Learning (NELL) [5] research project has been the inspiration of numerous researches to address the never-ending learning problem [2, 8, 16, 18]. Obviously, the techniques used by research works are informed by different assumptions in respect with the applications and goals. With a few exceptions [20, 28], most of the never-ending literature has focused on coverage of knowledge, while our approach tries to cover knowledge and accuracy as well as efficiency.

Learning in non stationary environment requires evolving approaches that can adapt to accommodate the changes accordingly. The adaptation problem has been addressed by either active or passive approaches. The active approach is designed to detect concept drift in order to trigger an adaptation [14], whereas the passive one continuously update the knowledge every time new data is received. While active approaches are more effective in online settings with abrupt drift, passive approaches are better suited for batch learning in settings with gradual drift and recurrent concepts [11]. Ensemble based approaches are the most popular group of passive methods due to higher accuracy, flexibility and stability to handle concept drift as well as class evolution [12]. A classic approach to track changes is to train new classifier(s) as new data arrives and to keep all the classifiers [9]. Accumulating large number of classifiers imposes serious costs (i.e. acute storage space and long prediction time) to the system. Although the cost seems negligible with relatively simple research datasets, they may become highly critical for real-word data. In fact, these approaches can easily generate thousands of classifiers under a time-evolving environment. Additionally, it is not always true that the bigger ensemble, the better it is [29]. Some research works tried to address this problem using a time-weighting strategy [12, 22], in which decisions made by models inside ensembles are combined in respect to time. However, by giving higher weights to the decision made by more recent models, the older ones are forgotten in time, still a substantial number of models are kept in the framework.

INEVIL was proposed in [20] for long term monitoring of objects by detecting the deviations in either feature distribution or learner. In this work, the problem is seen from a different perspective. We propose an unsupervised criterion to inspect whether the current knowledge is able to represent new observations "well enough"?

3 BACKGROUND ON THE NEVIL.UBM APPROACH

In this section, the Never Ending Visual Information Learning with UBM (NEVIL.ubm) framework is briefly presented. NEVIL.ubm [22] is designed for learning from multiple un-regulated streams in a non-stationary environment where no labelled data is available at the first place but the learning algorithm is able to interactively query the user to label the desired outputs at carefully chosen data points.

The system receives multiple visual streams, generated by a typical tracking algorithm, which analyses sequential video frames and tracks RoIs over time. For each RoI the features corresponding to some pre-selected object representation (e.g. bag of words) are extracted ($\upsilon l\ l = 1, ..., B$). A batch $\upsilon_t^{m_t}$ is a temporal sequence of frames $\upsilon_{t,f}^{m_t}$, where f runs over 1 to the batch size B. Initially, the composite model is initialized to yield the same probability to every class (uniform prior). When the features of batches of RoIs $\upsilon_{t,f}^{m_i}$ in time slot t become available, the framework starts computing the scores $\mathscr{S}(\upsilon_t^{m_i}|C_k, H_{t-1})$ for each batch $\upsilon_t^{m_i}$ in the time slot. The scores are obtained from the likelihood ratio test of the batch data obtained by the individual class model C_k and the UBM.

The composite model H_t is an ensemble of Micro-classifiers ensembles ($MCE_t^j, j = 1, ..., k$). Each MCE_t^j includes classifiers that are incrementally trained (with no access to previous data) on incoming batches of j_{th} class at t, h_t^j. The individual models h_t^j are combined using a weighted majority voting, where the weights are dynamically updated with respect to the classifiers' time of design.

The prediction output by the composite model MCE_t^j for a given ROI ($\upsilon_{t,f}^{m_i}$) is

$$p\left(C_k|\upsilon_{t,f}^{m_i}, MCE_t^j\right) = \Sigma_{\ell=1}^{t} W_\ell^t h_\ell\left(C_K|\upsilon_{t,f}^{m_i}\right) \quad (1)$$

where h_ℓ^j. is the classifier trained from batches of j_{th} at TS ℓ, W_ℓ^t is the weight assigned to classifier ℓ, adjusted for time t. The weights are updated and normalised at each time slot and chosen to give more credit to more recent knowledge. After combining the decisions of classifiers inside every MC-ensemble, the ensemble will assign a batch to the label of MC-ensemble with highest score ($\mathscr{S}(\upsilon_t^{m_i}|C_k, H_{t-1})$.

