IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received 10 September 2023, accepted 16 October 2023, date of publication 25 October 2023, date of current version 1 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3327463

RESEARCH ARTICLE

Motion Pattern-Based Scene Classification Using Adaptive Synthetic Oversampling and Fully Connected Deep Neural Network

MOHAMMED SULTAN MOHAMMED[1,2], (Member, IEEE), **AHLAM AL-DHAMARI**[1,2], **WADDAH SAEED**[3], **FATIMA N. AL-ASWADI**[2,4], **SAMI ABDULLA MOHSEN SALEH**[5], AND **M. N. MARSONO**[1]

[1]Faculty of Electrical Engineering, Universiti Teknologi Malaysia, Johor Bahru, Johor 81310, Malaysia
[2]Faculty of Computer Science and Engineering, Hodeidah University, Al Hudaydah, Yemen
[3]School of Computer Science and Informatics, De Montfort University, LE1 9BH Leicester, U.K.
[4]School of Computer Sciences, Universiti Sains Malaysia, Gelugor, Pulau Pinang 11800, Malaysia
[5]School of Electrical and Electronic Engineering, Universiti Sains Malaysia, Nibong Tebal, Pulau Pinang 14300, Malaysia

Corresponding authors: Ahlam Al-Dhamari (ahlam.aldhamari@hoduniv.net.ye) and M. N. Marsono (mnadzir@utm.my)

This work was supported by the Research Management Centre (RMC), Universiti Teknologi Malaysia (UTM) under the Professional Development Research University grant (R.J130000.7113.06E45).

ABSTRACT Analyzing crowded environments has become an increasingly researched topic in computer vision community, largely due to its myriad practical applications, including enhanced video surveillance systems and the estimation of crowd density in specific settings. This paper presents a comprehensive method for progressing the study of crowd dynamics and behavioral analysis, specifically focusing on the classification of movement patterns. We introduce a specialized neural network-based classifier explicitly designed for the accurate categorization of various crowd scenes. This classifier fills a unique niche in the existing literature by offering robust and adaptive classification capabilities. To optimize the performance of our model, we conduct an in-depth analysis of loss functions commonly employed in multi-class classification tasks. Our study encompasses four widely used loss functions: Focal Loss, Huber Loss, Cross-Entropy Loss, and Multi-Margin Loss. Based on empirical findings, we introduce a Joint Loss function that combines the strengths of Cross-Entropy and Multi-Margin Loss, outperforming existing methods across key performance metrics such as accuracy, precision, recall, and F1-score. Furthermore, we address the critical challenge of class imbalance in motion patterns within crowd scenes. To this end, we perform a comprehensive comparative study of two leading oversampling techniques: the synthetic minority oversampling technique (SMOTE) and adaptive synthetic sampling (ADASYN). Our results indicate that ADASYN is superior at enhancing classification performance. This approach not only mitigates the issue of class imbalance but also provides robust empirical validation for our proposed method. Finally, we subject our model to a rigorous evaluation using the Collective Motion Database, facilitating a comprehensive comparison with existing state-of-the-art techniques. This evaluation confirms the effectiveness of our model and aligns it with established paradigms in the field.

INDEX TERMS Computer vision, crowd analysis, collective behavior, cross-validation, ADASYN, SMOTE.

I. INTRODUCTION

Technological advances, coupled with sustained growth in the human population, have heightened the demand for the development of efficient automated video surveillance-based technologies [1], [2]. In the domain of crowd video surveillance, extensive research is currently being conducted across multiple critical dimensions. Crowd behaviour analysis [3], [4] examines the movement and interactions within crowds to improve safety. Crowd density estimation and crowd

The associate editor coordinating the review of this manuscript and approving it for publication was Ting Wang.

counting [5] are focused on assessing the number of people and the compactness of a crowd, which have applications in public safety and event management. Crowd anomaly detection [6] identifies unusual patterns that may indicate danger or suspicious activities, while group detection [7] explores the formation and behavior of smaller groups within the crowd. These research areas contribute to developing sophisticated monitoring and management systems with broad applications in urban planning, law enforcement, and emergency response.

In most of the leading research areas related to crowd analysis, the effectiveness of the proposed methods depends on the characteristics and classification of the crowded scene being studied. This relationship underscores a subtle complexity where a method demonstrating robust performance in a particular scene may not necessarily replicate that success when applied to a disparate scenario. This lack of generalizability becomes particularly pronounced when the dynamics of the scene undergo a transformation. The multifaceted nature of crowd behavior, encompassing variables such as density, movement patterns, and group interactions, necessitates a comprehensive understanding and adaptive approach to ensure the applicability of the methods across various contexts. Therefore, the development of versatile techniques that can accommodate the diverse and evolving dynamics of different crowd scenarios remains a critical and ongoing challenge in the field.

Understanding variations in scene dynamics involves a study of object motion in crowded environments. These objects, in the context of crowd analysis, could be automobiles or humans. It is possible to spot patterns and trends that can shed light on the scene's underlying dynamics by focusing attention on how objects behave and communicate. Analyzing historical and current motion patterns allows for predicting future movements in a given scene, a capability essential for applications like traffic management, crowd control, and public safety [8]. Motion patterns in crowded scenes can be categorized into three primary types: structured, unstructured, and semi-structured. In structured scenes, movements are orderly and predictable. For instance, in a well-regulated traffic system, vehicles adhere to lanes and obey traffic signals.

Likewise, in controlled public spaces, people tend to follow designated paths or queues. Such consistency simplifies the analysis and prediction of future movements. In contrast, unstructured scenes feature chaotic or random motion patterns, lacking clear rules for object movement. A bustling market street, where people and vehicles move freely without designated paths, serves as an example. The analysis of unstructured scenes is usually more complex, necessitating advanced methods to comprehend the underlying dynamics. Many real-world scenes do not strictly conform to either structured or unstructured categories but rather exhibit a mix of both. These are termed semi-structured scenes, displaying a combination of order and chaos in object movements.

In the area of crowd analysis, identifying the nature of the scene is crucial for multiple reasons. First, understanding scene dynamics is pivotal, as different scenes possess unique underlying dynamics that can be instrumental in predicting and managing crowd behavior. Second, discerning the type of scene enables researchers and practitioners to tailor analytical models that are precisely aligned with the characteristics of the given scene. Finally, the adaptive management of crowd scenarios is another critical consideration. Real-world scenes are often dynamic and may evolve over time; therefore, the ability to recognize and adapt to these changes is vital for effective and responsive management. However, identifying the scene type provides intermediate knowledge that bridges low-level raw data (such as individual object positions) and high-level insights (like overall crowd behavior trends). Continuous monitoring and re-assessment of the scene type at regular intervals are of paramount significance. Crowds can transition from stable to unstable states, underscoring the importance of periodic reassessments to track these fluctuations. Furthermore, through ongoing evaluation of the scene, authorities or systems can refine their strategies to effectively respond to evolving conditions. The practice of continuous monitoring fosters improved safety measures and enhanced crowd management, particularly in dynamic or rapidly changing environments.

