Ensemble classification of video-recorded crowd movements

Mounir Bendali-Braham, Jonathan Weber, Germain Forestier, Lhassane Idoumghar and Pierre-Alain Muller
IRIMAS, Université de Haute-Alsace, Mulhouse, France
Email: {first-name.last-name@uha.fr}

***Abstract*—Ensemble learning methods often improve results in problems addressed by single Machine Learning models. In this work, we apply Ensemble Learning on video-recorded crowd movements. First, we build Ensembles of homogeneous Convolutional Neural Networks (CNN) to compare their performance on the Crowd-11 dataset and show the gain of performance demonstrated by Ensembles compared to single CNN models. Secondly, we evaluate all the possible combinations of these homogeneous Ensembles to build a global Ensemble of heterogeneous models, and we analyze the combination of Ensembles that achieves the best results. Our experiments reveal that Ensemble classification often obtains better results than single models and combining different Ensembles can make the predictions accuracy even better.**

***Index Terms*—Ensemble classification, Deep learning, Video analysis, Crowd behavior analysis**

I. Introduction

More and more cities are subject to massive crowd movements due to cultural or political events. During a crowd movement, law enforcement services rely on video surveillance and can count on recent approaches allowing an optimal deployment of cameras [1]. However, although the video data collection is increasingly widespread, the automatic analysis of these videos is not always performed in real time which can delay the intervention of law enforcement services. One of the reasons for this is the lack of generic statistical models that can be used in real time to detect any type of anomalies that may arise from crowd events [2]. Furthermore, the development of such models is a tedious task. This can be explained by the scarcity of annotated datasets [3]. However, recent years have seen the emergence of datasets illustrating massive and various crowd movements such as Crowd-11 [4].

The authors of Crowd-11 trained models to classify crowd clips. The model that obtains the best classification results in their article derives from the C3D architecture [5]. In previous works, we obtained better results [6], by using a model derived from the TwoStream Inflated 3D architecture (2S-I3D) which already outperforms the C3D models on action recognition datasets [3].

In this article, we aim to improve the classification results on the Crowd-11 dataset by applying Ensemble Learning methods. We first create Ensembles of homogeneous models deriving from different architectures and benefiting from different training pre-conditions. The theoretical aspect of this comparison is developed in Section IV-A. Secondly, an evaluation of all the possible combinations of these models allows us to elect a global Ensemble of heterogeneous models that gathers several homogeneous Ensembles. The theoretical aspect of this combination is developed in Section IV-C.

This article is organized as follows: in Section II, we present Ensemble methods applied to video classification and crowd analysis. In Section III, we present our approaches. We discuss our experiments in Section IV.

II. Review

Ensemble methods perform very well in several machine learning tasks [7]. Zhou [8] divides the Ensemble methods into three major categories:

- Boosting, illustrated by its most famous algorithm AdaBoost [9], which consists in learning T models by associating each time different weights to the training examples. At the beginning, of similar values, these weights change at the t^{th} iteration of the AdaBoost algorithm by taking into account the error obtained from the model trained on $t^{th} - 1$ iteration. In the end, a weighted majority vote is used to combine the decisions of the T models.
- Bagging, which is a contraction of Bootstrap Aggregating, where statistical methods are trained on samples created by Bootstrap sampling [10]. Subsequently, these methods are combined into an Ensemble by a majority vote.
- Stacking, where different statistical methods are trained on a dataset. Subsequently, a second statistical method, called a meta-classifier, learns to combine the trained models.

Furthermore, Zhou considers that some Ensemble methods do not fall into any of these three major categories.

Here we explore some recent Ensemble approaches applied to image processing, video analysis, and crowd analysis.

For image processing and video analysis, Lia et al. [11] apply an Ensemble method to provide a solution to classes that lack examples for a vehicle image classification problem. They apply balanced sampling and data augmentation. Their Ensemble method consists of a combination of multiple ResNet models [12] and the decision is inferred using the majority vote.

