GRAPHIC - GRAPH-BASED REPRESENTATION FOR ANALYZING PEOPLE'S HIGH-LEVEL INTERACTIONS IN CROWDS

Francesco Longobardi and Daniel Riccio

University of Naples Federico II
via Claudio 21, 80125, Naples (Italy)
{francesco.longobardi3,daniel.riccio}@unina.it

2024 IEEE International Conference on Image Processing (ICIP) | 979-8-3503-4939-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICIP51287.2024.10647770

ABSTRACT

The need for automated systems to aid law enforcement during densely packed events arises from the inherent danger of large crowds, evidenced by historical instances of stampedes and crushes. Existing methods vary from basic crowd statistics extraction to detailed anomaly detection in behavior classification, but often focus on single, pre-segmented scenes. Our work addresses classifying crowd behaviors in environments where multiple behaviors coexist within a single scene, defined as a multi-class crowd motion characterization challenge. We use a microscopic approach for scenes captured by drones at varying altitudes, without prior manipulation. This approach combines graph-based representations of individuals and flow images, facilitating classification of diverse crowd behaviors in unsegmented scenes. Tested on a public dataset, our method shows promising results in analyzing complex crowd dynamics.

***Index Terms*—** Crowd behaviour classification, Graph neural networks, Drone video analysis

1. INTRODUCTION

In recent years, due to the overpopulation many cities are experiencing, there has been a trend of trying to focus on important dangers related to mass social gatherings such as concerts, strikes, parades, political demonstrations etc. A prime example of these risks can be found in [1], where Bauckhage analyzed the stampede that occurred in 2010 during a musical event in Duisburg, Germany where dozens of people tragically lost their lives. This is not an isolated case as many more tragedies of similar proportions and characteristics have happened, even in the near past, such as the Seoul October 2022 crowd crush which resulted in a massive casualty rate during the Halloween festival or the 2017 Turin stampede where three people died in Piazza San Carlo as a result of the incident. A first approach to preventing such cases and mitigating the inherent risks of densely gathering multiple people, comprise the installation of surveillance cameras operated and monitored by humans. However, these have been shown to be error-prone and inefficient solutions due to the fact that humans posses limited proficiency in monitoring multiple signals [2]. This is especially valid when one has to keep an eye on several people which contribute in multiple independent groups that can have complex interactions among them. These factors have contributed to the emerging research on crowd analysis systems, in order to implement automatic video surveillance systems able to assist security forces during massive gatherings [3]. In the last decade, many methods have been proposed in the literature to address the problem of crowd behavior analysis. As extensively discussed in Section 2, all these methods assume that there is an initial phase of spatial and temporal segmentation of the scene, whose task is to isolate the sequences to be classified. This is particularly unrealistic in real-world applications where an automatic system receives a raw video directly from the input device and shifts elsewhere a very complex problem like the detection and segmentation of flows, to focus on a less challenging aspect like classification. For this reason, the method we propose is structured in such a way as to operate on an entire scene without presupposing any pre-processing phase. To the best of our knowledge, this is a seminal work in this sense and it stands out for its holistic and multi-dimensional approach, integrating spatial and temporal analysis into a single framework. Unlike traditional methods, it uses high-level graphs and strategic division for accurate and detailed classification of crowd behaviors, even in adjacent areas. A key element is the integration of motion flow images, enriching the analysis with dynamic data such as individual and collective movements. Furthermore, our two-stream neural network model processes both graphs and motion flow images, optimizing data integration and enhancing the model's generalization capacity, especially effective in complex and variable scenarios.

We aknowledge financial support from the project PNR MUR project PE0000013-FAIR

2. RELATED WORKS

An early contribution to crowd behavior analysis by Shao et al. focused on identifying groups in crowded scenes. This

work involved the extraction of four key descriptors from each video to train a Support Vector Machine (SVM). This SVM was capable of classifying group states in videos as gaseous, solid, and fluid-like behaviors, further distinguished between pure and impure fluid states, based on the movement patterns and interactions of people within the crowd [4]. The crowd behavior states initially described by Shao et al. were further expanded in a study by Dupont, Tobías, and Luvison. In this study, they introduced the Crowd-11 dataset, a new resource for crowd behavior analysis [5]. This dataset encompasses similar behavior definitions, such as gaseous and fluid-like behaviors, and also introduces static motion patterns. These patterns are categorized as either calm or agitated and include interactions within crowds that lack a defined motion flow. Furthermore, the authors demonstrated the usefulness of these classes for anomaly detection tasks in video analysis.