In this article, a novel methodology is introduced with the aim of classifying a specific crowded scene into one of three distinct classifications: structured, semi-structured, or unstructured. This categorization is grounded in the analysis of motion patterns within the scene, which are manifested in the form of trajectories. The contributions of this work are manifold and can be summarized as follows:

- **Development of a specialized classifier for crowd scene categorization**: We introduce a robust and innovative classifier that leverages a fully connected deep neural network. This classifier is tailored to adeptly categorize various types of crowd scenes, fulfilling a distinct niche in the field of crowd dynamics and behavior analysis.
- **In-depth analysis of loss functions in multi-class classification**: This paper presents an exhaustive investigation into the role of the optimization landscape, with a particular focus on the significance of different loss functions. We examine four eminent loss functions—Focal Loss, Huber Loss, Cross-Entropy Loss, and Multi-Margin Loss—and evaluate their efficacy in multi-class classification scenarios. Our empirical results demonstrate that Cross-Entropy and Multi-Margin Loss exhibit superior performance attributes over their counterparts. To take advantage of this, we formulate a Joint Loss function that amalgamates the strengths of these two loss functions. This approach surpasses existing methods in key performance indicators such as accuracy, precision, recall, and F1-score.

- **Comparative study of oversampling techniques in pattern motion classification**: We address the prevalent issue of class imbalance in the motion patterns observed within crowd scenes, leveraging advanced oversampling techniques to rectify this challenge. Specifically, we conduct an exhaustive comparative analysis of two leading oversampling methods: SMOTE and ADASYN. These methodologies are rigorously evaluated within the specialized model of pattern motion classification. Our empirical analysis unequivocally demonstrates that ADASYN surpasses SMOTE in enhancing classification performance across various metrics. Importantly, this multifaceted approach serves a dual purpose: it effectively mitigates the problem of class imbalance and furnishes compelling empirical evidence that validates the efficacy of our proposed method.
- **Rigorous model evaluation using the collective motion database**: A comprehensive evaluation was conducted using the Collective Motion Database. This meticulous assessment validates the effectiveness of our proposed model and facilitates a detailed comparison with existing state-of-the-art methodologies based on collectiveness measures. Through this analysis, we shed light on the unique aspects of our model and establish its congruence with established theoretical frameworks in the domain.

The paper is structured as follows: Section II extensively reviews relevant literature and prior research. Section III presents the paper's core, delving into the methodology used for crowd scene classification. Section IV details the results obtained, showcasing the practical application and evaluation of the proposed method. Concluding remarks and insights into potential future developments and research directions are presented in Section V.

II. RELATED WORK

Despite the abundance of research conducted on motion pattern-based crowd analysis [3], [9], [10], there exists only a limited body of work specifically targeting the classification of a scene into the three distinct categories of interest: structured, semi-structured, and unstructured [7], [11]. Most existing research has explored various aspects of motion patterns without delving into the precise categorization that distinguishes between these three fundamental types of crowd behavior. In [12], Zhou et al. proposed a novel descriptor aimed at quantifying the level of collectiveness within crowded scenes. This descriptor focuses on evaluating the degree to which individuals within a group engage in collective motion. The study introduces three distinguished classifications of collectiveness metrics: those representing high, moderate, and low levels of collectiveness.

Derived from [3] and [12], an elevated level of collectiveness typifies structured scenes, while scenes with low collectiveness are commonly observed in unstructured scenarios. Semi-structured scenes, on the other hand, exhibit a moderate degree of collectiveness. Following [3], since the concepts of elevated, moderate, and low collectiveness correspond respectively with the attributes of (structured, semi-structured, and unstructured) crowded scenes, we proceed to compare our method with existing approaches for classifying crowd scenes that are based on the concept of collectiveness.

In [12], the term 'Collective Merging' was designed to identify patterns of collective movement amidst random outlier motions. They conducted rigorous tests to validate the robustness and efficacy of their proposed collectiveness metric, initially applying it to systems of self-propelled particles. Their results indicated a strong correlation with human intuitions regarding collective movement. Additional empirical studies involving video footage of walking crowds and bacterial colonies, underscored the descriptor's broader applicability, suggesting its potential utility in both video-based surveillance and academic research. As an integral aspect of this study, Zhou et al. presented the Collective Motion Dataset (CMD) to assess the effectiveness of their descriptor. We similarly leveraged this extensive dataset to validate our newly proposed method.

Shao et al. [13] conducted a comprehensive analysis of the essential and collective characteristics of groups present across diverse crowd structures. These characteristics are inspired by socio-psychological research and play a critical role in understanding crowded scenarios. The authors introduce a reliable algorithm for detecting groups that are informed by discovering collective transitions. Adopting a graph-centric perspective, they develop an extensive array of visual descriptors that capture various aspects of group properties, such as geometric configuration, topological arrangement, and collective intensity. These descriptors prove to be highly effective for a range of applications, including monitoring crowd dynamics, categorizing crowd videos, and retrieving specific crowd videos.

In [14], a novel methodology for measuring collectiveness was introduced. This approach includes the development of a point selection technique capable of isolating the most relevant tracked feature points to symbolize individuals within a crowd. Additionally, a stability descriptor is formulated to assess the consistency of an individual's interactions with others. By concurrently examining both spatial and temporal indicators within the crowd, the method enables the quantitative computation of a collectiveness metric rooted in the topological connections among individuals.

Li et al. [15] studied the measurement and detection of collective motion dynamics. In contrast to conventional approaches that overlook the time-dependent nature of crowd actions, the authors propose using a hidden-state model to characterize individual movements. They then employ a probabilistic similarity assessment technique for comparison. Utilizing the derived similarity metrics, they establish a structure-oriented measure of collective behavior. This allows for the exploration of topological relationships among individuals and provides a means to quantify behavioral consistency at both the individual and overall scene levels.