Such on-line learning may suffer if labelling errors accumulate, which is inevitable. To help mitigate this issue, the system is designed to interact wisely with a human. Once $\mathscr{S}(\upsilon_t^{m_i}|C_k, H_{t-1})$ is obtained, a batch confidence level (BCL) is estimated. In NEVIL.ubm framework, if the scores associated to all observed classes are significantly low (below a predetermined threshold), it is very likely that this class has not been observed before and it is considered novel and a new label (y) is automatically assigned to this batch. Having decided that the batch data belongs to an existing class, one needs to decide if the automatic prediction is reliable (the reliability test is positive) and accepted or rather a manual labelling needs to be requested. If BCL is high enough (above a predefined threshold), the predicted label

$$\hat{y} = \arg\max_{C_k} \mathscr{S}\left(\upsilon_t^{m_i}|C_k, H_{t-1}\right) \quad (2)$$

is accepted as correct; otherwise the user is requested to label (y) the data batch.

The choice of Gaussian Mixture Models (GMM) to model feature distributions in biometric data is motivated by extensive research of related areas. From the most common interpretations, GMMs are seen as capable of representing broad "hidden" classes, reflective of the unique structural arrangements observed in the analysed biometric traits [24]. Besides this assumption, Gaussian mixtures display both the robustness of parametric unimodal Gaussian density estimates, as well as the ability of non-parametric models to fit non-Gaussian data [23]. This duality, alongside the fact that GMM have the noteworthy strength of generating smooth parametric multimodal densities, confers such models a strong advantage as generative model of choice. To train the Universal Background Model a large amount of un-labeled data, is used, so as to cover a wide range of possibilities in the individual search space [27]. The training process of the UBM is simply performed by fitting a k-mixture GMM to the set of feature vectors extracted from all the "impostors". In this framework, the UBM is trained offline, before the deployment of the system. It is designed from a pool of streams of disjoint individuals that is representative of the complete set of potentially observable 'objects'.

At each time slot, the batches predicted to belong to the same class are used to generate the class model by *tuning the UBM parameters* in a maximum *a posteriori* (MAP) sense. The adaptation process consists in two main estimation steps. First, for each component of the UBM, a set of sufficient statistics is computed from a set of M class specific feature vectors. Each UBM component is then adapted using the newly computed sufficient statistics, and considering diagonal covariance matrices. Note that the UBM is trained offline, before the deployment of the system. It is designed from a large pool of streams aimed to be representative of the complete set of potentially observable 'objects'.

4 LONG-TERM LEARNING OF A CONCEPT

Long-term learning has been mostly addressed with two strategies in the literature; one trains a new classifier as new data arrives [12, 22], which obviously impose serious cost to the system, on the other side, a less expensive method incrementally updates a learner with new observations [10], however it may fail to detect recurrent drift after awhile. Between two extremes, we proposed a method to actively update a passive learning composite in an unsupervised manner [20].

The first step is to inspect whether at least one of the classifiers inside a micro-ensemble is able to represent new batches "well" or a new model needs to be added to the ensemble.

The answer lies within *model selection techniques*, which stands out as one of the most important problems of inductive inference. The Minimum Description Length (MDL) -based model selection is a well-established method in machine learning [15, 25]. In our work we adopt MDL principle to develop a strategy for controlling complexity of an ensemble in active learning over evolving multi-dimensional data streams.

The description length of a fitted model is the sum of two parts. The first part of the description length represents the complexity of the model $L(h)$. This part encodes the parameters of the model itself; it grows as the model becomes more complex. The second part of the description length represents the fit of the model to the data $L_h(x_n)$; as the model fits better, this term shrinks. The best model is the one which minimizes the total codelength of the two-stage code:

$$Cost\ \left(h, \upsilon\right) = argmin\ \left(L\left(h\right) + L_h\left(\upsilon_t\right)\right) \quad (3)$$

Note that for the sake of simplicity, we omit the indices. In [17], the success of an ensemble of M models compression is assessed by:

$$Cost_{\mathscr{H}} = \Sigma Cost\ \left(h, \upsilon\right) \quad (4)$$

Once a new batch of RoIs is received, the framework assigns a label (m). Then the cost of the ensemble is predicted in two ways:i)if a new model adds to the ensemble $cost_add$. ii) if the most representative model inside the current ensemble is updated $cost_update$. If $cost_add < cost_update$, a new model is added, otherwise the framework updates the most representative model. However, in our framework, GMM-UBM are utilized as the base learners, so the complexity of individual models do not change over time and the data code length is the most effective measure. Since, adding a new model increases the complexity of the ensemble, it is preferable to update the model instead of adding a new member to the ensemble. This can be interpreted as if at least one of the models inside ensemble is "good" to represent the new observations.

Once a new batch of RoIs is received, the framework assigns a label (let assume, m). Then the best predictive model inside MCE_m, that yields the shortest code length with new observations, is identified. If the model is "good", the model requires reasonably short codes to describe new observations below a predefined threshold (T'). If so, the framework will update the best model in MCE_m with the most recent data (h_t^m). The method is detailed in Section 4.2. Otherwise, none of the models is not able to describe newly captured data due to abrupt drift, the framework trains a new model and stores into MCE_m. The algorithm is detailed in Alg. 1.