Building on the crowd behavior analysis frameworks previously mentioned, the method introduced by Su et al. focuses on group state analysis and crowd video classification. This method employs an unsupervised feature extractor using an LSTM autoencoder, enhanced with a coherent regularization term to capture spatio-temporal hidden features and nonlinear behaviors [6]. In addition, the field has seen advancements with neural network-based methods, notably the two-stream convolutional architectures proposed by Bendali-Braham et al. These architectures demonstrate the effectiveness of pre-trained convolutional neural networks on datasets like Crowd-11. Specifically, they fine-tuned a 3D convolutional neural network (C3D) and a two-stream inflated neural network for this dataset, employing a 5-fold cross-validation approach that yielded enhanced accuracy compared to the original methods [7].

Similarly, Wei et al. in [8] proposed the C-BMO crowd model, which conceptualizes a crowd as a triad of Behavior, Mood, and Organization. Their model distinguishes three primary behaviors: heterogeneous, homogeneous, and violent crowds. To analyze these, they utilized a two-stream convolutional neural network comprising a static channel for RGB images and a motion channel for motion maps, both using VGG-16 networks [9]. The resulting feature maps are then merged to make a prediction. Furthering the two-stream approach, [10] introduced a spatio-temporal method divided into a spatial stream for individual frames and a temporal stream for motion flow fields. The predictions from these streams are combined through class score fusion. Recently, the use of Graph Neural Networks in this domain has been advanced by Behera et al. They developed a deep convolutional graph neural network to classify crowds in surveillance videos into structured and unstructured categories [11].

While previous studies such as those using the Crowd-11 or CUHK Crowd datasets have focused on modeling behaviors in pre-segmented scenes, our work diverges by addressing the classification of crowd behaviors in scenarios where multiple behaviors coexist within the same, unsegmented scene. This approach, namely GRAPHIC, is formulated as a multi-class crowd motion characterization problem on scenes that are not manipulated beforehand. GRAPHIC faces the challenge of unknown segmentation of crowd flows and the complexity of identifying behaviors that may only be present in parts of the scene or change over time. The novelty of the method mainly lies in:

The integration of spatial and temporal analysis – we employ top and bottom level graphs within a single framework, enabling more accurate and specific classification of various crowd behaviors, even in adjacent areas.

The incorporation of motion flow Images – GRAPHIC enriches the analysis with dynamic information like individual movements and collective flows in a format easily processed by neural networks.

A two-stream neural network model – we developed a model that simultaneously processes graphs and motion flow images, optimizing data integration and enhancing the model's ability to generalize.

A new benchmark for crowd behavior analysis – by annotating different flows within the same scene in from a publicly available dataset, we provide a novel training and testing benchmark for the research community.

3. THE PROPOSED METHOD

GRAPHIC is designed to classify the behavior of crowds occurring concurrently in different areas of a video scene, without relying on predefined spatial or temporal segmentation. It employs a graph-based representation, built using a bottom-up approach, where each individual in the crowd is treated as a distinct unit with properties like position, direction, and speed. Static features, such as the position, are calculated relative to the upper left corner of the image, while dynamic features, including direction and velocity, are determined considering a sequence of frames. Individuals with similar features are grouped together, moving coherently, and these groups are represented as graphs. Each group is defined by specific features like location, density, and relative displacement within the group, allowing for differentiation between various clusters of people within the scene. This method facilitates the identification of different areas within a scene and the types of interactions occurring among the assembled people.