In [11], a straightforward approach for classifying sequences of moving crowds in videos was presented. Key points identified in the initial frame are monitored throughout the sequence via the optical flow technique. This eliminates the need for tracking every point in the frame, focusing instead on a selected subset. A descriptor is then calculated based on the directional motion of these tracked points. Subsequently, histograms of these motion orientations at the block level are combined to form comprehensive frame-level features. In [7], a feature vector was introduced that utilizes the histogram of angular deviations from average trajectory vectors to categorize crowded scenes as either structured, semi-structured, or unstructured, depending on overarching motion patterns. The methodology put forth comprises values that represent the frequency of each conceivable angular deviation, ranging from 0 degrees to 180 degrees.

Similar to [7] and [11], many of the previously discussed methods calculate a collectiveness value for each individual frame and then take an average across all frames to determine the overall crowd collectiveness of a given video. Such per-frame methodologies tend to yield considerable fluctuations in the measured collectiveness, owing mainly to the ever-changing motion patterns of the trajectory's key points from one frame to the next. Additionally, these methodologies are highly sensitive to the initial parameters of their respective models and are also computationally intensive. In contrast, we introduce a multi-frame strategy that averages trajectory data across a predefined set of frames. This allows for the incorporation of historical motion data and results in a more stable feature vector for quantifying crowd collectiveness, offering an improvement over the traditional frame-by-frame method.

Our method fundamentally diverges from [7] and [11], as it introduces a refined, fully connected neural network model aimed at segregating crowds into three well-defined categories: structured, semi-structured, and unstructured. This classification hinges on the consistency or randomness of the movement behaviors displayed by the assembled entities. In the context of structured crowds, the motion patterns are coherent and exhibit uniformity, implying that either the collective crowd or particular subgroups therein manifest uniform directional and speed attributes. In contrast, unstructured crowds display a gamut of erratic and variable movements, leading to a diverse array of both speed and direction. Semi-structured crowds occupy an intermediate position between the structured and unstructured classifications, adding a layer of complexity to the categorization challenge.

III. CROWD SCENE CLASSIFICATION USING FULLY CONNECTED DEEP NEURAL NETWORK

This section outlines our comprehensive method of crowd classification, employing a multi-step process. Fig. 1 illustrates the overall block diagram of the proposed method. The procedure commences with a large collection of real video clip sequences that capture a variety of crowd scenarios. These clips are then subjected to trajectory data extraction via the generalized Kanade-Lucas-Tomasi (gKLT) algorithm, isolating the movement patterns of individual participants within the crowd. Following this, the histogram of angular deviation features (HADF) is computed to quantify motion patterns.

HADF consolidates separate angular differences into a unified, coherent representation, thereby shedding light on overarching patterns in motion. By concentrating solely on angular features, the proposed method seeks to distill the complexity of motion within a crowded scene into a form that can be quantitatively analyzed and visually interpreted. Computing angular deviations in pairs offer insights into their directional relationships within a given scene. Furthermore, compiling these deviations into a histogram captures an overarching view of movement patterns. This approach aims to realize the scene's dynamics by focusing on particular geometric aspects of motion.

The utilization of the gKLT tracker, the focus on angular deviations, and the aggregation of this information into a histogram collectively contribute to a sophisticated picture of the scene's motion structure. It's a strategy that translates the intricacies of motion within a crowded environment into tangible data and visual patterns, facilitating both analytical exploration and practical application. A detailed exploration of the process for extracting trajectory data and computing the HADF features is presented in subsection III-A.

Prior to training the model, it is necessary to scale the feature vectors to ensure they fall within the range of 0 to 1. This normalization process is crucial for network convergence. Subsequently, the scaled data is divided using 10-fold cross-validation, a technique characteristically used to evaluate the performance of machine learning models. Finally, the dataset is augmented using the ADASYN method to potentially rectify issues related to class imbalance. The ultimate goal of this exhaustive process is to categorize crowds into one of three distinct classes: structured, unstructured, or semi-structured.

The classification task falls under the umbrella of supervised learning methods, aiming to systematically categorize data into classes based on labels selected from a predetermined list of potential options. This assumes that each data pattern is assigned to a single label. Classification can differentiate between two categories, known as binary classification, or multiple categories, referred to as multiclass classification. Classification can be employed in diverse fields where there's an uneven data distribution, including anomaly detection [6], [16], [17], object detection [18], and medical forecasting [19]. Commonly used classification approaches tend to perform optimally when the pattern sizes across groups are roughly equal. This is because these methods aim to maximize accuracy while simultaneously reducing the error rate. When there's a balance in sample sizes, the classifier is less likely to be biased towards a

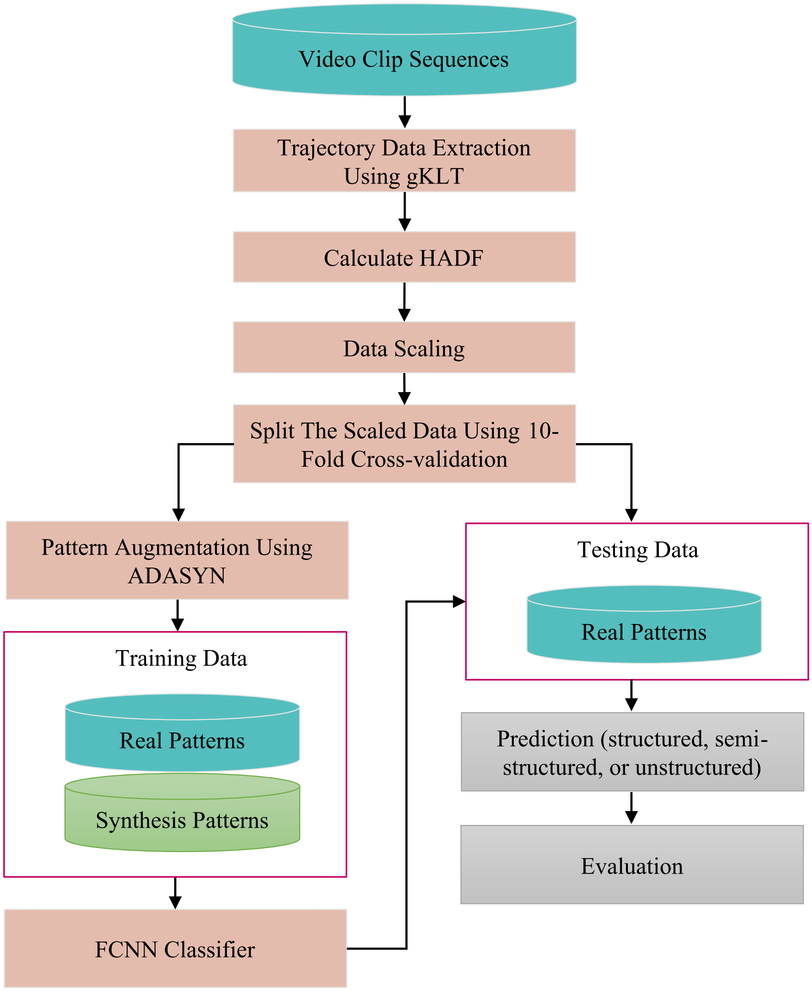
Video Clip Sequences
Trajectory Data Extraction Using gKLT
Calculate HADF
Data Scaling
Split The Scaled Data Using 10-Fold Cross-validation
Pattern Augmentation Using ADASYN
Training Data
Real Patterns
Synthesis Patterns
FCNN Classifier
Testing Data
Real Patterns
Prediction (structured, semi-structured, or unstructured)
Evaluation

FIGURE 1. End-to-end process of the proposed method for crowd scene classification.

particular group, leading to more reliable and consistent results.

