Apprentissage par transfert pour la classification de séquences vidéo de mouvements de foule

Mounir Bendali-Braham, Jonathan Weber
Germain Forestier, Lhassane Idoumghar, Pierre-Alain Muller

IRIMAS, Université de Haute-Alsace, 68100 Mulhouse, France
prénom.nom@uha.fr

Résumé. La reconnaissance automatique d'un mouvement de foule, capturé par une caméra de vidéo-protection, peut être d'une aide considérable pour les forces de l'ordre qui ont pour mission d'assurer la sécurité des personnes sur la voie publique. Dans ce cadre, nous proposons d'entraîner un modèle issu de l'architecture *TwoStream Inflated 3D*, pré-entrainé sur les jeux de données sources ImageNet et Kinetics, à classer des séquences vidéo de mouvements de foule issues du jeu de données cible Crowd-11. En comparant nos résultats à l'état-de-l'art, notre modèle supplante son score de précision.

1 Introduction

Afin de gérer un mouvement de foule en amont, les forces de l'ordre peuvent recourir à l'usage des caméras de vidéo-protection (Drews et al., 2010; Sultani et al., 2018; Porikli et al., 2013). L'installation de ces caméras couvre une grande partie de l'espace public (Kritter et al., 2019). Bien que l'un de leurs usages le plus commun est la récupération d'images attestant d'une activité criminelle et leur emploi par la suite à des fins judiciaires, l'usage qui commence à en être fait est l'analyse des comportements de foule afin de prédire des situations anormales (Sultani et al., 2018). Toutefois, malgré l'abondance d'images brutes, provenant des caméras de vidéo-protection, il n'existe pas à ce jour de modèle issu de l'apprentissage profond qui serve dans tous les scénarios possibles de scènes de foule. Ceci est dû à la rareté de données annotées disponibles publiquement (Carreira et Zisserman, 2017).

Récemment, une équipe du CEA (Commissariat à l'énergie atomique et aux énergies alternatives) a créé un jeu de données appelé Crowd-11 (Dupont et al., 2017). Ce jeu de données, de plus de 6000 séquences vidéo, constitue une contribution majeure pour l'analyse du comportement de foule car il décrit une dizaine de comportements observables dans la voie publique.

Dans ce travail, nous appliquons l'apprentissage par transfert pour classer les séquences vidéo de mouvements de foule. Dans ce cadre, notre tâche consiste à étiqueter une vidéo. Pour ce faire, nous affinons un modèle issu de l'architecture *TwoStream Inflated 3D ConvNet* (*TwoStream-I3D*) (Carreira et Zisserman, 2017) pré-entrainé sur les jeux de données ImageNet (Deng et al., 2009) et Kinetics (Kay et al., 2017), sur ce qui a pu être récupéré du jeu de données Crowd-11. Le modèle *TwoStream-I3D* affiné est comparé à un modèle issu de l'architecture *3D Convolutional Networks* (*C3D*) (Tran et al., 2015), pré-entrainé sur le jeu de

données Sports-1m avant d'être affiné sur Crowd-11. La suite du papier est organisée comme suit : dans la Sous-section 2.1, nous parlons brièvement de ce qui se fait dans le domaine de l'analyse des foules. Dans la Sous-section 2.2, nous présentons le jeu de données Crowd-11. Dans la Section 3, nous introduisons l'apprentissage par transfert dans le cadre de la classification de vidéos, et nous présentons les architectures pour lesquelles nous l'avons appliqué. Dans la Section 4, nous présentons les différents modèles que nous avons entrainés, et puis évalués, sur le jeu de données Crowd-11.

2 Contexte

2.1 État de l'art

Depuis plus de deux décennies, l'analyse des foules fait partie de la recherche en vision par ordinateur. Les travaux réalisés dans ce domaine se subdivisent en deux grandes catégories : le calcul des statistiques de foule, et l'analyse des comportements de foule (Zhan et al., 2008; Lamba et Nain, 2017; Grant et Flynn, 2017).

Calcul des statistiques de foule :

- **Comptage du nombre de personnes d'une foule :** comme son nom l'indique, cette branche de l'estimation des statistiques de foule consiste à compter le nombre de personnes constituant la ou les foules d'individus capturées dans une scène (Ranjan et al., 2018).
- **Estimation de la densité des foules :** les travaux qui estiment la densité d'une scène de foule, tel que Xu et al. (2017), peuvent être d'une aide considérable pour les forces de l'ordre ou les organisations de gestion des mouvements de foule.