However, to classify the behaviour of people crowds, static features are not sufficient and dynamic characteristics must be considered, too. Temporal features are measured at the scene level. Thus, the entire scene is partitioned into disjointed tiles of fixed size. Moving within the scene, groups may traverse one or more tiles. Each time a people group crosses a tile, features of the corresponding graph are associated with the tile. Tiles that are not traversed by any group are excluded from further processing, while the others are represented as nodes of a higher-level graph describing the entire

scene. Each node of this scene graph encapsulates features associated with the corresponding tile, while the arcs account for the distance between tiles. In order to also take into account people who move individually, i.e. without constituting any group, but who nevertheless contribute to determining flows in a scene, a motion flow image is also integrated into the classification process. The graph representing the scene together with the flow image feed a graph neural network that assigns each node to a crowd behaviour class. The details of the different steps constituting the pipeline of the proposed method are discussed in more detail below.

3.1. The Bottom-level graph

Since the processed videos are acquired by drones at a variable altitude, velocity and movement features require normalization against drone height. This is done using an automatically estimated scale normalization factor derived from pedestrian trajectories. Assuming pedestrians move at a fairly constant speed over a fixed distance d, this scale is defined as the frame count required for the most linearly moving pedestrian with least velocity variance to cover distance d. Individuals in a frame are represented as nodes with spatial position (x_p, y_p) and estimated direction (d_{px}, d_{py}), velocity v_p, and movement ds_p, computed over k subsequent frames. To group similar behaviors, a k-Nearest Neighbors agglomerative clustering in a 6-dimensional space aggregates all spatio-temporal descriptors. The clustering space for frame f and people set P_f is:

$$\{(x_p^f, y_p^f, d_{px}^f, d_{py}^f, v_p^f, ds_p^f) \mid p \in P_f\} \tag{1}$$

Euclidean distance is the metric for group identification, with parameters number of neighbors $Cl_{neighbors}$ and distance threshold Cl_{th} determining each node's maximum edges and edge formation distance. This approach creates the bottom-level graph for each scene frame.
Bottom-level graphs may have disconnected nodes or unconnected node groups. Thus, for each bottom-level graph (representing a frame), connected components are identified, where each connected component CC represents a group of persons. Groups are singularly analyzed to create collective descriptors for group behavior. For a connected component $CC = \{p_1, \ldots, p_n\}$, descriptors include group density and nearest person distance, as well as an aggregation of the previous features:

- For positional features, a pseudo-centroid c is computed over all members p of the group $c = (\frac{\sum_{p \in CC} x_p}{|CC|}, \frac{\sum_{p \in CC} y_p}{|CC|})$
- The group resultant normalized direction is computed using the parallelogram law $\overrightarrow{d} = \frac{(\sum_{p \in CC} d_{px}, \sum_{p \in CC} d_{py})}{\| \sum_{p \in CC} d_{px}, \sum_{p \in CC} d_{py} \|}$.
- Velocities and movements are averaged among all nodes of the component.
- An estimation of the group's density is computed as $\rho = \frac{\mu_d}{\mu_d + \sigma_d}$ where $\mu_d = \frac{1}{|CC|} \sum_{p \in CC} \frac{d(c, pos_p)}{scale}$ and $\sigma_d = \sqrt{\frac{1}{|CC|} \sum_{p \in CC} (\frac{d(c, pos_p)}{scale} - \mu_d)^2}$, this measure can be beneficial to distinguish cluttered scenes from sparser distributions of people
- The scaled distance from the centroid of the group to the nearest pedestrian is computed using a 1-Nearest Neighbor search
- The number of people in the group $|CC|$ is also taken into account and processed as a node feature

Each node of the resulting graph represents a group CC and encapsulates the features of that group.