Pouyanfar et al. [13] propose EDL (Ensemble Deep Learning) which they use for the classification of videos on the Trecvid [14] and Disaster [15] datasets. EDL is a suite of deep feature extractor models from images, which are Convolutional Neural Networks (CNNs) [16] pre-trained on ImageNet, and each extractor is followed by a Support Vector Machine (SVM) [17] which serves as a weak learner in the Ensemble. The learned features are extracted from the last Fully Convolutional Network (FCN) layer of each model. The used architectures are: AlexNet [18], CaffeNet [19], Region based CNN [20], GoogleNet [21], and ResNet. The decision is inferred following a weighted vote.

Inspired by Liu et al. [22], Chen et al. [23] propose an Ensemble approach named Ensemble Weighted Multi-Instance Learning. They start by sampling several subsets of the majority class, and by combining each time a subset of the majority class with a minority class, they train a model using AdaBoost [9]. The trained models are combined for the final decision.

In the context of crowd analysis, Walach et al. [24] apply gradient boosting and selective sampling on a simple CNN architecture to perform objects counting in images. Their approach is applied to microscopic bacterial cells datasets, and crowd counting datasets.

Wu et al. [25] stack several models whose outputs are used as new features which will be sent as inputs of a new model.

Due to the paucity of annotated data, Gong et al. [26] learn in a semi-supervised manner an Ensemble of pose-sensitive DPM (Deformable Part-based Model) mixtures [27] for pedestrian detection. Several postures are taken into account such as: front, rear, left, right. Each DPM mixture specialized to a specific pedestrian posture is trained using a Latent-SVM [28].

Contrary to previously mentioned approaches [11], [22], we do not aim to solve the problem of unbalanced data. We do not do Boosting, as in Walach et al. [24], because in Boosting the training sessions are repeated several times to change the weights of the training examples. The same cannot be reproduced for video data without requiring a huge calculation time. We opt for a compromise between a form of Stacking [25], without a meta-classifier because we combine the models at the evaluation phase, and a form of Bagging, because we perform an aggregation of models without applying Bootstrap sampling. Here the samples are the folds already obtained following the cross validation that we did for our previous work [6]. Our split is stratified, which means that each fold maintains the classes distribution of the original dataset. We do not do semi-supervised learning combined with the use of Ensemble methods, as in Gong et al. [26], because our dataset does not suffer from missing annotations. All the video clips are weakly annotated here.

Zhou et al. [29] argue that it is not useful to put a very large number of models in an Ensemble. Choosing a small number of models that are already yielding good results is enough to yield better results when they end up gathering into an Ensemble. Therefore, we decided to split the dataset into 5 folds, as it was already done in Bendali-Braham et al. [6], which allows each Ensemble to be equipped with 4 single models that extract different knowledge from the Crowd-11 dataset.

III. Ensemble classification

In the light of the approaches mentioned in Section II, and based on the definition proposed by Zhou [8], Ensemble learning consists in training, or evaluating, a set of statistical models, whether they are of similar nature or not, trained in similar conditions or not.

First, we set up Ensemble approaches made up of homogeneous models that possess the same pre-training conditions. Afterwards, we propose to create global Ensemble approaches which mix heterogeneous models of different architectures and that have various pre-training conditions.

A. Creation of homogeneous models ensembles

In a previous work on the Crowd-11 dataset, we showed that models from the 2S-I3D network perform better than the C3D network and the Inflated 3D Nets (I3D) [6]. However, the 2S-I3D results peak approximately at 68% accuracy. These results were confirmed by a 5-fold cross-validation.

In this work, we split the dataset into 5 folds, and we train, for each possible combination of these folds, a model from one of the following architectures:

- The 2S-I3D architecture [3],
- The I3D architecture [3].
- The C3D architecture [5].
- The Resnet 3D (R3D) architecture [30].

In each combination, 3 folds are used for training, 1 fold for validation, and the last fold for testing. By selecting a test fold each time, we can produce 4 models by combining the remaining folds. During the evaluation on a test fold, the decisions of the 4 models are summed to deduce the decision.

With this procedure, we can make up to 20 different combinations of the training, validation, and test sets. We mention in the following paragraphs the composition of each of the used architectures.

The I3D architecture is composed of a basis of two 3D convolution layers. Each of these layers is supported by a Batch Normalization and is followed by a 3D MaxPooling operation. These 2 layers are followed by 9 Inception modules whose internal composition changes slightly from one module to another. The last Inception module is connected to a 3D AveragePooling whose outputs are sent to a SoftMax function for the classification task.