Algorithm 1 Long-Term Learning

procedure
Input: $\mathscr{H}^m_{t-1}, \upsilon^m_t, \forall m = 1, ..., K$
Model Quality Assessment (Sec.4.1)
$d^m_t = d\upsilon^m_t, h^m_j, \forall j = 1, ..., K$
Closest models
$\exists k \in 1, ..., K, d^t_k < d^t_j, \forall j \neq k$
if $d^t_m > T'$ **then**
Adding criterion
$h^m_t \leftarrow \upsilon^m_t$
$\mathscr{H}^m_t = h^m_t, \mathscr{H}^m_{t-1}$
else
Updating a concept (Sec. 4.2)
$hl^m_t = \boldsymbol{update}\,\upsilon^m_t, h^m_{t-1}$
$\mathscr{H}^m_t = hl^m_t, \mathscr{H}^m_{t-1}$

4.1 Model Quality Assessment

We propose a simple yet intuitive model quality assessment criterion based on MDL principle which yields a particularly simple way to evaluate how well a model will encode and describe a set of new observations. The rationale behind MDL criterion is: if you can build a short code for your data, this means that you have a good data generation model [19]. Inspired by [13], the minimum length between observation of the batch predicted to belongs to class j at t and the model ($h^{m_i}_j x = \sum_{i=1}^{c} \alpha_i h_i x$) inside MCE_j can be obtained as:

$$dh^m_j, \upsilon^m_t = -\log pC_k|\upsilon^m_{t,f}, h^m_j + \frac{c}{2}\log\frac{B}{12} + \frac{cN+1}{2} + \frac{N}{2}\sum_{i=1}^{c}\log\frac{B\alpha_i}{12} \tag{5}$$

where, $pC_k|\upsilon^m_{t,f}, h^m_j$, c, and B are code-length of the frames inside the batch, the number of model parameters, and the number of RoIs, respectively. N is a constant that grows quadratically with the dimension d of the data and for a case of free covariance matrix equals to $d + \frac{dd+1}{2}$.

4.2 Updating a learner with new observations

Once gradual drift is observed, the data from the batches predicted to belong from the same class is used to generate the class model by *tuning of the (h^m_{t-1}) parameters*, in a maximum *a posteriori* (MAP) sense. The rationale behind this method is basically similar to updating the individual models for UBM. The adaptation process consists of two main estimation steps. First, for each component of the hl^m_t, a set of sufficient statistics is computed from a set of B class specific feature vectors, $\upsilon^{*m}_t = \{\mathbf{x}_1, ..., \mathbf{x}_B\}$ computed from the batch data:

$$n_i = \Sigma_{b=1}^{B} p\left(i|\mathbf{x}_b\right) \tag{6}$$

$$E_i\left(\mathbf{x}\right) = \frac{1}{n_i}\Sigma_{b=1}^{B} p\left(i|\mathbf{x}_b\right)\mathbf{x}_b \tag{7}$$

$$E_i\left(\mathbf{x}\mathbf{x}^t\right) = \frac{1}{n_i}\Sigma_{b=1}^{B} p\left(i|\mathbf{x}_b\right)\mathbf{x}_b\mathbf{x}^t_b \tag{8}$$

where $p(i|\mathbf{x}_b)$ represents the probabilistic alignment of $\mathbf{x}_b$ into each h^m_{t-1} component. Each h^m_{t-1} component is then adapted using the newly computed sufficient statistics, and considering diagonal covariance matrices. The update process can be formally expressed as:

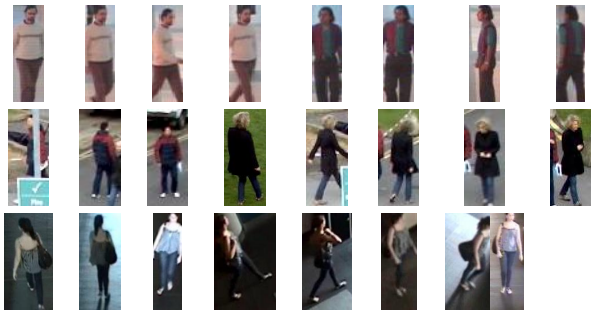


Figure 2: An example of diversity in appearance

$$\hat{w}_i = \left[\alpha_i n_i B + \left(1-\alpha_i\right)w_i\right]\xi \tag{9}$$

$$\hat{\mu}_i = \alpha_i E_i\left(\mathbf{x}\right) + \left(1-\alpha_i\right)\mu_i \tag{10}$$