Analyse des comportements de foule :

- **Analyse des trajectoires :** l'analyse des trajectoires fait partie de ce qui se fait le plus en analyse des comportements de foule (Lu et al., 2017). L'analyse des trajectoires peut aider à détecter des groupes d'individus (Solera et al., 2015), détecter des trajectoires anormales (Coşar et al., 2016), ou prédire l'évolution des trajectoires (Alahi et al., 2016).
- **Reconnaissance des actions de groupes :** suite à une détection des groupes à partir des formations qu'ils observent, certains travaux se penchent sur la reconnaissance des actions menées en groupe (Ibrahim et al., 2016). La détection et l'étude des comportements de groupes font partie des approches mésoscopiques en analyse des foules, car un groupe est à mi-chemin entre l'individu et la foule (Shao et al., 2018).
- **Détection d'anomalies :** souvent considérée comme un sujet à part entière ou couplée avec un autre sous-sujet de l'analyse des foules, l'on peut faire de la détection d'anomalies pour n'importe quelle tâche de l'analyse des foules (Zhou et al., 2018).

L'analyse des foules peut recourir à l'extraction manuelle d'un certain nombre d'indices visuels (Zhan et al., 2008; Lamba et Nain, 2017; Grant et Flynn, 2017; Li et al., 2015). Cette tâche difficile, et sujette à un certain nombre d'omissions, peut être déléguée aux réseaux de neurones profonds qui sont souvent capables de mieux repérer les indices visuels significatifs (Tripathi et al., 2018).

2.2 Le jeu de données Crowd-11

Créé par une équipe du CEA-LIST (Dupont et al., 2017), ce jeu de données récent et totalement annoté, contient plus de 6000 séquences vidéo. Les séquences vidéo disposent de résolutions variables allant de 220×400 à 700×1250, et proviennent à la base d'une multitude de sources pré-existantes. Les vidéos sont classées en 11 catégories.

Dans ce qui suit, nous décrivons les comportements correspondants aux 11 classes contenues dans le jeu de données Crowd-11 :

0. **Gas Free :** Individus marchant dans toutes les directions sans rencontrer d'obstacles.
1. **Gas Jammed :** Foule congestionnée.
2. **Laminar Flow :** Individus marchant dans une seule direction.
3. **Turbulent Flow :** Foule marchant dans une seule direction perturbée par un individu marchant à contresens.
4. **Crossing Flows :** Deux foules qui se croisent.
5. **Merging Flows :** Deux foules qui convergent.
6. **Diverging Flow :** Une foule qui se subdivisent en deux foules.
7. **Static Calm :** Une foule d'individus statiques et calmes.
8. **Static Agitated :** Une foule d'individus statiques et agités.
9. **Interacting Crowd :** Deux foules d'individus qui s'opposent. Cette classe contient des scènes de conflits.
10. **No Crowd :** Aucune présence humaine dans la scène.

Les vidéos proviennent principalement de trois sites d'hébergement de vidéos et qui sont Youtube [1], Pond5 [2], et GettyImages [3].

Le reste provient des jeux de données suivants : UMN SocialForce, AgoraSet, PETS-2009, Violent-Flows, Hockey Fights and Movies, WWW Crowd, CUHK Crowd, et Shanghai WorldExpo'10 Crowd.

La plupart de ces jeux de données sont publiquement disponibles et facilement accessibles. Toutefois, certains ne le sont plus tels que WWW Crowd, CUHK Crowd, et Shanghai WorldExpo'10 Crowd. À cause de cela, nous n'avons pas pu récupérer le jeu de données Crowd-11 dans sa totalité. Ce qui a pu être récupéré représente approximativement 90% du jeu de données initial. Une estimation de la répartition des séquences récupérées par classe permet de constater qu'il n'y a pas eu une perte majeure par rapport au jeu de données initial, comme nous pouvons l'observer dans le tableau 1.