Handling imbalanced data presents a significant challenge. In situations where the data is skewed, the minority class, which often represents the category of primary interest, tends to suffer from higher misclassification costs. This is because classifiers can become biased towards the majority group, potentially overlooking or inaccurately classifying instances from the less-represented category. Ensuring accurate classification in these circumstances is crucial to avoiding misleading results [20], [21]. The characteristics of the dataset can exacerbate the issue of having insufficient training observations, which in turn can lead to overfitting. When a model is overfitted, it becomes too closely tailored to the training data, potentially compromising its ability to generalize and perform well on new, unseen data [22], [23].

Achieving unbiased classification in the presence of imbalanced data can be accomplished by adjusting the distribution of the minority class. This can be done through various methods, such as preprocessing techniques or employing resampling strategies. The methods employed are designed to achieve a balanced class representation, thereby improving the model's predictive accuracy across various categories. Therefore, to mitigate the impact of highly imbalanced data and avoid model overfitting, we augmented the proposed model by recalibrating the minority class distribution using an ADASYN-based resampling method. This adjustment enhances the model's predictive capabilities by rectifying the class imbalance, resulting in outcomes that are both more dependable and widely applicable. A detailed explanation of the ADASYN method is provided in subsection III-B.

A. EXTRACT TRAJECTORIES AND OBTAIN HADF

Relying on the following state-of-the-art studies [7], [12], [24], the movement patterns of crowds are deemed to display a noteworthy degree of collective or organized behavior when a substantial proportion of individuals within the crowd navigate in a unified direction along a shared path. This organized motion frequently occurs when participants coordinate their movements and synchronize their activities, resulting in a unified and harmonious flow of the crowd as a collective entity. Capitalizing on this intrinsic aspect

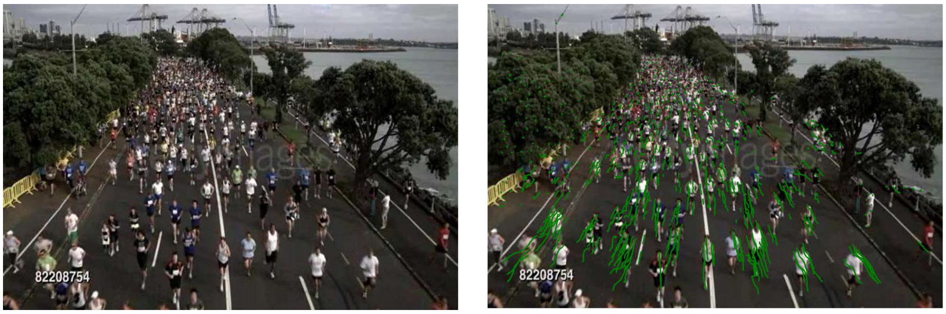

FIGURE 2. An example of a structured scene with its motion patterns. The trajectories elucidate the movement patterns within the crowd during a specific timeframe.

of crowd dynamics, we present an innovative method for classifying video clips of crowds into structured, semi-structured, or unstructured categories. To accomplish this, we introduce an adaptive synthetic sampling fully connected neural model (ADASYN-FCNN). The initial step involves the extraction of individual trajectories from the video data. Subsequently, we compute the histogram of angular deviation features (HADF), as described in [7] and [25]. Below, we outline the process of extracting trajectories and acquiring HADF features.

- **Step 1.** In the initial phase of processing an input video featuring a crowd, capturing the crowd's movement is paramount. To accomplish this, we employ the gKLT tracker, a robust and computationally efficient algorithm for feature tracking, as proposed by Zhou et al. [12]. This tracker identifies and follows key points, specifically corner features, across successive video frames. However, certain variables like occlusion and variations in lighting can result in the generation of errant trajectories characterized by either minimal length or predominantly zero displacement values. To mitigate the impact of such noise, trajectories that do not meet specific empirical thresholds are systematically filtered out. Let $V = \{f_1, f_2, \ldots, f_n\}$ be the set of video frames. For each video frame f_i, identify a set of features $C_i = \{c_{i_1}, c_{i_2}, \ldots, c_{i_m}\}$. Then, use the gKLT tracker to track these features across frames.

$$T_i = gKLT(C_i, f_i, f_{i+1}) \tag{1}$$

where T_i is the set of trajectories for frame f_i to f_{i+1}. Let τ_{length} is an empirical threshold for minimal trajectory length. Then, remove all trajectories that do not meet the following criteria.

$$\overline{T}_i = \{t \in T_i \mid length(t) \geq \tau_{length}\} \tag{2}$$

The outcome comprises a collection of refined trajectories (as shown in Fig. 2), denoted as $\bar{T} = \{\overline{T}_1, \overline{T}_2, \ldots, \bar{T}_{n-1}\}$, adeptly depicting the movement patterns within the crowd video while effectively attenuating the influences stemming from disruptive elements like occlusions and fluctuations in lighting conditions.

- **Step 2**. In the analysis of object movements within a given scene, a sequence of two-dimensional coordinates across consecutive frames serves as a crucial data set. This sequence, known as a trajectory, traces the spatial path of a key-point associated with the object. To effectively ascertain the crowd's movement direction using this trajectory data, an initial step involves the calculation of the mean displacement for each frame within each trajectory.

  Following this, the mean angular orientation, denoted as θ_{ti}, for each trajectory is extracted from the previously computed mean displacement vector. This is achieved by projecting this mean vector onto a unit vector aligned with the horizontal ($x-$axis), as outlined in [25]. In each unique trajectory, the derived average angular orientation effectively captures the directional tendencies of the path's movement over a designated time interval. When the bulk of these averaged vectors within a given scene converge towards a common directional focus, the scene is categorized as being structured.