$$\hat{\Sigma}_i = \alpha_i E_i\left(\mathbf{x}\mathbf{x}^t\right) + \left(1-\alpha_i\right)\left(\sigma_i\sigma_i^{\,t} + \mu_i\mu_i^{\,t}\right) - \hat{\mu}_i\hat{\mu}_i^{\,t} \tag{11}$$

$$\sigma_i = \mathrm{diag}\left(\Sigma_i\right) \tag{12}$$

where $\{w_i, \mu_i, \sigma_i\}$ are the original h^m_{t-1} parameters and $\{\hat{w}_i, \hat{\mu}_i, \hat{\sigma}_i\}$ represent their adaptation to the specific class. To assure that $_i w_i = 1$ a weighting parameter ξ is introduced. The α parameter is a data-dependent adaptation coefficient. Formally it can be defined as:

$$\alpha_i = \frac{n_i}{r+n_i} \tag{13}$$

The relevance factor r weights the relative importance of the original values and the new sufficient statistics.

5 EXPERIMENTAL METHODOLOGY

5.1 Datasets

In order to explore the properties of the proposed framework, we evaluated it on multiple datasets covering various possible scenarios in a multi-camera surveillance system. Experiments were conducted on public indoor (CAVIAR) and outdoor (PETS) datasets. Seven scenarios of CAVIAR (*OneLeave ShopReenter1, Enter ExitCrossingPaths1, OneShopOneWait1, OneStop Enter2, WalkBy Shop1front*) as well as two views of scenario S2.L1 of PETS2009 have been applied in our experiments. SAIVT-Softbio is the only dataset that simultaneously meets all the requirements for a full open-world task: Multi-shot data and multiple cameras with camera-transition uncertainty [3]. This dataset consists of 152 subjects travelling in a building environment through up to eight camera views, appearing from various angles and in varying illumination conditions reflecting real-world conditions (see Figure 2). To evaluate the system, each dataset is divided into 3 disjoint subsets (different individuals). The first subset is used to train UBMs. The second set is used to calibrate all the threshold T'. The final portion is used to evaluate the performance. To extract the RoIs, we employed an automatic tracking approach to track objects in the scene and generate streams of bounding boxes, which define the tracked objects' positions. As the tracking method fails to perfectly track the targets, a stream may include RoIs of distinct objects.

Dataset	No. of Streams	Range	No. Classes	Imbalance Degree	No. of Cameras	Setting
OneLeaveShopReenter1	3	85 – 160	2	0.28	2	Overlapped
OneLeaveShopReenter2	3	63 – 347	2	0.11	2	Overlapped
WalkByShop1front	6	40 – 225	4	0.22	2	Overlapped
EnterExitCrossingPaths1	6	34 – 216	4	0.23	2	Overlapped
OneStopEnter2	7	51 – 657	4	0.19	2	Overlapped
OneShopOneWait1	10	36 – 605	4	0.25	2	Overlapped
OneStopMoveEnter1	42	10 – 555	14	0.14	2	Overlapped
PETS2009	19	85 – 576	10	0.13	2	Overlapped
SAIVT-SOFTBIO	240	21 – 211	152	0.12	8	Overlpped, Nonoverlpped

Table 1: The datasets characteristics. Imbalance degree is defined by the ratio of sample size of minority class to that of the majority ones ; Range is defined by the length of shortest and longest streams in a given dataset, respectively.

5.2 RoI Representation

Our reference image descriptor is an improved version of FV, since the FV was found to serve as the most effective encoding technique for pooling approaches in recent studies [6]. Given an image (RoI), the IFV υ is obtained by extracting a dense collection of patches and corresponding local image features (herein, SIFT) from the image at multiple scales. To avoid the curse of dimensionality, Principle Component Analysis (PCA) is applied to the full set of features as a pre-processing step. The number of features in each stream is reduced to 200 dimensions.

5.3 Baseline Methods

The work closest in spirit to this work is [16], that proposed a never-ending framework for one dimensional real value time series. Since, we deal with multiple high-dimensional data streams, the framework is not applicable in our scenario two baseline approaches: 1) Ensemble Classifier Model (here, NEVIL.UBM), that adds a new member to the ensemble as new data arrives. 2) Incremental methods (single classifier models): at the other side of extreme these methods perform a continuous adaptation of the model, once new observations received.

5.4 Confidence Measure

Various criteria have been introduced as uncertainty measures to invoke the teachers in an interactive scenario [26]. *Most confident measure (MC):* Perhaps the simplest and most commonly used criterion relies on the probability of the most confident class, defining the confidence level as $\max_{C_k} \mathscr{S}\left(C_k|\upsilon_t^{m_i}, H_{t-1}\right)$.