The 2S-I3D architecture is made up of two streams. Each branch reproduces the architecture of the I3D network. One of the two streams extracts features from an RGB video clip, and the other one extracts features from an optical flow version of the video clip. The outputs of these two streams are connected to the Softmax classification function.

The C3D architecture, proposed by Tran et al. [5], consists of 5 layers of 3D convolutions, followed by two layers of FCN whose outputs are sent to a Softmax classification function.

The R3D architecture, proposed by Hara et al. [30], is made up of several residual blocks. Each residual block is composed of two 3D convolution layers. We choose the version with 34 hidden layers because of its good performance which is demonstrated by Hara et al. [30] on the Sports-1m action recognition dataset. This version of the R3D architecture consists of a first layer of 3D convolutions followed by 16 residual blocks, and ends with an FCN layer before the Softmax classification function.

1) Constitution of training, validation and test sets: The version of the Crowd-11 dataset available to us corresponds to the version that we worked on in our previous project [6]. This version of Crowd-11 consists of 1641 scenes that can be considered as contextual boundaries between video clips. These scenes are split into 5769 video clips which are labeled into 11 classes by Dupont et al. [4]. These 11 classes are: Gas Free, Gas Jammed, Laminar Flow, Turbulent Flow, Crossing Flows, Merging Flows, Diverging Flow, Static Calm, Static Agitated, Interacting Crowd, No Crowd.

In order to split the dataset into folds to apply cross-validation, we divided the dataset starting from the scenes in our previous work [6]. To ensure that the division of the dataset is stratified, we apply the algorithms, presented below:

1) The first algorithm splits the scenes of the dataset into multiple folds. In our case, we split it into 5 folds.
2) The second algorithm is by the first algorithm to update the score of each fold w.r.t its distribution.

These two algorithms are used to constitute the folds that will be used for the creation of the training, validation, and test sets, as illustrated in Figure 1.

The following program splits the scenes of the dataset into multiple folds
$Sc \leftarrow$ list of scenes
$Nb\_folds \leftarrow$ number of folds
$Sc\_freq \leftarrow$ array listing the number of clips per scene
$Cls\_freq \leftarrow$ array listing the number of clips per class {Sc_freq and Cls_freq are pre-computed}
$Folds\_scenes \leftarrow$ array of lists of scenes of each fold {The dataset is split into the $Folds\_scenes$ scenes.}
$Folds\_distribs \leftarrow$ scores distributions of each fold with respect to the diversity of the clips it contains
while Sc *not empty* **do**

- Select the fold with the worst score from $Folds\_distribs$
- Select the scene that contains the biggest number of clips from Sc_freq
- Remove the selected scene from Sc_freq and Sc
- Add the selected scene to the selected fold on $Folds\_scenes$
- Update the score of the selected fold on $Folds\_distribs$

end while
return $Folds\_scenes$

The following program updates the score of a selected fold
Require: $Folds\_distrib$, s, Nb_folds, Cls_freq
$s \leftarrow$ scene previously selected
$Vids \leftarrow$ list of all the dataset's clips
$Database \leftarrow$ dataframe containing the information about the dataset that link the scenes to their video clips
$Sc\_vids \leftarrow$ intersection between the scene s and the dataframe $Database$, and retrieval of all the video clips of these scenes from $Vids$
for all *class* c *in* $Sc\_classes$ **do**
$Fold\_distrib_c \leftarrow Fold\_distrib_c + \frac{Nb\_folds}{Cls\_freq_c}$
end for
return New fold's score $Fold\_distrib$ stored in $Folds\_distribs$

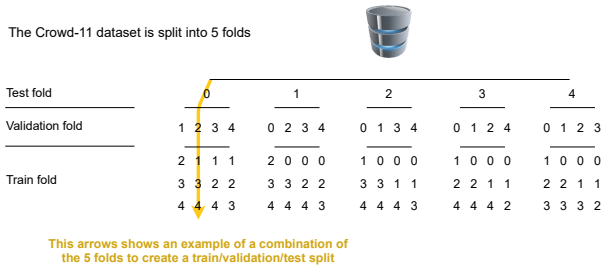

Fig. 1. Illustration of the constitution of the train, validation and test sets from different combinations of folds of the Crowd-11 dataset