3.2. Top-level graphs

In order to take into account the dynamic aspect of a scene a tiling is generated over video frames (see Fig. 1 (a)). Considering the frame resolution $(W \times H)$ a parameter l controls the side of each square. Moreover, in order to limit the computational cost, only tiles involved by people movement are considered, while all others are excluded (see Fig. 1 (b)). Given tiles and a bottom-level graph for each frame of a scene, the goal is to construct a top-level graph that also describes a temporal portion of the video. For this purpose, considering disjoint time spans composed of $F = \{f_i, \ldots, f_{i+k}\}$ frames with $i = 1, k+1, ...$, for each tile t, a matrix $M_{F,t}$ containing the spatio-temporal features of the groups that crossed t in any of the given time span is computed. Specifically, let be g a group in F, the following rule has been implemented:

$$G = \{g \mid g \in f \ \wedge \ \text{centroid(g)} \in t\} = \{g^{(1)}, \ldots, g^{(n)}\} \quad \forall f \in F, \forall t \tag{2}$$

$$M_{F,t} = \begin{pmatrix} d_{g_x^{(1)}} & d_{g_y^{(1)}} & v_{g^{(1)}} & ds_{g^{(1)}} & \rho_{g^{(1)}} & cd_{g^{(1)}} & np_{g^{(1)}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ d_{g_x^{(n)}} & d_{g_y^{(n)}} & v_{g^{(n)}} & ds_{g^{(n)}} & \rho_{g^{(n)}} & cd_{g^{(n)}} & np_{g^{(n)}} \end{pmatrix} \tag{3}$$

where F is a time span consisting of a sequence of frames f_i, with i representing the index of frames in the timeline, g represents a group of individuals in a given frame. Moreover, the columns of the matrix $M_{F,t}$ are: i) x-component of the direction of group g in frame i, ii) y-component of the direction of group g in frame i, iii) velocity of group g in frame i, iv) displacement of group g in frame i, v) density of group g in frame i, vi) closest distance between the centroid of group g and any other individual in frame i, vii) number of people in group g in frame i.

Due to the nature of this aggregation, tiles may have features of different sizes, depending on the number of groups that crossed a tile. However, neural network models are designed to have input of fixed shape. This brings the necessity of embedding different matrices in a fixed-size space. This problem has been tackled by applying the two-dimensional Discrete Cosine Transform to the feature matrix $M_{F,t}$ of a tile. In order to actually extract a fixed-size representation from such a matrix, the first n_c dct coefficients are sampled according to a zig-zag scan, which ensures that low-frequency coefficients (i.e. coefficients resulting from the computation using smaller cosine oscillations) are placed before high-frequency coefficients [12]. If the original number of coefficients is lower than n_c, it is padded with zeros. Each tile t, that has been crossed by at least one group becomes a node that encapsulates the feature vector of the tile. Doing so, all graphs share the same feature shape and can be used to define the input layer of a graph neural network model.

In the video sequences used for training the classification model, the different areas of a scene are labelled with the corresponding crowd behaviour class (see Fig. 1 (c)), so all tiles falling in a labeled region assume the the label of that region (see Fig. 1 (d)). In order to relate information together, edges between tiles are generated based on their adjacency relation, every tile has an edge with its eight surrounding neighbors and their weights are defined as the Spearman correlation between their feature vectors.

Crowd behavior classification is framed as a graph classification problem, with each graph depicting a group of people. In unsegmented scenes, different behaviors might occur in various areas, especially adjacent ones. The scene, partitioned into tiles, leads to adjacent tiles representing different behaviors, connected as nodes by arcs in the same graph, potentially having different labels. To manage the varying behaviors within a graph, a graph splitting process is implemented to maintain low cardinality and uniform labeling across nodes. Specifically, for each top-level graph's connected component, the node with the highest degree is selected as the BFS source node, considering a depth limit δ. This BFS identifies a node set V_s reachable within δ edges, and the corresponding subgraph G_{V_s} is extracted (see Fig. 1 (e)).

3.3. Motion Flow images

In an effort to give broader contextual information to the model without introducing too many approximations by enlarging graphs, the neural network adopted for classification is designed in a two-stream fashion, using 3 channels images representing the flow of pedestrians in every video. Each tile is represented by a single pixel across three channels in a motion flow image.

In particular, the first channel contains the cumulative number of people present in each tile across all frames, the second represents the cumulative time, in number of frames, spent in each square by every pedestrian and the third one contains the cumulative curvature of the paths followed by people. The last information has been extracted considering a path as a sequence of points $s_p = \{(x_p^{(1)}, y_p^{(1)}), \ldots, (x_p^{(n)}, y_p^{(n)})\}$, where, given a tile t, $(x_p^{(i)}, y_p^{(i)}) \in t \; \forall i \in \{1, \ldots, n\}$. Using a circle approximation of the path (like a least squares circle fitting [13]), the curvature for s_p is defined as $\frac{1}{R_{s_p}}$.