5.5 Evaluation Criteria

Active learning aims to achieve high accuracy using as little annotation effort as possible. Thus, a trade-off between accuracy and proportion of labelled data can be considered as one of the most informative measures.

Accuracy. In a classical classification problem the disparity between real and predicted labels explains how accurately the system works. However, in our scenario the labels do not carry any semantic meaning. The same person should have the same label in different batches, whichever the label. As such, when evaluating the performance of our framework we are just comparing the partition of the set of batches as defined by the reference labelling with the partition obtained by the framework. Adopting a generic partition-distance method for assessing set partitions, which is initially proposed for spatial segmentations of images assessment [4], the accuracy is formulated as:

$$Accuracy = \frac{N - Cost}{N} \tag{14}$$

where N denotes the total number of batches, and *Cost* refers to the cost, yielded by the assignment problem.

Annotation. Assume MLB and TB denote the manually labelled batches and all the batches available during a period (includes one or more time slots), respectively. The *Annotation Effort* is formulated as:

$$Annotation\ effort = \frac{\#MLB}{\#TB} \tag{15}$$

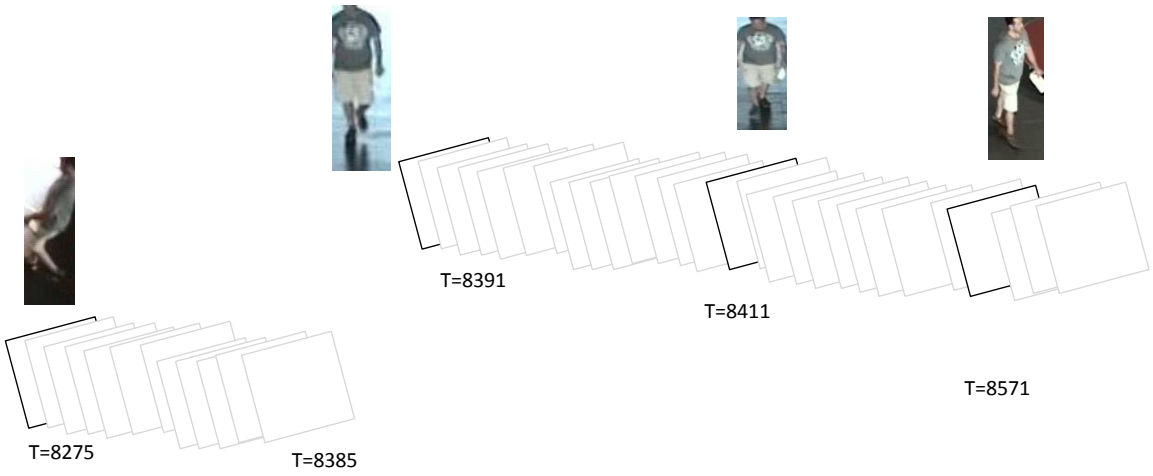
It is expected that the accuracy increases with the increase of the annotation effort.

Area under the learning curve (ALC). is a standard metric in active learning research that combines *accuracy* and *annotation effort* into a single measurement, which provides an average of accuracy over various budget levels. Herein, the learning curve is the set of accuracy plotted as a function of their respective annotation effort, a, $Accuracy = fa$. The ALC is obtained by:

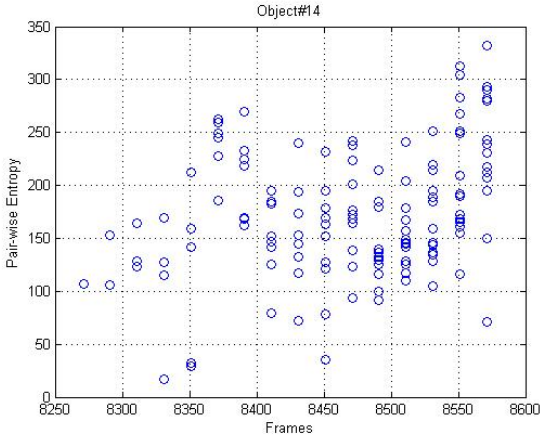
$$ALC = \int_0^1 f a da \tag{16}$$

6 RESULTS

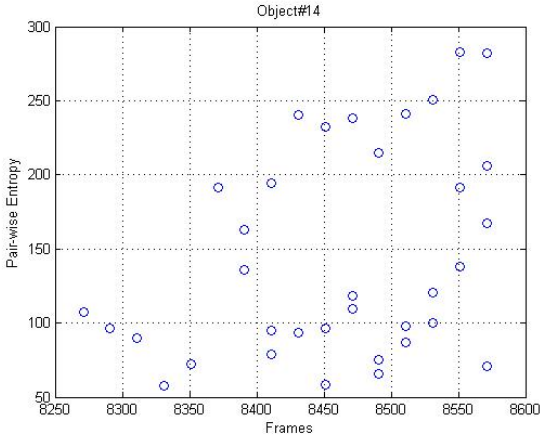
To evaluate the effectiveness of the adaptation algorithm on the size and accuracy of learning system, we compared our method with three baseline approaches. Figure 4 illustrates the comparative results across baseline approaches on multiple video datasets as a function of number of classifiers. We can clearly observe from the figure that keeping all the classifiers (denoted with red points) does not bring an advantage for the system. Our *wise update and add* strategy performs fairly well by keeping only a limited number of classifiers. The number of the learners is a function of number of classes that have been observed at the scene. For example, the system obtained 90% ALC (which is the best ALC obtained for this set) by keeping 21 models for the 7 classes present at "SAIVT-NonOver" dataset. The average cost is 3 models per person. However the cost increased for more occluded datasets (e.g. PETS with average cost of 6 classifiers per classes), still the framework controls a dramatic



(a) streams correspond to object #14



(b) Pair-wise entropy of the models present inside MCE #14 using NEVIL.ubm



(c) Pair-wise entropy of the models present inside MCE #14 using INEVIL

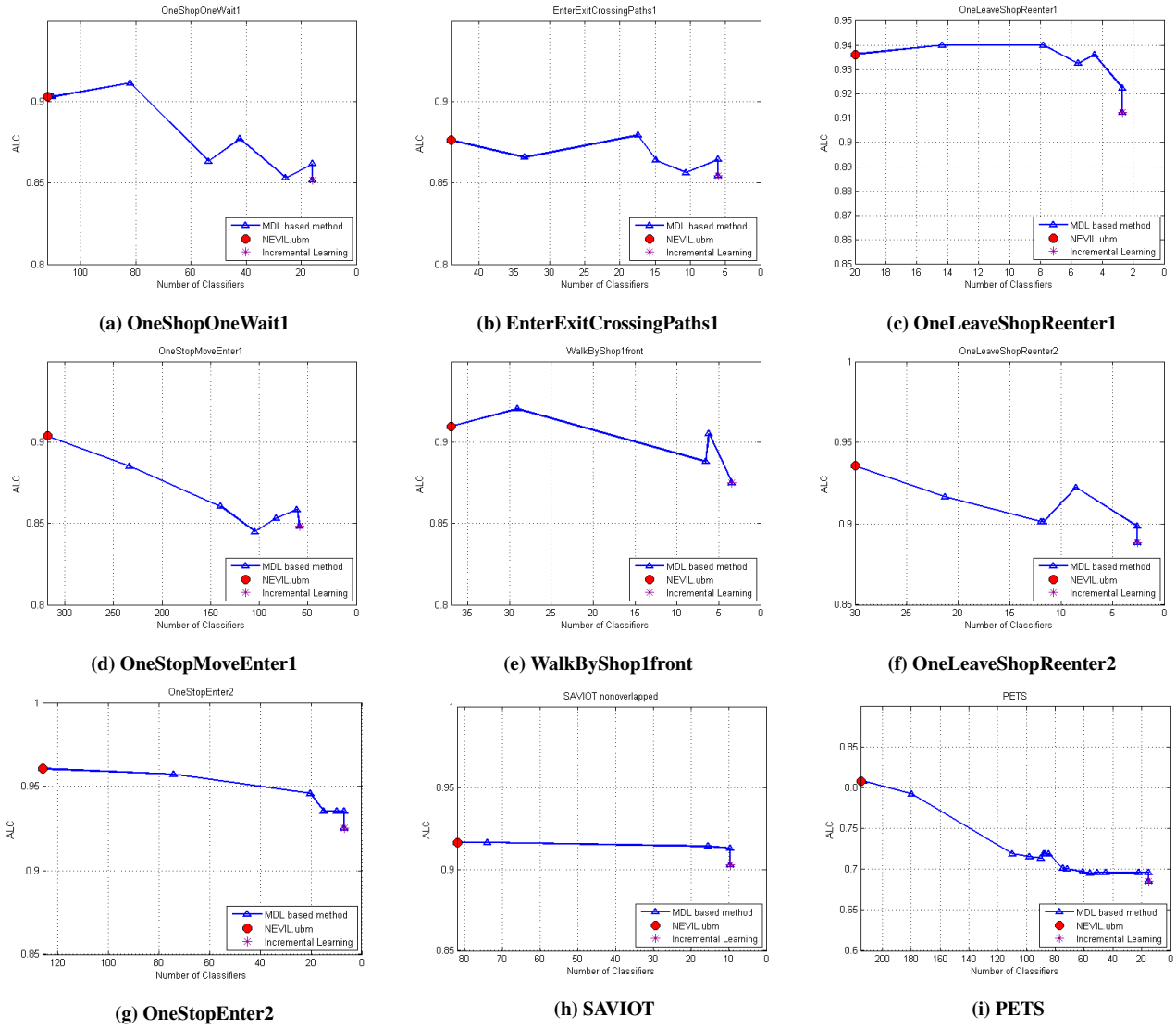
Figure 3: An example of micro-ensemble diversity using Model-Level adaptation mechanism