3 Apprentissage par transfert

Le but de l'apprentissage par transfert est de transmettre les connaissances apprises par un modèle à partir d'un jeu de données source vers un jeu de données cible (Pan et Yang, 2010). Dans des travaux récents, l'apprentissage par transfert pour la classification des clips vidéo a été appliqué pour la reconnaissance d'actions dans des scènes individuelles (Carreira et Zisserman, 2017; Tran et al., 2015). Dans cette situation, l'objectif est de transférer les connaissances

1. Youtube : https://www.youtube.com/
2. Pond5 : https://www.pond5.com/
3. GettyImages : https://www.gettyimages.fr/

Étiquette	Nom de la classe	#vidéos (qté originale)	#vidéos récupérées
0	Gas Free	529	477
1	Gas Jammed	520	508
2	Laminar Flow	1304	1189
3	Turbulent Flow	892	862
4	Crossing Flows	763	717
5	Merging Flow	295	267
6	Diverging Flow	184	189
7	Static Calm	737	686
8	Static Agitated	410	351
9	Interacting Crowd	248	153
10	No Crowd	390	370

TAB. 1 – *Tableau comparatif entre le nombre de vidéos récupérées et le nombre de vidéos original par classe pour le jeu de données Crowd-11.*

acquises d'un jeu de données source vers un jeu de données cible appartenant au même domaine. Dupont et al. (2017) a appliqué cette opération en transférant les connaissances qu'un modèle a apprises d'un jeu de données source de reconnaissance d'actions à un jeu de données cible illustrant des mouvements de foule. Afin de surpasser les problèmes liés à l'apprentissage par transfert, en passant d'un domaine à un autre, nous appliquons l'apprentissage par transfert en lançant la procédure d'ajustement sur un nombre important d'époques (entre 30 et 40).

3.1 Architectures implémentées

Nous avons sélectionné trois modèles à affiner de deux architectures : *C3D* et *TwoStream-I3D*. Le choix de l'architecture *TwoStream-I3D* est principalement motivé par les bons résultats obtenus par ses modèles par rapport aux modèles *C3D* lorsqu'ils effectuent la reconnaissance d'actions dans des scènes individuelles à partir des jeux de données UCF-101 et HMDB-51 (Carreira et Zisserman, 2017). L'équipe du CEA ayant obtenu les meilleurs résultats avec l'architecture *C3D*, son choix dans nos expériences est naturel car nous n'avons pas été en mesure de récupérer le jeu de données Crowd-11 dans son intégralité. Un modèle *C3D* pré-entrainé sur Sports-1m a obtenu ses meilleurs résultats en classant les vidéos de Crowd-11 (Dupont et al., 2017). Ce modèle représente donc pour nous le résultat de base à améliorer au cours de nos expériences. Plus de détails sur les architectures implémentées peuvent être trouvés dans ce papier Bendali-Braham et al. (2019).

3.1.1 Réseaux de neurones *3D Convolutional Neural Network*

Nous avons décidé de ré-implémenter une version des réseaux de neurones convolutifs 3D correspondant à l'architecture décrite dans Tran et al. (2015).

Comme nous l'avons déjà mentionné, l'équipe du CEA obtient sa meilleure performance avec *C3D* après avoir pré-entrainé le modèle sur le jeu de données Sports-1m (Karpathy et al., 2014).

3.1.2 Réseaux de neurones *Two-Stream Inflated 3D*

Carreira et Zisserman proposent l'architecture *Two-Stream Inflated 3D Neural Network* (Carreira et Zisserman, 2017). Cette architecture a été utilisée pour apprendre la reconnaissance d'actions dans des scènes individuelles, où elle a obtenu de très bons résultats par rapport à *C3D*. Nous l'utilisons pour apprendre à reconnaître les mouvements de foule.

Carreira et Zisserman ont pré-entrainé un modèle *TwoStream-I3D* sur ImageNet (Deng et al., 2009) et Kinetics (Kay et al., 2017). En testant ce modèle sur les jeux de données UCF-101 et HMDB-51, ils ont considérablement dépassé les performances des modèles *C3D* qui ont été pré-entraînés sur Sports-1m (Carreira et Zisserman, 2017). Dans notre cas, nous avons décidé de transférer les connaissances acquises d'une branche RVB de l'architecture *I3D* sur les jeux de données sources ImageNet et Kinetics vers le jeu de données cible Crowd-11. Nous avons fait la même chose pour le modèle *TwoStream-I3D* en transférant les connaissances apprises de la branche RVB et de la branche flux optique de l'architecture au jeu de données cible. Nous avons extrait le flux optique de chaque clip vidéo en utilisant l'algorithme TV-L1 (Zach et al., 2007).