2) Global ensembles of heterogeneous models: We create global Ensembles of models having either different architectures, e.g. fine-tuned 2S-I3D Ensembles coupled with C3D Ensembles trained from scratch, or different training conditions, e.g. fine-tuned I3D Ensembles and I3D Ensembles trained from scratch, or various Ensembles accumulating the two differences, for example C3D Ensembles trained from scratch, fine-tuned I3D Ensembles, and 2S-I3D Ensembles trained from scratch. We compose global Ensembles from the Ensembles of homogeneous models compared in Section IV-A. Next, we evaluate on the test set all the possible combinations from these Ensembles of homogeneous models. Equation (1) computes the number of combinations without repetition that can give rise to global Ensembles.

$$nb\_combinations = \sum_{i=2}^{K} C(K, i) \tag{1}$$

Where K represents the maximum size of a combination constituting a global Ensemble. $C(K, i)$ represents the function that computes a combination without repetition where the number of choices is set to i. i is the length of the tuple which represents the number of homogeneous Ensembles combined into a global Ensemble of heterogeneous models. As we already evaluate the Ensembles of homogeneous models in Section IV-A, i starts from 2 which is considered as the minimum size of a combination[1].

[1]The source code of this project is available here: https://github.com/MounirB/Crowded-scenes-Ensemble-classification

IV. Experiments

In this section, we detail the different types of experiments that we have conducted:

- We compared the performance of Ensembles made up of fine-tuned models to other Ensembles made up of models which did not benefit from pre-training;
- We compared Ensembles that were trained on augmented data from the Crowd-11 dataset and compared them to Ensembles that did not benefit from augmented data.

In this work, the chosen hyperparameters for the training process correspond to the hyperparameters that we used in Bendali-Braham et al. [6]. The video clips of the Crowd-11 dataset last approximately for ≈ 5 seconds. For I3D and 2S-I3D architectures, 20 frames are selected from a video clip. These frames are found at regular intervals all along the clip. The size of each image is fixed to 224×224 pixels. For R3D and C3D architectures, 16 frames are selected from a video clip and the size of each frame is fixed to 112×112 pixels. For the 2S-I3D models, the optical flow version of each clip is obtained via the TV-L1 algorithm [31]. During the comparison of the Ensembles that benefited from data augmentation or not, we will substitute the Farnebäck optical flow extraction algorithm for the TVL1 algorithm because the former is quicker than the latter. However, we also undertake experiments to check if the use of the Farnebäck algorithm does not significantly reduce the performance of the 2S-I3D models. The Farnebäck algorithm is theoretically known to be less accurate than the TVL1 algorithm.

A. Comparison of Ensembles of models with homogeneous architectures

We want to verify whether an Ensemble of fine-tuned models perform better than an Ensemble trained from scratch. Fine-tuned 2S-I3D and I3D models were pre-trained on the ImageNet [32] and the Kinetics [33] datasets. The fine-tuned C3D models were pre-trained on the Sports-1m dataset [34]. Furthermore, we train from scratch models from the 2S-I3D, I3D, C3D, and R3D architectures.

In all these situations, we create 5 major training/validation and testing contexts, where we fix a test sample upstream, and we vary the validation samples (that must be different from the test sample) downstream. Under these circumstances, for each Ensemble, three samples are selected for training. These latter are different from the test sample and the selected validation sample. At the end, for the 5 test folds, 20 individual models are trained from scratch or fine-tuned. Each group of 4 singles models, whether trained from scratch or fine-tuned, constitutes an Ensemble of models at the evaluation phase.

The prediction results of these models are evaluated in terms of accuracy in Table I. Overall results demonstrate that the constitution of Ensemble models increases the average accuracy for the classification task.