When processing a video, given a graph the corresponding patch is extracted finding the node in the graph with the highest degree, said node corresponds logically to a tile which corresponds to a pixel in a motion flow image. Centered in that pixel, the patch can be extracted expanding from its center with an arbitrary number of pixels in every direction, using a parameter to controls its size. Figure 1(f) shows an example of motion flow image generated from a scene of the training set.

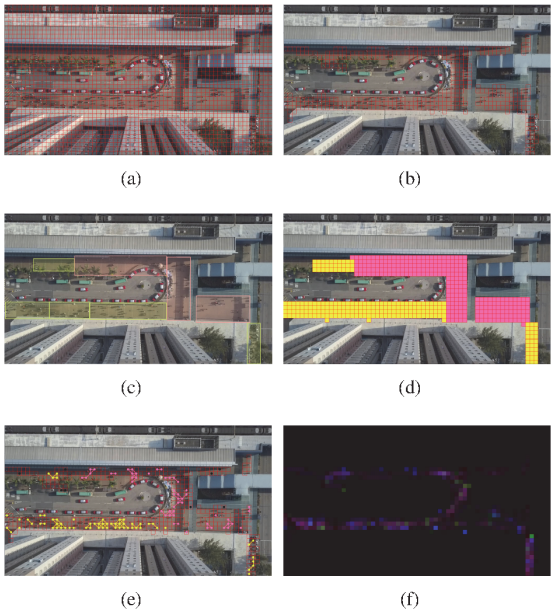

Fig. 1. Some steps of the tile processing pipeline. From top to bottom and from left to right respectively: a) all tiles generated for a frame, b) tiles traversed by people, c) manual annotation of regions with different crowd behaviour, d) tiles labelled according to the regions to which they belong, e) components of the graph after splitting, f) flow image.

4. THE TWO-STREAM GRAPH NEURAL NETWORK

The proposed model is a two-stream GNN which, during the training process, takes as input a batch of graphs, in the form of node and edge features, with their corresponding batch of

patches. The first stream deals with graphs: each batch goes through two graph convolutional layers, each one of these implements the first two steps of the message passing paradigm (aggregate and update), in addition to that, a GraphNorm [14] layer has been introduced after each convolution. The con-

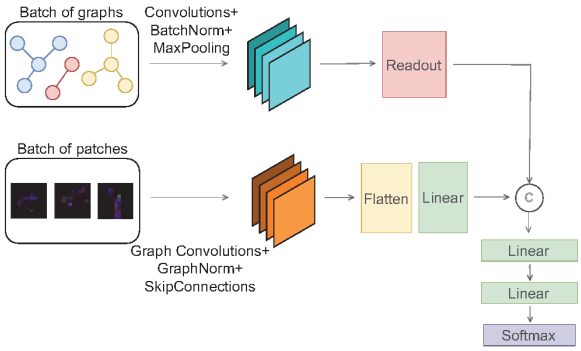

Fig. 2. Illustration of the two-stream graph neural network model used

catenation of the two convolutional layers batched results is then fed into a readout (mean) operator to obtain a batch of graph representations. This approach, which resembles skip-connections, has been proven to improve networks generalisation capabilities [15, 16]. The latter is then concatenated to the batched feature maps obtained from the second stream, which is composed of two convolutional layers followed by max pooling and batch normalization [17] layers. Figure 2 shows a visual summary of the described model.

The model is trained by using the Stochastic Gradient Descent (SGD) with momentum in order to optimise a cross entropy loss using per-class probabilities extracted with the softmax function. The loss function is weighted in order to reduce the interference introduced by well represented classes over under represented ones, that is each class is weighted by the ratio of the total number of samples divided by the product of the number of classes and the number of samples belonging to that class. Moreover, the model weights are updated according to the mini-batch rule, using batches of 512 graphs and flow images.