  On the other hand, if the vectors display a lack of directional consistency and are dispersed in multiple directions, the scene is identified as unstructured. As a result, an in-depth analysis of the orientation value distribution yields a well-defined perspective on the dominant movement patterns exhibited by objects in a specific context. This analytical strategy offers a comprehensive view into collective behaviors and directional proclivities, thus serving as an essential asset for evaluating the global dynamics within the scene [7].
- **Step 3.** Utilizing the orientation values obtained in Step 2, the angular difference or deviation matrix is computed, denoted as A_{dev}. This square matrix has entries $[i, j]$ that represent the angular deviation between the i-th and j-th angles in the data set. The angular difference D between any two angles a and b is determined by the formula: $D(a, b) = \min(2\pi - |a - b|, |a - b|)$. This formulation acknowledges the cyclical nature of angular measurements, ensuring that $D(a, b) = D(b, a)$ and $D(a, b)$ falls within the range of $[0, \pi]$.

- **Step 4.** Deriving a histogram from the A_{dev} for each crowd scene acts as a potent instrument for elucidating the global relationships among trajectory vectors, especially concerning their angular orientations. A histogram featuring a marked peak close to zero degrees of angular deviation suggests that the majority of trajectory vectors are aligning in a consistent direction. In other words, this clustering near a zero-angle deviation in the histogram essentially indicates that the movement directionality within the scene is largely organized.

Fig. 3 displays three scene examples, each representing structured, semi-structured, and unstructured environments, respectively. Each scene is accompanied by its histogram of angular orientation and deviation matrix. For the structured scene, the histogram is highly peaked around specific angles, indicating that the crowd in this scene is primarily oriented in particular directions. This is characteristic of structured crowds, where people tend to face the same or similar directions. For the semi-structured scene, the histogram has multiple peaks but is less sharply peaked than in the structured scene. This suggests that while there are predominant directions, there is also some variability in the crowd's orientation. This is typical of semi-structured crowds, where people generally face a few common directions, though not as uniformly as in structured crowds. For the unstructured scene, the histogram is more evenly distributed, indicating a greater variety of orientations. This is indicative of an unstructured crowd, where people are oriented in various directions without any clear pattern.

B. PATTERN AUGMENTATION

ADASYN [26] is an advanced oversampling technique designed to address the class imbalance in machine learning datasets, particularly for classification tasks. Class imbalance occurs when the number of instances in one class significantly outweighs the instances in another class, which can lead to biased model performance. ADASYN aims to alleviate the impact of class imbalance by generating synthetic samples for the minority class while focusing on regions where the class distribution is dense.

Unlike traditional oversampling techniques that generate synthetic samples uniformly across the feature space, ADASYN adapts its synthetic sample generation based on the distribution of the data. Here's a step-by-step explanation of how ADASYN works: Given training data patterns $HADF_{training} = \{x_i, y_i\}_{i=1}^{m}$, where x_i is a pattern vector with n-dimensional columns and m is the number of patterns, in our case, the $n = 180$, and $y_i \in \{0, 1, 2\}$ for the three defined classes $\{$(0): Unstructured, (1): Semi-structured, (2): Structured$\}$. Moreover, choose the ω and k, where ω is the desired level of class balance, and k is the number of nearest neighbors.

1) Let n_{maj} and n_{min} denote the number of patterns of the majority and minority classes, respectively. Next, compute $\mathcal{G}$, a quantity that signifies the difference between the quantity of patterns in the majority class and the quantity of patterns in the minority class, weighted by the parameter ω that belongs to the interval **[0,1]**, as in Eq. (3):

$$\mathcal{G} = (n_{maj} - n_{min}) \times \omega \tag{3}$$

2) Extract the minority set M, where $M \in \{x\}_i^{n_{min}}$.
3) Compute the Euclidean distance between the vector x_i and all the components of M to acquire the $k-$nearest neighbors of x_i. Let E_{ik} indicates the set of the obtained $k-$nearest neighbors.

$$d(x_i, x_j) = \sqrt{\sum_{l=1}(x_{i,l} - x_{j,l})^2} \tag{4}$$

where $x_{i,l}$ and $x_{j,l}$ are the $l-$th components of the points x_i and x_j, respectively.
4) Obtain the ratio a_i, based on ρ_i, which refers to the number of patterns in the $k-$nearest neighbors' area of x_i.

$$a_i = \frac{\rho_i}{k}, i = \{1, \ldots .n_{min}\} \tag{5}$$

Then, normalize a_i to $\hat{a}_i$ using Eq. (6), where $\sum_{i=1}^{n_{min}} \hat{a}_i = 1$.

$$\hat{a}_i = \frac{a_i}{\sum_{i=1}^{n_{min}} \hat{a}_i} \tag{6}$$

5) Compute the synthetic patterns g_i required for each minority class pattern based on the following Eq. (7) below:

$$g_i = \hat{a}_i \times \mathcal{G} \tag{7}$$

6) Randomly select g_i synthetic patterns denoted x_{ij}, $(j = 1, \ldots, g_i)$ from E_{ik} with replacement.
7) For a given $x_{ij}, j = \{1, \ldots g_i\}$, synthesize a new pattern based on Eq. (8), where φ is selected uniformly between 0 and 1 for each x_k.

$$x_s = x_i + \varphi(x_i - x_{ij}) \tag{8}$$

C. FULLY CONNECTED DEEP NEURAL NETWORK

The proposed neural network architecture, termed FCNN (Fully Connected Neural Network), encompasses a structured hierarchy of layers designed for effective motion pattern classification. Comprising an initial input layer with 180 neurons, the network seamlessly progresses through hidden layers tailored to capture intricate data patterns. The excessive hidden layers may lead to prolonged training time and overfitting. To achieve an optimal balance, our FCNN model is composed of four hidden layers. The first hidden layer, with 512 neurons, utilizes the Rectified Linear Unit (ReLU) activation and dropout regularization to enhance model generalization. Subsequently, the 256-neuron layer employs batch normalization for stable training, followed by a 128-neuron layer incorporating further dropout. A 64-neuron layer contributes to dimensionality reduction