B. Data augmentation

By augmenting the video data of Crowd-11, we aim to evaluate the impact of data augmentation on the classification of video-recorded crowd movements. Data augmentation was only applied on the training sets of the Crowd-11 dataset. To augment data, we applied the following operations: random crop, salt and pepper effect, video flip. In our experiments, we compare two data augmentation strategies:

- A fixed pre-computed data augmentation before the beginning of the training session. The augmentation methods are chosen randomly, alone or combined with other augmentation methods. As a result of data augmentation, the size of the training set was multiplied by 3. In each epoch of the training session, the model explores the augmented data as well as the non-augmented data.
- An on-the-fly data augmentation that is renewed at each epoch of the training session. In this context, at each training epoch, data augmentation has a probability of 75% to occur. The epochs are repeated 4 times to allow the on-the-fly data augmentation to be similar in terms of quantity and variety to that of the pre-computed augmentated data.

a) Discussion of the results of data augmentation: As a result of our experiments illustrated in Table II, we find that using the Farnebäck extraction algorithm slightly reduces the performance of 2S-I3D models. The average score goes from 69.02% to 68.41%, which is an insignificant reduction that can be neglected for the rest of our experiments. On-the-fly data augmentation does not improve the results. Conversely, this type of data augmentation worsens the performance of the 2S-I3D models. The pre-computed data augmentation greatly improves the results of fine-tuned 2S-I3D models whose second branch feeds on flow video clips obtained via the Farnebäck algorithm. These Ensembles even progress, on average, by 1 point of accuracy from 68.41% to 69.81% thus beating the Ensembles of fine-tuned 2S-I3D models whose second streams were fed by the pre-computed flow clips extracted using the TVL1 algorithm. This good performance is offset by the 5 times greater training time required by models benefiting from the precomputed data augmentation.

C. Evaluation of global sets of models with heterogeneous architectures

We create global Ensembles by combining the models of homogeneous Ensembles whose results are illustrated in Table III. These 8 Ensembles can participate into 247 combinations containing at least 2 heterogeneous Ensembles.

The combination that achieves the best results combines the fine-tuned 2S-I3D Ensembles that have benefited from pre-computed data augmentation, with the fine-tuned 2S-I3D Ensembles that have not benefited from data augmentation, fine-tuned C3D Ensembles, and I3D Ensembles trained from scratch. The results of this combination are shown in Table IV. We find that this combination improves the overall performance by 1.5% in terms of accuracy.

Table IV shows that the Ensembles complete each other for the classification task. In this case, the least performing Ensembles do not curb the performance of the best global Ensemble.

TABLE I
COMPARISON BETWEEN THE RESULTS OBTAINED BY THE PRE-TRAINED 2S-I3D ENSEMBLES FINE-TUNED ON CROWD-11 AND THEIR SINGLE MODELS

Test sample involved	**0**	**1**	**2**	**3**	**4**	μ	σ
Validation sample: accuracy per associated individual model	1: 67.86	0: 66.55	0: 66.04	0: 63.09	0: 69.73		
	2: 69.08	2: 66.55	1: 67.28	1: 66.26	1: 70.60		
	3: 67.33	3: 66.29	3: 68.52	2: 63.52	2: 69.30		
	4: 69.86	4: 65.44	4: 66.66	4: 62.15	3: 67.39		
Accuracies standard deviations of the models sharing the same test sample	0.99	0.45	0.91	1.52	1.17		
Accuracies mean of the models sharing the same test sample	68.53	66.21	67.13	63.76	69.26	66.98	**1.92**
Accuracy per ensemble	**70.48**	**67.57**	**68.61**	**66.43**	**72.00**	**69.02**	1.99

TABLE II
PERFORMANCE COMPARISON BETWEEN ENSEMBLES OF 2S-I3D MODELS THAT WERE FINE-TUNED OR WERE NOT FINE-TUNED ON AUGMENTED DATA

Involved test sample	**0**	**1**	**2**	**3**	**4**	μ	σ
2S-I3D Ensembles FarneBack (Flow) Non Augmented accuracies	70.56	66.55	69.93	64.21	70.78	68.41	2.59
2S-I3D Ensembles FarneBack (Flow) Augmented Precomputed accuracies	**71.00**	**69.10**	**71.88**	**65.58**	**71.47**	**69.81**	2.31
2S-I3D FarneBack (Flow) Augmented On The Fly accuracies	68.47	67.23	69.23	64.21	69.30	67.69	**1.89**