5. DATASET AND EXPERIMENTS

All the experimens have been conducted on video sequences extracted from the DroneCrowdDataset [18]. This dataset was designed for crowd counting, tracking and localization and was introduced by Wen et al. in 2021. It contains a total of 112 videos shot with different drones (DJI Phantom 4, Phantom 4 pro and Mavic) all at a resolution of 1920×1080@25fps. The featured scenes comprise several scenarios such as markets, parking lots, campuses etc, which are also variably crowded.

For the testing phase, an evaluation dataset has been constructed by removing 20 videos (from a total of 112) from the original data. These scenes have been chosen, after the process of graph generation, to preserve the class distribution across training and test set as much as possible.

The original dataset, having been designed to solve a task which does not comprise behavioral information, only presents annotations in the form of bounding boxes on each person. In order to provide behavioral labels, the dataset has been manually annotated in its entirety using the same class definitions from the Crowd-11 dataset paper [5]. In particular, some classes have been left out such as no crowd and interacting crowd due to the nature of the scene. The labeling task has been carried out using an open-source annotation tool called *Label studio*. After this process, it has been immediately noticed that the dataset's behavioral annotations resulted to be very skewed towards certain classes, while leaving others under-represented. Table 1 illustrates the histogram of the dataset's annotations, particularly showing the very different distributions of class instances among scenes.

Table 1. Annotations distribution throughout the dataset

Static Calm	Gas Jammed	Static Agitated	Merging Flow	Crossing Flow	Turbulent Flow	Gas Free	Diverging Flow	Laminar Flow
51	30	8	18	182	59	184	16	147

5.1. Experimental results

The unique nature of our proposed approach and the specific video annotation it requires mean that many standard datasets for crowd behavior classification are unsuitable for training/testing. This limits comparisons with existing methods, which typically analyze short sequences from various angles and focus on singular behaviors. In testing our method, we have conducted experiments across varying label specificities. The first experiment evaluated binary classification, distinguishing crowds as structured or unstructured. The subsequent two experiments delved into multiclass classification: one assessing three classes based on the presence of none, one, or multiple flows, and the other examining more detailed classes as per dataset annotations.

5.1.1. Results for binary and multiclass classification

A binary classification is defined with respect to two classes that are unstructured and structured crowd. Taking advantage of the similarities of the flows considered in this work and the analysis of similar crowd behaviors conducted by Behera et al. in [11], annotations comprising gas free, gas jammed, merging flow, diverging flow and static, where people move in highly variable directions in a cluttered or sparse way, move from a point to join in another, disperse from a source or show little to no movement at all have been considered unstructured, while behaviors characterised by distinguishable

streams of people such as Laminar, Turbulent and Crossing flows have been labeled as structured. Experiments result in an F-1 measure of 0.71.
The second experiment is based on three classes which describe the number of flows present: class 0) represents no flow with data points coming from gas free, jammed and static where there is no noticeable stream; class 1) contains laminar and turbulent flows which describe behaviors with a single distinguishable stream of people either stable or turbulent due to interference from other people; class 2) holds behaviors with more than one flow like crossing, merging and diverging flows. Experimental results conducted on this type of classification, show that the model is more error-prone on classes with flows while it performs better on the first class. In fact, F-1 scores on the three classes are 0.73, 0.59 and 0.56 respectively, averaging at 0.63. The third experiment aims at testing the predictive capabilities of the model on finer annotations from the dataset. In this section, class indices, in order, correspond to: (0) gas free; (1) gas jammed; (2) laminar flow; (3) turbulent flow; (4) crossing flows; (5) static. As it can be noticed from table 1, the imbalanced nature of the dataset reflects in classes being severely under represented. Due to this reason, merging and diverging flows have been excluded from this experiments, while static agitated behaviours have been merged with static calm ones forming a single static class. Referencing table 2, it is immediately noticeable that turbulent flows are very problematic. In fact, they are often misclassified as laminar flows in scene where disturbances are milder and crossing flows in the other cases.