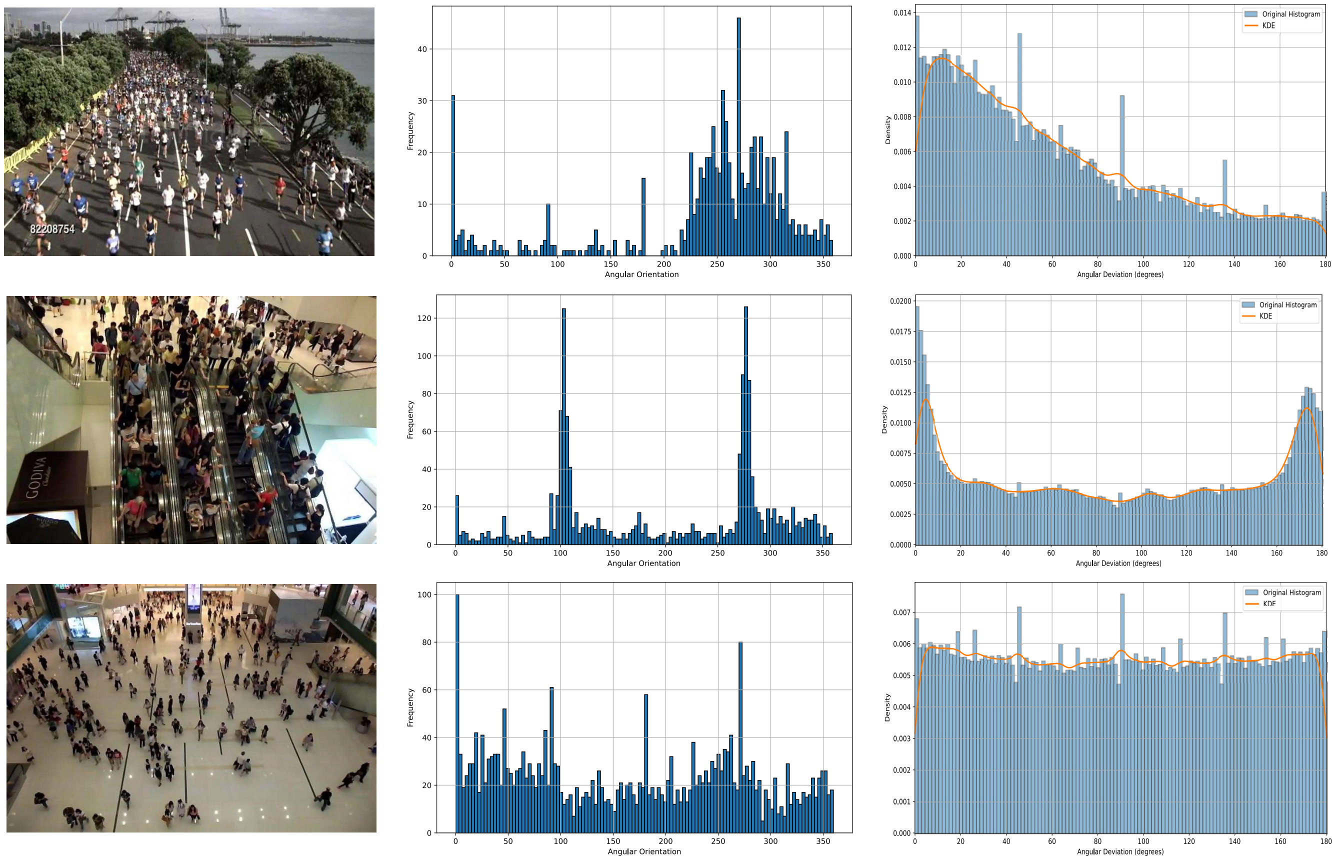

FIGURE 3. The top row displays a structured scene, the middle row features a semi-structured scene, and the bottom row portrays an unstructured scene, each accompanied by histograms illustrating its angular orientation and deviation matrix. The histogram of the deviation matrix is overlaid with a Kernel Density Estimation (KDE) curve. The KDE curve offers a smooth approximation of the data's probability density function.

before culminating in a 3-neuron output layer, supporting classification into three predefined classes. These layers, alongside dropout and batch normalization, synergistically facilitate robust classification performance while mitigating overfitting.

Fig. 4 visualizes the architecture of the proposed FCNN model. This graph represents how data flows through the model's layers. The neuron in each layer are connected to each other neurons by weight and bias as shown in Fig. 4. This process can be expressed as follows:

$$y = \sigma(\sum_{i=1}^{n} w_i x_i + b_i) \tag{9}$$

where x_i represent the input vectors, y represents the output of the neuron, w_i and b_i denote the weight of each input and bias of the neuron, respectively, and σ represents the activation function, which makes the neuron generate nonlinear outputs. These processes primarily serve to adjust the weights of the neural network. The neural network weights and biases are assigned initial values during the network initialization phase. During training, the weights of each neuron are then adjusted based on predicted differences, which allows for iterative refinement of the network's performance.

$$\bar{w} = w - \alpha \frac{\partial loss}{\partial w} \tag{10}$$

where $\bar{w}$ represents the updated weight, α denotes the learning rate, and $loss$ represents the loss function. During training, the neural network carries out a backward computation for each forward computation. Finally, the weights are adjusted to attain optimal values that minimize the differences between the predicted and actual values. FCNNs offer several advantages compared to other deep learning architectures like CNNs, RNNs, ResNets, and Transformers. FCNNs are simpler to understand and implement, making them easier to debug and interpret. They can be more parameter-efficient in scenarios where every input feature is relevant to every output class. FCNNs are highly flexible, capable of handling various data types, and unconstrained by input shapes. They generally require fewer computational resources, particularly when the architecture is not very deep. Additionally, FCNNs consider the global context of the data, as each node in a layer is connected to every node in the subsequent layer, unlike the local context considered in convolutional layers.