TABLE III
COMPARISON BETWEEN MODELS ENSEMBLES WITH HOMOGENEOUS ARCHITECTURES

Test sample	**0**	**1**	**2**	**3**	**4**	μ	σ
C3D scratch	31.26	32.76	32.27	31.76	38.08	33.23	2.47
C3D pretrained	61.13	60.59	61.00	58.13	61.56	60.48	**1.21**
I3D scratch	54.93	55.91	58.53	53.85	58.86	56.42	1.97
I3D pretrained	64.10	60.25	62.24	57.70	60.95	61.05	2.12
R3D (w 34 layers) scratch	47.42	52.00	50.13	48.63	50.43	49.72	1.57
2S I3D scratch (TVL1)	54.41	56.42	60.83	54.45	61.30	57.48	3.01
2S I3D pretrained (TVL1) w/o DA	70.48	67.57	68.61	**66.43**	**72.00**	69.02	1.99
2S I3D pretrained (Farnebäck) w DA	**71.00**	**69.10**	**71.88**	65.58	71.47	**69.81**	2.31

TABLE IV
COMPARISON BETWEEN THE BEST COMBINATION GIVING RISE TO A GLOBAL SET AND THE SET MODELS CONSTITUTING IT

Test sample	**0**	**1**	**2**	**3**	**4**	μ	σ
(1) C3D pretrained	61.13	60.59	61.00	58.13	61.56	60.48	**1.21**
(2) I3D scratch	54.93	55.91	58.53	53.85	58.86	56.42	1.97
(3) 2S I3D pretrained (TVL1) w/o DA	70.48	67.57	68.61	**66.43**	72.00	69.02	1.99
(4) 2S I3D pretrained (Farnebäck) w DA	71.00	69.10	71.88	65.58	71.47	69.81	2.31
Global ensemble (1) + (2) + (3) + (4)	**72.05**	**70.04**	**73.20**	66.35	**74.95**	**71.32**	2.95

The main result of this work is that Ensemble learning improves the performances of the classification task and combining multiple Ensembles in a global Ensemble can make the results even better. Despite these good results, it is worth noting that the computation time for the decision making of the Ensembles is obviously higher than for the individual models.

V. CONCLUSION AND PERSPECTIVES

Ensemble learning improves the accuracy of the video classification task. In this work, we verified this improvement for video-recorded crowd movements illustrated by the Crowd-11 dataset. First, we found that the Ensemble of fine-tuned 2S-I3D models improve the results of their single models which were trained on different training samples. Here, thanks to Ensemble methods, the average accuracy of single models increases from 66.98% to 69.02%.

Afterwards, the comparison between the Ensembles of homogeneous models that have different training pre-conditions and deriving from different architectures shows the superiority of the pre-computed data augmentation strategy for the Ensembles of 2S-I3D models. This augmentation strategy is the appropriate regularization method that allows these Ensembles to generalize well. The pre-computed data augmentation strategy helped the 2S-I3D Ensembles to increase their accuracy from 69.02% to 69.81%.

Finally, the best global Ensemble of heterogeneous models increases the classification accuracy from 69.81% to 71.32%.

Currently, our models do not realize real-time predictions. As a remedy, we intend to focus in our future work transferring the knowledge learned by our Ensembles to lighter and quicker models, by applying knowledge distillation; a concept coined by Hinton et al. [35].

ACKNOWLEDGMENTS

The authors would like to thank NVIDIA Corporation for the GPU Grant and the Mésocentre of Strasbourg for providing access to the GPU cluster. This work was supported by the ANR OPMoPS project (grant ANR-16-SEBM-0004) funded by the French National Research Agency.

REFERENCES

[1] J. Kritter, M. Brévilliers, J. Lepagnot, and L. Idoumghar, "On the real-world applicability of state-of-the-art algorithms for the optimal camera placement problem," in *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*. IEEE, 2019, pp. 1103–1108.

[2] M. Thida, Y. L. Yong, P. Climent-Pérez, H.-l. Eng, and P. Remagnino, "A literature review on video analytics of crowded scenes," in *Intelligent multimedia surveillance*. Springer, 2013, pp. 17–36.

[3] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6299–6308.