Table 2. Metrics for second multiclass classification test

Metric	Gas Free	Gas Jammed	Laminar Flow	Turbulent Flow	Crossing Flows	Static	Avg
Accuracy	0.62	0.30	0.69	0.007	0.61	0.66	0.48
Precision	0.70	0.19	0.48	0.074	0.54	0.39	0.40
Recall	0.62	0.29	0.69	0.007	0.61	0.66	0.48
F-1 measure	0.65	0.23	0.57	0.013	0.57	0.49	0.42

5.1.2. Discussion

Comparing this work with previous studies is challenging as the existing approaches only work on pre-segmented scenes. However, in order to compare Shao et al.'s method with GRAPHIC, we divided the DroneCrowdDataset scenes into multiple videos per scene, each representing a different flow according to the corresponding label. This adaptation allowed Shao et al.'s method to be applicable to the types of videos it was designed for. On the contrary, GRAPHIC classifies the original unprocessed videos. Despite not pre-segmenting scenes or flows, our method surpassed [4] in most classes, except turbulent flows, as summarized in Table 3. Results show that incrementing the complexity of the classification, a performance drop can be noticed, especially on less represented classes. Indeed, when validating the approach on a simpler binary task (structured/unstructured crowd), the model is able to achieve an accuracy and F1-measure of 73%, while tests on classes representing the number of distinguishable flows in a stream of people show a noticeable performance drop in accuracy and F1-measure values that reach 63%, with the first class (no flow) being the most reliably predicted. Finally, using a finer labeling, performances particularly drop on turbulent behaviours. The average score obtained reaches an average accuracy of 48% and an average F1-measure of 42% while still retaining similar performances when compared to the other discussed tests with simpler settings, when only considering well-represented classes such as gas free, laminar flow, crossing flows and static (the class resulting from static calm and static agitated behaviors): around $65\pm 4\%$ in accuracy score. Comparing our work with Shao et al.'s one shows a promising starting point and proves that working with raw scenes in crowd behaviour analysis is indeed possible.

Table 3. Comparisons between our method and the one proposed by Shao et al

Metric	Gas Free		Laminar Flow		Turbulent Flow		Crossing Flows	
	Shao et al.	GRAPHIC	Shao et al.	GRAPHIC	Shao et al.	GRAPHIC	Shao et al.	GRAPHIC
Accuracy	0.46	**0.62**	0.33	**0.69**	**0.04**	0.007	0.38	**0.61**
Precision	0.64	**0.70**	0.32	**0.48**	**0.25**	0.074	**0.64**	0.54
Recall	0.46	**0.62**	0.67	**0.69**	**0.17**	0.007	0.50	**0.61**
F1-measure	0.54	**0.65**	0.43	**0.57**	**0.20**	0.013	0.56	**0.57**

6. CONCLUSIONS AND FUTURE WORKS

This work aimed at tackling tasks in the field of crowd behavior analysis in a setting where videos were not trimmed beforehand, this constraint introduces multiple difficulties such as an unknown a-priori flow segmentation and a rather ambiguous definition of the behavior in portion of a scene.
The proposed method starts from clustering persons in each frame of a video sequence based on their estimated features, then extracts people movements in fixed length sequences of frames and ends with constructing a graph based representation of the scene. In order to add more spatio-temporal context, motion flow images built from pedestrian trajectories are also considered. A two-stream network is feed with both representations and provides a local crowd classification with respect to a fixed number of different behaviour classes.
Results are encouraging. In particular, they show that a classification of crowd behaviour is possible without a prior spatio-temporal segmentation of the scene. As expected, the complexity of the classification problem increases proportionally with the number of classes. In particular, the error is greater for classes that are less represented in the data. Future work will focus on ad-hoc data augmentation techniques for balancing the dataset. Given the nature of the annotations and the specific type of input provided to the classification model, this is a quite non-trivial issue.

References

[1] Barbara Krausz Christian Bauckhage. "Loveparade 2010: Automatic video analysis of a crowd disaster". In: *Computer Vision and Image Understanding* (2012).

[2] N. Sulman T. Sanocki D. Goldgof and R. Kasturi. "How effective is human video surveillance performance?" In: *IEEE Internation Conference on Pattern Recognition* (2008).