expansion of the size of the models without sacrificing or in some cases (e.g. EnterExitCrossingPath1, OneLeaveShopReenter1) even improving the performance.

A pair-wise distance between models inside an ensemble is applied as notion of diversity in this framework. Figure 3 shows an example of the the micro-ensemble diversity corresponds to subject #14 using NEVIL.ubm (Figure 3b) and INEVIL (Figure 3c) over time (horizontal axis denotes the frame number). Although the number of the models has been dramatically reduced using INEVIL (Figure 3c), range of diversity (difference between the minimum and maximum distance) has not been changed.

All the codes and data is publicly available through the first author's website.

Time Efficiency. Since the framework was developed in MATLAB without any efficiency concerns (running in an Intel Core i7 at 3.2GHz), a straightforward assessment of the time efficiency is not adequate. Number of classes and drift level are two key factors of the complexity. The complexity linearly increases as the number of classes grows in time. The higher drift level occurs, the more models is added to the framework. The experiments proved that if abrupt drift happens all the time, a bigger ensemble is generated, however it still manageable and does not explode. For a frame rate of 25fps, one second is spanned by the batch in our experiments. The analysis time grows naturally with the complexity of the dataset; however, the maximum processing time of a second video for the most complex dataset is less than half a second. Thus, the proposed framework is applicable for surveillance in real time.



(a) OneShopOneWait1 **(b) EnterExitCrossingPaths1** **(c) OneLeaveShopReenter1**

(d) OneStopMoveEnter1 **(e) WalkByShop1front** **(f) OneLeaveShopReenter2**

(g) OneStopEnter2 **(h) SAVIOT** **(i) PETS**

Figure 4: Comparing the performance of mid-level fusion with late fusion on some video clips.

7 CONCLUSIONS

We presented a novel strategy for long-term learning of the patterns of RoIs in various un-regulated streams. It employs an information theoretic-based criterion to predict and evaluate the potential of current knowledge (classifiers in an ensemble) to represent the new data. We embedded our strategy in the active learning framework, where the assessment triggers an adaptation process by either updating or training a new classifier. We experimentally investigated the impact of the proposed approach on accuracy and complexity over an un-bounded time frame in various possible scenarios in a multi-camera surveillance system. The results of our experiments indicate the potential of our approach for on-line applications, as they attained the promising performance with much lower runtime complexity.

For future work, we plan to employ concentration inequalities in order to make the query selection procedure wiser and automatic.

ACKNOWLEDGMENT

The authors would like to thank NWO-RATE Analytics project for supporting this work.

REFERENCES

[1] Nurjahan Begum and Eamonn Keogh. 2014. Rare Time Series Motif Discovery from Unbounded Streams. *Proc. VLDB Endow.* (2014), 149–160.
[2] Eugen Berlin and Kristof Van Laerhoven. 2012. Detecting Leisure Activities with Dense Motif Discovery. (2012), 250–259.
[3] Brais Cancela, Timothy M. Hospedales, and Shaogang Gong. 2014. Open-world Person Re-Identification by Multi-Label Assignment Inference. In *British Machine Vision Conference, BMVC 2014, Nottingham, UK, September 1-5, 2014*. http://www.bmva.org/bmvc/2014/papers/paper091/index.html
[4] Jaime S. Cardoso and Luis Corte-Real. 2005. Toward a Generic Evaluation of Image Segmentation. *IEEE Transactions on Image Processing* 14 (november