4 Expérimentations sur Crowd-11

Dans les expériences que nous avons réalisées, nous avons décidé pour chaque architecture d'affiner un modèle pré-entrainé et d'entraîner un modèle à partir de zéro sur Crowd-11. Dans le cas du modèle pré-entraîné *C3D*, le pré-entraînement a été réalisé sur le jeu de données Sports-1m. Dans le cas des modèles *I3D*/*TwoStream-I3D*, le pré-entraînement a été effectué sur ImageNet, puis sur la version RVB de Kinetics pour la branche RVB, et la version flux optique de Kinetics pour la branche du flux optique.

En prenant en compte les paramètres d'apprentissages trouvés sur Tran et al. (2015) et Carreira et Zisserman (2017) respectivement pour les modèles *C3D* et *TwoStream-I3D*, nous avons choisi d'appliquer la descente du gradient stochastique (SGD) comme fonction d'optimisation, et avons fixé le taux d'apprentissage initial à $0,003$. La fonction de perte choisie pour ces expériences est l'entropie croisée catégorielle. Afin d'être très proche des hyper-paramètres utilisés pour *C3D* par Dupont et al. (2017), nous avons divisé le taux d'apprentissage par 10 toutes les 4 époques. Cependant, nous n'avons pas reproduit cette opération lors de l'entraînement des modèles *I3D* et *TwoStream-I3D*. Pour ces derniers, nous avons choisi de diviser le taux d'apprentissage par 10 uniquement si la valeur de l'erreur augmente sur l'ensemble de validation. Pendant la phase d'entraînement, le nombre d'époques a été fixé à 40 pour les modèles *C3D* et à 30 pour les autres, afin de maximiser les chances des modèles *C3D* d'obtenir de meilleurs scores. Un modèle est enregistré à la fin de chaque époque. À la fin de la phase d'apprentissage, nous avons choisi de sauvegarder le modèle minimisant la fonction de perte lors de la phase de validation. Lors de l'affinement des modèles, nous avons décidé de ne geler aucune couche des réseaux, car les jeux de données sources sur lesquels nos modèles ont été pré-entraînés diffèrent beaucoup de ceux que nous voulons apprendre. Par conséquent, nous avons décidé de rétropropager la mise à jour des poids des réseaux sur l'ensemble des architectures des réseaux lors des phases d'apprentissage. Contrairement à Dupont et al. nous n'avons pas appliqué de méthodes d'augmentation des données pour entraîner nos modèles. Sachant que l'augmentation des données est une méthode de régularisation, nous voulons voir si nos mo-

Modèle	Condition d'entraînement	Précision
Notre *C3D*	Sans pré-entraînement	31.88%
C3D Dupont et al.	Sans pré-entraînement	46.9%
Notre *C3D*	Pré-entrainé	58.29%
C3D Dupont et al.	Pré-entrainé	61.6%

TAB. 2 – *Comparaison entre notre version de C3D et celle de Dupont et al. (2017)*

Architecture	Condition d'entraînement	Moyenne	Min	Max
I3D	Sans pré-entraînement	47.01%	40%	53.36%
C3D	Sans pré-entraînement	31.88%	28.82%	36.43%
TwoStream-I3D	Sans pré-entraînement	47.85%	43.91%	52.42%
I3D	Pré-entrainé	58.97%	56.33%	60.17%
C3D	Pré-entrainé	58.29%	57.19%	60%
TwoStream-I3D	Pré-entrainé	**68.2%**	**66.01%**	**70.34%**

TAB. 3 – *Précision obtenue à la suite de la validation croisée avec K=5.*

dèles ne souffrent pas d'un sur-apprentissage sur la version basique du jeu de données (Dvornik et al., 2018). Par ailleurs, nous voulons déterminer quelles classes nuisent à l'apprentissage de nos modèles, sans parer à ce problème en utilisant l'augmentation des données. Comme nous comptons tester plusieurs méthodes d'augmentation des données vidéo, nous préférons nous consacrer à ce problème ultérieurement.