[3] Mounir Bendali-Braham Jonathan Weber Germain Forestier Lhassane Idoumghar Pierre-Alain Muller. "Recent trends in crowd analysis: A review". In: *Machine Learning with Applications* (2021).

[4] Jing Shao, Chen Change Loy, Siham Tabik, and Xiaogang Wang. "Scene-Independent Group Profiling in Crowd". In: *IEEE Conference on Computer Vision and Pattern Recognition* (2014).

[5] Camille Dupont, Luis Tobías, and Bertrand Luvison. "Crowd-11: A Dataset for Fine Grained Crowd Behaviour Analysis". In: *Conference on Computer Vision and Pattern Recognition Workshops* (2017).

[6] Hang Su, Yinpeng Dong, Jun Zhu, Haibin Ling, and Bo Zhang. "Crowd Scene Understanding with Coherent Recurrent Neural Networks". In: *Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)* (2016).

[7] Mounir Bendali-Braham, Jonathan Weber, Germain Forestier, Lhassane Idoumghar, and Pierre-Alain Muller. "Transfer learning for the classification of video-recorded crowd movements". In: *IEEE 2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA)* (2019).

[8] Xinlei Wei, Junping Du, Zhe Xue, Meiyu Liang, Yue Geng, Xin Xu, and JangMyung Lee. "A very deep two-stream network for crowd type recognition". In: *Neurocomputing* (2020).

[9] Karen Simonyan and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: (2014). DOI: 10.48550/ARXIV.1409.1556. URL: https://arxiv.org/abs/1409.1556.

[10] Habib Ullah, Sultan Daud Khan, Mohib Ullah, Muhammad Uzair, and Faouzi Alaya Cheikh. "Two stream model for crowd video classification". In: *2019 8th European Workshop on Visual Information Processing (EUVIP)* (2019).

[11] Shreetam Behera, Debi Prosad Dogra, Malay Kumar Bandyopadhyay, and Partha Pratim Roy. "Crowd Characterization in Surveillance Videos Using Deep-Graph Convolutional Neural Network". In: *Transactions on cybernetics* (2021).

[12] G.K. Wallace. "The JPEG still picture compression standard". In: *IEEE Transactions on Consumer Electronics* 38.1 (1992), pp. xviii–xxxiv. DOI: 10.1109/30.125072.

[13] N. Chernov and C. Lesort. *Least squares fitting of circles and lines*. 2003. DOI: 10.48550/ARXIV.CS/0301001. URL: https://arxiv.org/abs/cs/0301001.

[14] Tianle Cai, Shengjie Luo, Keyulu Xu, Di He, Tie-Yan Liu, and Liwei Wang. *GraphNorm: A Principled Approach to Accelerating Graph Neural Network Training*. 2020. DOI: 10.48550/ARXIV.2009.03294. URL: https://arxiv.org/abs/2009.03294.

[15] TNima Dehmamy, Albert-László Barabási, and Rose Yu. "Understanding the Representation Power of Graph Neural Networks in Learning Graph Topology". In: *International Conference on Learning Representations (ICLR)* (2017).

[16] Gabriele Corso, Luca Cavalleri, Dominique Beaini, Pietro Liò, and Petar Veličković. "Principal Neighbourhood Aggregation for Graph Nets". In: 33 (2020). Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, pp. 13260–13271. URL: https://proceedings.neurips.cc/paper/2020/file/99cad265a1768cc2dd013f0e740300ae-Paper.pdf.

[17] Sergey Ioffe and Christian Szegedy. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*. 2015. DOI: 10.48550/ARXIV.1502.03167. URL: https://arxiv.org/abs/1502.03167.

[18] Longyin Wen, Dawei Du, Pengfei Zhu, Qinghua Hu, Qilong Wang, Liefeng Bo, and Siwei Lyu. "Detection, Tracking, and Counting Meets Drones in Crowds: A Benchmark". In: *CoRR* abs/2105.02440 (2021). arXiv: 2105.02440. URL: https://arxiv.org/abs/2105.02440.