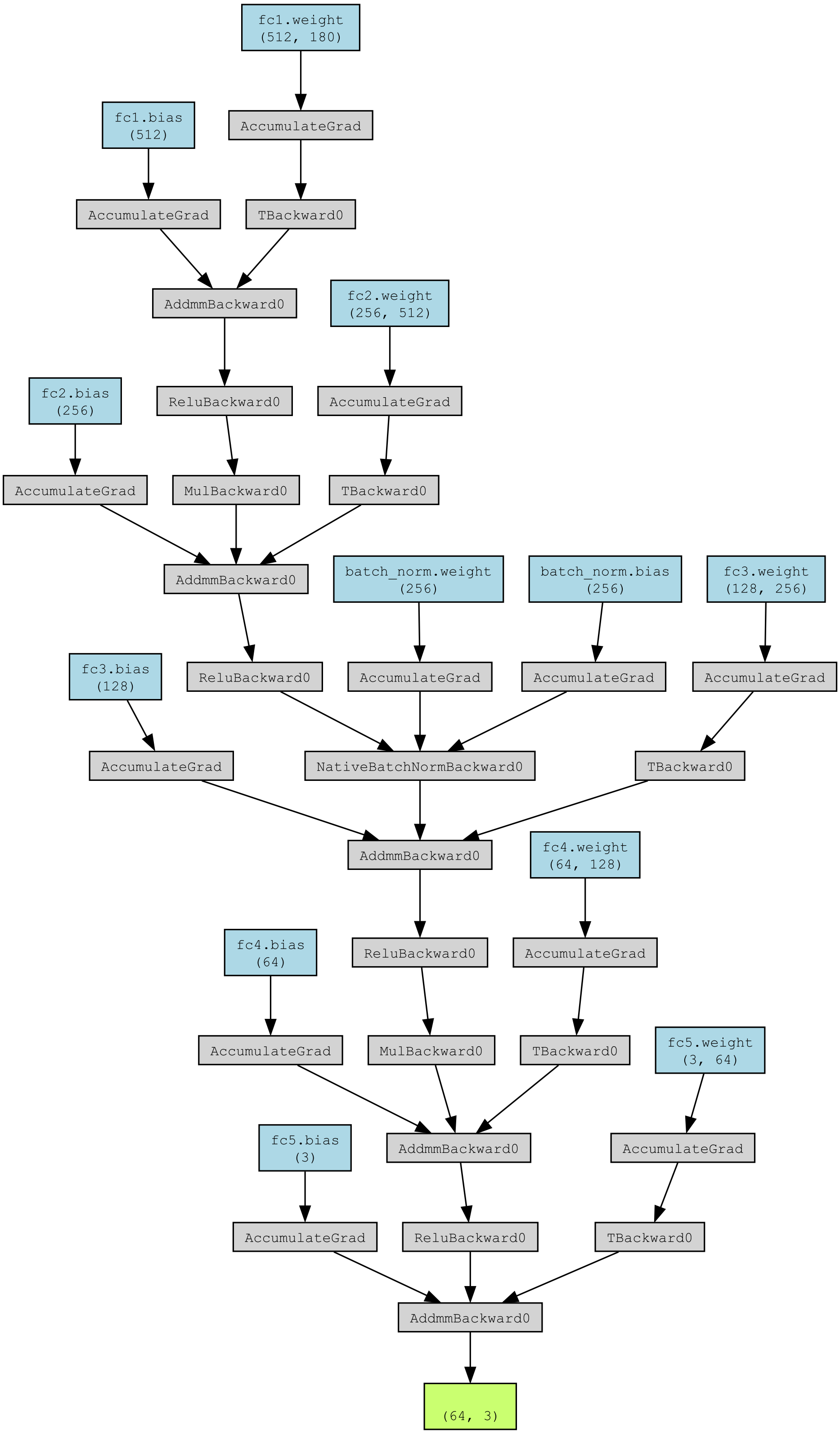

FIGURE 4. The architecture of the proposed FCNN model is visualized using Torchviz. The model takes as input a tensor with a batch size of 64 and an input dimension of 180. It consists of one input layer, four hidden layers, and one output layer, each with varying numbers of neurons. In this diagram, the term 'fcx' denotes a fully connected layer, where 'x' identifies the layer number. The numbers in brackets indicate the dimensions of the output and input, respectively.

D. REGULARIZATION TECHNIQUES

The model incorporates vital regularization techniques to enhance its learning capacity and mitigate overfitting. Employing dropout after the first and third hidden layers, with a dropout rate of 0.5, strategically introduces randomness by temporarily deactivating neurons during training, thereby promoting robust feature learning and preventing excessive reliance on any specific neurons. Furthermore, batch normalization is seamlessly integrated after the second hidden layer, ensuring stabilized activations and facilitating smoother convergence during training. These regularization strategies collectively reinforce the model's ability to generalize well to unseen data while maintaining an optimal trade-off between complexity and generalization performance.

E. LOSS FUNCTION FOR MULTICLASSIFICATION

1) FOCAL LOSS (L_F)

The L_F is a specialized loss function designed to address class imbalance in binary and multi-class classification tasks. It was introduced in the paper [27]. The L_F aims to give more emphasis to hard-to-classify patterns during training, thereby mitigating the impact of the dominant class in imbalanced datasets. It does so by downweighting the losses assigned to well-classified examples. The L_F is defined as follows [27]:

$$L_F(\hat{p}_y) = -\omega_t(1-\hat{p}_y)^{\gamma}\, log(\hat{p}_y) \tag{11}$$

where $y \in \{0, \ldots, C-1\}$ denotes an integer class label (C is the number of the classes), p_y is the estimated probability over the C, and ω_t is a weighting factor for each class (typically set to 1 for binary classification but can be a vector for multi-class classification). γ is a focusing parameter that determines the extent to which higher-confidence correct predictions influence the overall loss. As γ increases, the weight given to easily-classifiable examples diminishes.

2) MULTI-CLASS MARGIN LOSS (L_{MCM})

The L_{MCM} is a loss function designed to increase the margin between predicted scores for the true class and other classes. It uses hinge loss with a power exponent to emphasize loss for larger differences and allows for class-specific weighting. The loss is computed element-wise for each sample and then summed and normalized to obtain a scalar loss value. The loss encourages the model to have higher confidence in the correct class prediction while considering weighted and margin-adjusted score differences. It is defined as follows [28]:

$$L_{MCM}(x, y) = \frac{\sum_i max(0, w[y] \times (m - x[y] + x[i]))^k}{N} \tag{12}$$

where x and y are the sample and the true label, respectively. $w[y]$ is a weight associated with the true class label, and m is a margin value. $x[y]$ is the predicted score for the true class y. N is the number of samples in the batch.

3) CATEGORICAL CROSS ENTROPY LOSS (L_{CCE})

The L_{CCE} is an essential loss function commonly used in multiclass classification. It is critical for training algorithms to produce accurate class probability estimates. Calculated by comparing predicted probabilities with actual class labels, each value in the resulting loss vector corresponds to the cross-entropy loss for a single sample. The function effectively balances prediction accuracy across classes while penalizing errors. Its primary strength is its ability to handle multiple categories, directing the model to allocate higher probabilities to the correct classes. Through the use of the softmax function, L_{CCE} converts raw output scores into valid class probabilities, aiding in stable optimization and convergence during training. The loss function is instrumental in elevating the performance of machine learning classifiers in multiclass settings. The L_{CCE} is defined as follows [29]:

$$L_{CCE}(x, y) = \{l_1, \ldots, l_N\}^T,\ l_n = -\sum_{c=1}^{C} y_{n,c} \times log(p_{n,c}) \tag{13}$$

$$p_{n,c} = \frac{e^{x_{n,c}}}{\sum_{i=1}^{C} e^{x_{n,i}}} \tag{14}$$

where N is the number of patterns, l_n refers to the cross-entropy loss for the nth pattern, C is the number of classes. $y_{n,c}$ is the true label (ground truth) of the nth pattern for class c. $p_{n,c}$ is the predicted probability of the nth pattern belonging to class c. $x_{n,c}$ is the raw score (logit) for the nth sample in class c. e is the base of the natural logarithm. $\sum_{i=1}^{C} e^{x_{n,i}}$ is the sum of exponential raw scores for the nth sample over all classes.