4.1 Validation croisée à 5 échantillons

Notre version de Crowd-11 est composée de 1641 scènes. Ces scènes ont été divisées en 5769 clips vidéo. Pour éviter que des échantillons se chevauchent, nous avons décidé de conserver tous les clips d'une même scène dans un même échantillon. Lorsque nous sélectionnons une scène à ajouter à un échantillon, notre sélection fait en sorte de maintenir une similarité approximative des échantillons en termes de nombre de clips par classe. Pour entraîner ou ajuster nos modèles, nous avons divisé le jeu de données en 5 échantillons, et avons décidé d'appliquer la validation croisée 5 fois. Pour chaque itération de la validation croisée, nous avons choisi 3 échantillons pour constituer l'ensemble d'apprentissage, un pour constituer l'ensemble de validation et un dernier pour l'ensemble de test. À chaque itération de la validation croisée, l'ensemble de test change. L'ensemble de validation est choisi de manière aléatoire parmi les 4 échantillons restants.

Comme nous avons appliqué une validation croisée 5 fois pour chacun de nos trois modèles en prenant en compte les deux conditions d'entraînement : l'entraînement à partir de zéro, et l'ajustement d'un modèle pré-entraîné ; nous avons lancé 30 procédures d'entraînement [4].

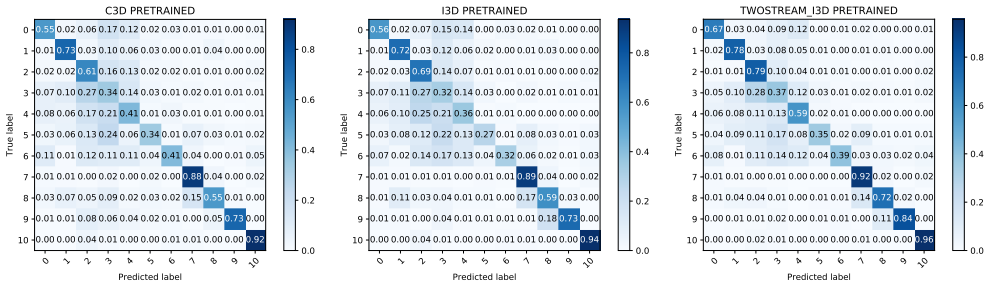

FIG. 1 – *Matrices de confusion globales, des modèles pré-entrainés, calculées à la suite de la validation croisée à 5 échantillons*

4.2 Discussion des résultats obtenus

Selon les résultats affichés sur le tableau 2, nous observons que le modèle *C3D* entrainé à partir de zéro n'est pas aussi performant que le modèle entraîné par Dupont et al. (2017). Cela peut avoir plusieurs raisons : une possible différence entre les hyperparamètres que nous utilisons et ceux qui sont utilisés par les auteurs de Crowd-11 pour l'entraînement de leur modèle, la différence entre nos deux jeux de données, et le fait que nous n'ayons pas recours à l'augmentation des données vidéo. Selon les résultats affichés dans le tableau 3, nous constatons que les modèles *C3D* et *I3D* obtiennent des résultats presque identiques lors de la classification des clips vidéo lors de phase de test. *C3D* n'est dépassé que d'environs $0,6\%$ de précision par le modèle *I3D*. Cette légère différence de performances peut s'expliquer par le fait que l'architecture *C3D* doit entraîner 78 millions de paramètres, tandis que l'architecture *I3D* compte 12 millions de paramètres ainsi qu'une structure profonde. De plus, nous observons que le modèle *TwoStream-I3D* arrive bien à tirer profit du flux optique lors de l'affinement. Cela n'est pas le cas lorsqu'il est entraîné à partir de zéro. Globalement, les modèles *TwoStream-I3D* obtiennent les meilleurs scores.