[4] C. Dupont, L. Tobias, and B. Luvison, "Crowd-11: A dataset for fine grained crowd behaviour analysis," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, vol. 2017-July, Honolulu, United States, 2017, pp. 2184–2191. [Online]. Available: https://hal-cea.archives-ouvertes.fr/cea-01831840

[5] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497.

[6] M. Bendali-Braham, J. Weber, G. Forestier, L. Idoumghar, and P.-A. Muller, "Transfer learning for the classification of video-recorded crowd movements," in *2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA)*. IEEE, 2019, pp. 271–276.

[7] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, "Deep neural network ensembles for time series classification," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–6.

[8] Z.-H. Zhou, "Ensemble learning." *Encyclopedia of biometrics*, vol. 1, pp. 270–273, 2009.

[9] Y. Freund and R. E. Schapire, "A desicion-theoretic generalization of on-line learning and an application to boosting," in *European conference on computational learning theory*. Springer, 1995, pp. 23–37.

[10] B. Efron, "Bootstrap methods: another look at the jackknife," in *Breakthroughs in statistics*. Springer, 1992, pp. 569–593.

[11] W. Liu, M. Zhang, Z. Luo, and Y. Cai, "An ensemble deep learning method for vehicle type classification on visual traffic surveillance sensors," *IEEE Access*, vol. 5, pp. 24 417–24 425, 2017.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[13] S. Pouyanfar and S.-C. Chen, "Automatic video event detection for imbalance data using enhanced ensemble deep learning," *International Journal of Semantic Computing*, vol. 11, no. 01, pp. 85–109, 2017.

[14] A. F. Smeaton, P. Over, and W. Kraaij, "Evaluation campaigns and trecvid," in *Proceedings of the 8th ACM international workshop on Multimedia information retrieval*, 2006, pp. 321–330.

[15] S. Pouyanfar and S.-C. Chen, "Semantic concept detection using weighted discretization multiple correspondence analysis for disaster information management," in *2016 IEEE 17th International Conference on Information Reuse and Integration (IRI)*. IEEE, 2016, pp. 556–564.

[19] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM international conference on Multimedia*, 2014, pp. 675–678.

[16] Y. LeCun, Y. Bengio *et al.*, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, p. 1995, 1995.

[17] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[20] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[21] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[22] X.-Y. Liu, J. Wu, and Z.-H. Zhou, "Exploratory undersampling for class-imbalance learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 2, pp. 539–550, 2008.

[23] G. Chen, M. Giuliani, D. Clarke, A. Gaschler, and A. Knoll, "Action recognition using ensemble weighted multi-instance learning," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 4520–4525.

[24] E. Walach and L. Wolf, "Learning to count with cnn boosting," in *European conference on computer vision*. Springer, 2016, pp. 660–676.

[25] C. Wu, T. Yin, S. Ge, and K. Yu, "Ensemble learning for crowd flows prediction on campus," in *International Conference on Smart Computing and Communication*. Springer, 2017, pp. 103–113.

[26] S. Gong, T. Xiang, and S. Hongeng, "Learning human pose in crowd," in *Proceedings of the 1st ACM international workshop on Multimodal pervasive video analysis*, 2010, pp. 47–52.

[27] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627–1645, 2009.

[28] C.-N. J. Yu and T. Joachims, "Learning structural svms with latent variables," in *Proceedings of the 26th annual international conference on machine learning*, 2009, pp. 1169–1176.

[29] Z.-H. Zhou, J. Wu, and W. Tang, "Ensembling neural networks: many could be better than all," *Artificial intelligence*, vol. 137, no. 1-2, pp. 239–263, 2002.

[30] K. Hara, H. Kataoka, and Y. Satoh, "Learning spatio-temporal features with 3d residual networks for action recognition," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2017, pp. 3154–3160.

[31] C. Zach, T. Pock, and H. Bischof, "A duality based approach for realtime tv-l 1 optical flow," in *Joint pattern recognition symposium*. Springer, 2007, pp. 214–223.

[32] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE conference on computer vision and pattern recognition*, 2009, pp. 248–255.

[33] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev *et al.*, "The kinetics human action video dataset," *ArXiv*, 2017.

[34] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1725–1732.

[35] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *ArXiv*, 2015.