2005), 1773–1782. Issue 11. https://doi.org/10.1109/TIP.2005.854491
[5] Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka Jr., and Tom M. Mitchell. 2010. Toward an Architecture for Never-Ending Language Learning. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI*.
[6] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. 2014. Return of the Devil in the Details: Delving Deep into Convolutional Nets. In *BMVC*.
[7] Ling Chen, Lingjun Zou, and Li Tu. 2012. A clustering algorithm for multiple data streams based on spectral component similarity. *Information Sciences* 183, 1 (2012), 35–47.
[8] Xinlei Chen, Abhinav Shrivastava, and Abhinav Gupta. 2013. Neil: Extracting visual knowledge from web data. (2013), 1409–1416.
[9] Gregory Ditzler and Robi Polikar. 2013. Incremental Learning of Concept Drift from Streaming Imbalanced Data. *IEEE Transactions on Knowledge and Data Engineering* 25, 10 (2013), 2283–2301.
[10] Gregory Ditzler and Robi Polikar. 2013. Incremental Learning of Concept Drift from Streaming Imbalanced Data. *IEEE Trans. Knowl. Data Eng.* 25, 10 (2013), 2283–2301.
[11] Gregory Ditzler, Manuel Roveri, Cesare Alippi, and Robi Polikar. 2015. Learning in Nonstationary Environments: A Survey. *IEEE Comp. Int. Mag.* 10, 4 (2015), 12–25. https://doi.org/10.1109/MCI.2015.2471196
[12] Ryan Elwell and Robi Polikar. 2011. Incremental Learning of Concept Drift in Nonstationary Environments. *IEEE Transactions on Neural Networks* 22, 10 (2011), 1517–1531.
[13] Mário A. T. Figueiredo and Anil K. Jain. 2000. Unsupervised Learning of Finite Mixture Models. *IEEE Transaction on Pattern Recognition and Machine Intelligence* 24 (2000), 381–396.
[14] João Gama, Indre Zliobaite, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. 2014. A survey on concept drift adaptation. *ACM Comput. Surv.* 46, 4 (2014), 44:1–44:37. https://doi.org/10.1145/2523813
[15] Peter Grünwald. 2000. Model selection based on minimum description length. *Journal of Mathematical Psychology* 44, 1 (2000), 133–152.
[16] Yuan Hao, Yanping Chen, Jesin Zakaria, Bing Hu, Thanawin Rakthanmanon, and Eamonn Keogh. 2013. Towards Never-ending Learning from Time Series Streams *(KDD '13)*. 874–882.
[17] B. Hayete, M. Valko, A. Greenfield, and R. Yan. 2016. MDL-motivated compression of GLM ensembles increases interpretability and retains predictive power. *ArXiv e-prints* (Nov. 2016). arXiv:stat.ML/1611.06800
[18] Xian-Sheng Hua and Jin Li. 2015. Prajna: Towards Recognizing Whatever You Want from Images Without Image Labeling. (2015), 137–144.
[19] Abdolrahman Khoshrou, A Pedro Aguiar, and Fernando Lobo Pereira. 2016. Adaptive Sampling Using an Unsupervised Learning of GMMs Applied to a Fleet of AUVs with CTD Measurements. In *Second Iberian Robotics Conference*. 321–332.
[20] Samaneh Khoshrou, Jaime S. Cardoso, Eric Granger, and Luís Filipe Teixeira. 2016. Unsupervised Long-term Monitoring Over Multiple Video Streams. In *Under Review*.
[21] Samaneh Khoshrou, Jaime S. Cardoso, and Luís Filipe Teixeira. 2014. Active Mining of Parallel Video Streams. *CoRR* abs/1405.3382 (2014).
[22] Samaneh Khoshrou, Jaime S. Cardoso, and Luís Filipe Teixeira. 2015. Learning from evolving video streams in a multi-camera scenario. *Machine Learning* 100, 2-3 (2015), 609–633. https://doi.org/10.1007/s10994-015-5515-y
[23] D. Reynolds. 2008. Gaussian mixture models. *Encyclopedia of Biometric Recognition* (2008), 12–17.
[24] D.A. Reynolds, T.F. Quatieri, and R.B. Dunn. 2000. Speaker verification using adapted Gaussian mixture models. *Digital signal processing* 10, 1 (2000), 19–41.
[25] Jorma Rissanen. 2010. Minimum Description Length Principle. In *Encyclopedia of Machine Learning*. 666–668. https://doi.org/10.1007/978-0-387-30164-8_540
[26] Burr Settles. 2009. *Active Learning Literature Survey*. Technical Report 1648. University of Wisconsin–Madison.
[27] K. Shinoda and N. Inoue. 2013. Reusing Speech Techniques for Video Semantic Indexing [Applications Corner]. *Signal Processing Magazine, IEEE* 30, 2 (2013), 118–122.
[28] Dieter Fox Yuyin Sun. 2016. NEOL: Toward Never-Ending Object Learning for Robots. IEEE International Conference on Robotics and Automation (ICRA).
[29] Yi Zhang, Samuel Burer, and W. Nick Street. 2006. Ensemble Pruning Via Semi-definite Programming. *J. Mach. Learn. Res.* 7 (Dec. 2006), 1315–1338. http://dl.acm.org/citation.cfm?id=1248547.1248595