À partir des matrices de confusion affichées sur la figure 1, nous observons que chaque modèle éprouve des difficultés face aux mêmes classes indicées de 3 à 6, qui sont respectivement : *Turbulent Flow*, *Crossing Flows*, *Converging Flow* et *Diverging Flow*. Nous constatons, également, que les clips appartenant à ces classes, y compris la classe *Laminar Flow*, sont fréquemment confondus. Alors que la classe *Laminar Flow* n'est pas une grande source de confusion, car la foule y suit une direction unique, les multiples transitions clés observables dans les quatre autres classes peuvent perturber la décision du classifieur. Par exemple, nous observons que la classe *Merging Flow* n'est pas confondue avec la classe *Diverging Flow*, ce qui montre que le classifieur apprend bien à différencier entre ces deux comportements. Cependant, ces deux classes sont fréquemment confondues avec la classe *Crossing Flows*. Lorsqu'une foule se croise avec une autre, des comportements de convergence et de divergence sont observés. De plus, alors que *Crossing Flows* est composée par ≈ 850 clips, les classes *Merging Flow* et *Diverging Flow* sont composées par ≈ 200 clips chacune (comme indiqué dans le tableau 1). Cette situation peut amener deux classes à être englouties par une classe plus globale, telle que la classe *Crossing Flows*.

4. Le code source de ce travail est disponible ici :
https://github.com/MounirB/Crowd-movements-classification

5 Conclusion et perspectives

Dans ce travail, nous avons étudié la capacité du réseau *TwoStream Inflated 3D* à tirer profit de son pré-entraînement sur les jeux de données ImageNet et Kinetics pour la classification des comportements de foule sur le jeu de données Crowd-11. Après avoir transféré les connaissances apprises des jeux de données sources vers le jeu de données cible, le modèle produit surpasse l'état-de-l'art, sur Crowd-11, avec une marge conséquente de $\approx 10\%$ de précision. Cependant, du fait du score qu'il a obtenu, le classifieur ne peut pas être, pour l'instant, considéré comme un outil de classification précis pour la gestion des mouvements de foule. Sur la base des résultats obtenus, nous avons l'intention de voir dans quelle mesure nous pouvons les améliorer en testant les méthodes suivantes :

- Appliquer l'augmentation des données vidéo ;
- Remédier aux classes défectueuses du jeu de données Crowd-11 en leur ajoutant des clips vidéo ;
- Tester des modèles issus des architectures *Temporal 3D ConvNets (T3D)* (Diba et al., 2017) et *ActionVLAD* (Girdhar et al., 2017), car les modèles de ces architectures obtiennent des scores supérieurs à 90% de précision sur les jeux de données UCF-101 et HMDB-51 ;
- Modifier l'architecture *Inflated 3D* via :
  - L'ajout de nouveaux modules Inception ;
  - L'hybridation de l'architecture *I3D* avec l'une des deux architectures *T3D* ou *ActionVLAD*.
- Prendre en compte des entrées d'une étape de prétraitement, comme l'extraction des trajectoires denses **(iDT)** (Wang et Schmid, 2013), avant de procéder à l'entraînement des modèles.

Remerciements

Les auteurs tiennent à remercier NVIDIA Corporation pour nous avoir fourni des GPUs et le Mésocentre de Strasbourg pour leur avoir permis de mener des calculs sur le cluster de GPUs. Ce travail a été soutenu par le projet ANR OPMoPS (subvention ANR-16-SEBM-0004) financé par l'Agence nationale de la recherche.

Références

Alahi, A., K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, et S. Savarese (2016). Social lstm : Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 961–971.

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

Carreira, J. et A. Zisserman (2017). Quo vadis, action recognition ? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308.

Coşar, S., G. Donatiello, V. Bogorny, C. Garate, L. O. Alvares, et F. Brémond (2016). Toward abnormal trajectory and event detection in video surveillance. *IEEE Transactions on Circuits and Systems for Video Technology 27*(3), 683–695.

Deng, J., W. Dong, R. Socher, L.-J. Li, K. Li, et L. Fei-Fei (2009). Imagenet : A large-scale hierarchical image database. In *IEEE conference on computer vision and pattern recognition*, pp. 248–255.

Diba, A., M. Fayyaz, V. Sharma, A. H. Karami, M. M. Arzani, R. Yousefzadeh, et L. Van Gool (2017). Temporal 3d convnets : New architecture and transfer learning for video classification. *ArXiv*.

Drews, P., J. Quintas, J. Dias, M. Andersson, J. Nygårds, et J. Rydell (2010). Crowd behavior analysis under cameras network fusion using probabilistic methods. In *International Conference on Information Fusion*, pp. 1–8.

Dupont, C., L. Tobias, et B. Luvison (2017). Crowd-11 : A dataset for fine grained crowd behaviour analysis. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Volume 2017-July, Honolulu, United States, pp. 2184–2191.

Dvornik, N., J. Mairal, et C. Schmid (2018). On the importance of visual context for data augmentation in scene understanding. *ArXiv*.

Girdhar, R., D. Ramanan, A. Gupta, J. Sivic, et B. Russell (2017). Actionvlad : Learning spatio-temporal aggregation for action classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 971–980.

Grant, J. M. et P. J. Flynn (2017). Crowd scene understanding from video : a survey. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM) 13*(2), 19.

Ibrahim, M. S., S. Muralidharan, Z. Deng, A. Vahdat, et G. Mori (2016). A hierarchical deep temporal model for group activity recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1971–1980.

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

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

Kritter, J., M. Brévilliers, J. Lepagnot, et L. Idoumghar (2019). On the optimal placement of cameras for surveillance and the underlying set cover problem. *Applied Soft Computing 74*, 133 – 153.

Lamba, S. et N. Nain (2017). Crowd monitoring and classification : a survey. In *Advances in Computer and Computational Sciences*, pp. 21–31. Springer.

Li, T., H. Chang, M. Wang, B. Ni, R. Hong, et S. Yan (2015). Crowded scene analysis : A survey. *IEEE transactions on circuits and systems for video technology 25*(3), 367–386.

Lu, W., X. Wei, W. Xing, et W. Liu (2017). Trajectory-based motion pattern analysis of crowds. *Neurocomputing 247*, 213–223.

Pan, S. J. et Q. Yang (2010). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering 22*(10), 1345–1359.

Porikli, F., F. Bremond, S. L. Dockstader, J. Ferryman, A. Hoogs, B. C. Lovell, S. Pankanti, B. Rinner, P. Tu, et P. L. Venetianer (2013). Video surveillance : past, present, and now the future [dsp forum]. *IEEE Signal Processing Magazine 30*(3), 190–198.

Ranjan, V., H. Le, et M. Hoai (2018). Iterative crowd counting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 270–285.

Shao, J., N. Dong, et Q. Zhao (2018). A real-time algorithm for small group detection in medium density crowds. *Pattern Recognition and Image Analysis 28*(2), 282–287.

Solera, F., S. Calderara, et R. Cucchiara (2015). Socially constrained structural learning for groups detection in crowd. *IEEE transactions on pattern analysis and machine intelligence 38*(5), 995–1008.

Sultani, W., C. Chen, et M. Shah (2018). Real-world anomaly detection in surveillance videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6479–6488.

Tran, D., L. Bourdev, R. Fergus, L. Torresani, et M. Paluri (2015). Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 4489–4497.

Tripathi, G., K. Singh, et D. K. Vishwakarma (2018). Convolutional neural networks for crowd behaviour analysis : a survey. *The Visual Computer*, 1–24.

Wang, H. et C. Schmid (2013). Action recognition with improved trajectories. In *Proceedings of the IEEE international conference on computer vision*, pp. 3551–3558.

Xu, X., D. Zhang, et H. Zheng (2017). Crowd density estimation of scenic spots based on multifeature ensemble learning. *Journal of Electrical and Computer Engineering 2017*.

Zach, C., T. Pock, et H. Bischof (2007). A duality based approach for realtime tv-l1 optical flow. In *Pattern Recognition*, pp. 214–223.

Zhan, B., D. N. Monekosso, P. Remagnino, S. A. Velastin, et L. Q. Xu (2008). Crowd analysis : A survey. *Machine Vision and Applications 19*(5-6), 345–357.

Zhou, P., Q. Ding, H. Luo, et X. Hou (2018). Violence detection in surveillance video using low-level features. *PLoS one 13*(10), e0203668.

Summary

The automatic recognition of a crowd movement captured by a CCTV camera can be of considerable help to security forces whose mission is to ensure the safety of people on the public area. In this context, we propose to fine-tune a model from the TwoStream Inflated 3D architecture, pre-trained on the ImageNet and the Kinetics source datasets, to classify video sequences of crowd movements from the Crowd-11 target dataset. The evaluation of our model demonstrates its superiority over the state-of-the-art in terms of classification accuracy.