4) HUBER LOSS (L_H)

It's generally used for regression but can be adapted for multiclass classification [30]. It's less sensitive to outliers because it's quadratic for small values and linear for large values. In multiclass classification, the L_H can be used as a custom loss function. The L_H is defined as follows:

$$L_H(y_p, y_t) = \begin{cases} 0.5 \times (y_t - y_p)^2 & |y_t - y_p| \le \delta \\ \delta \times |y_t - y_p| - 0.5 \times \delta^2 & |\mathrm{y}_t - \mathrm{y}_p| > \delta \end{cases} \tag{15}$$

where y_t and y_p are the true and predicted values, respectively. In regression tasks, the goal is to predict a continuous target variable, such as a person's age, the price of a house, or a temperature value. The L_H in this context aims to combine the properties of both the Mean Squared Error (MSE) loss and the Mean Absolute Error (MAE) loss. It behaves quadratically (like MSE) for small errors and linearly (like MAE) for larger errors.

In multiclass classification tasks, the goal is to classify an input into one of several possible classes. For example, the task may involve categorizing an animal based on its attributes. While the L_H can be adapted to handle multiclass classification, the method of implementation varies due to the distinct nature of multiclass problems. In this context, it becomes essential to factor in the existence of multiple classes, necessitating adjustments to the predicted probabilities.

A standard approach for employing L_H in multiclass scenarios involves converting the raw prediction scores into probabilities via a softmax function. Subsequently, the loss is computed by contrasting these predicted probabilities with the actual class labels. This calculation takes into account the characteristics of categorical cross-entropy loss. The softmax function first converts raw scores into positive values through exponentiation and then normalizes these by dividing them by the sum of all the exponentiated values. This step ensures that the final probabilities sum to 1, making them apt for representing class probabilities. The softmax transformation is commonly used in multiclass classification tasks to convert model outputs into a probability distribution over the classes, making it easier to interpret and use for making predictions.

$$L_H(y_p, y_t) = \begin{cases} max(0, 1 - y_p y_t)^2 & y_p y_t > -1 \\ -4 y_p y_t & \text{Otherwise} \end{cases} \tag{16}$$

5) JOINT LOSS (CATEGORICAL CROSS ENTROPY LOSS + MULTI-CLASS MARGIN LOSS) ($L_{CCE+MCM}$)

Experimentally, in the data of this study, Categorical Cross-Entropy and Multi-Class Margin Losses yielded the best performance results. Consequently, we combined these losses $L_{CCE+MCM}$ by linearly scaling them with their respective weight coefficients. The weights α and δ determine the balance between the two losses. We can adjust these weights based on the desired importance of each loss component. However, choosing appropriate weight values is critical to ensuring that both losses contribute effectively to the learning process. If one loss is significantly larger than the other, it may dominate the training, potentially affecting the convergence and the learning behavior of the model.

$$L_{CCE+MCM} = \alpha \times L_{CCE} + \delta \times L_{MCM} \tag{17}$$

A detailed study of the chosen values of α and δ in our experiments is provided in subsection IV-E.

IV. EXPERIMENTS

To comprehensively assess the competency of the proposed method, specifically its ability to classify a particular scene as structured, semi-structured, or unstructured, a methodical evaluation process was conducted. This involved using the collective motion database (CMD), a publicly accessible database, to train the classifier using the proposed model. Details on the CMD dataset are in subsection IV-A. The essence of this evaluative procedure lies not merely in evaluating the effectiveness of our model but also in comparing it with state-of-the-art approaches in the domain of crowd scene classification. These cutting-edge methodologies, notable for their reliance on the quantification of collectiveness within the scene, provided a benchmark against which our model could be critically examined.

A. DATASET

The CMD, put forth by [16], serves as the benchmark to assess the efficacy of our proposed method. Comprising 413 distinct crowd video clips, this dataset offers one hundred frames per clip. Additionally, it includes ground truth labels for every clip, falling under the classification scheme of $\{0, 1, 2\}$. Examples of the CMD dataset are shown in Fig. 5.

- **Unstructured scenes (Label 0)**: These comprise the majority of the scenes, with a total of 216 instances. This category may represent scenes where there's no clear pattern or organization, such as a busy market street where people and vehicles move freely without specific paths.
- **Semi-structured scenes (Label 1)**: There are 107 instances of semi-structured scenes, falling in between the other two categories. These scenes might contain some rules or patterns governing movement, but these may not be consistently followed.
- **Structured scenes (Label 2)**: The structured category represents the most orderly scenes, with 90 instances. These scenes could include well-organized traffic systems or controlled public spaces where movement follows specific paths or rules.

As shown in Fig. 6, the data indicate that unstructured scenes are the most common, making up more than half of the total scenes. This suggests a need for more sophisticated analysis techniques capable of handling the complexity of unstructured environments. There is a noticeable imbalance between the categories, with structured scenes being the least represented. This could have implications for modeling and classification tasks, as the imbalance might lead to biased predictions towards the majority class (unstructured).

B. FINDINGS AND ANALYSIS

A comparative analysis of the F1-score across different machine learning models and three types of data structure combinations is presented in Fig. 7. While ensemble methods such as adaptive boosting (AdaBoost), random forest (RF), and eXtreme gradient boosting (XGBoost) [7] generally outperform earlier models by Zhou et al. [12], Shao et al. [13], and Li et al. [15], they excel primarily in specific data structure combinations. Additionally, the ν-support vector machine (ν-SVM) [7] and multi-layer perceptron (MLP) exhibited performance almost comparable to XGBoost. In contrast, FCNN, SMOTE-FCNN, and ADASYN-FCNN consistently achieve high F1-scores across all types of data structures, establishing them as more versatile solutions.

Moreover, SMOTE-FCNN and ADASYN-FCNN are uniquely designed to handle imbalanced datasets, offering robust performance across a wide range of real-world scenarios. Due to their consistently high F1-score across various data structures, these models, along with FCNN, also exhibit superior generalization capabilities. Additionally, these models offer greater adaptability to different data types and structures. Unlike ensemble methods, which rely on

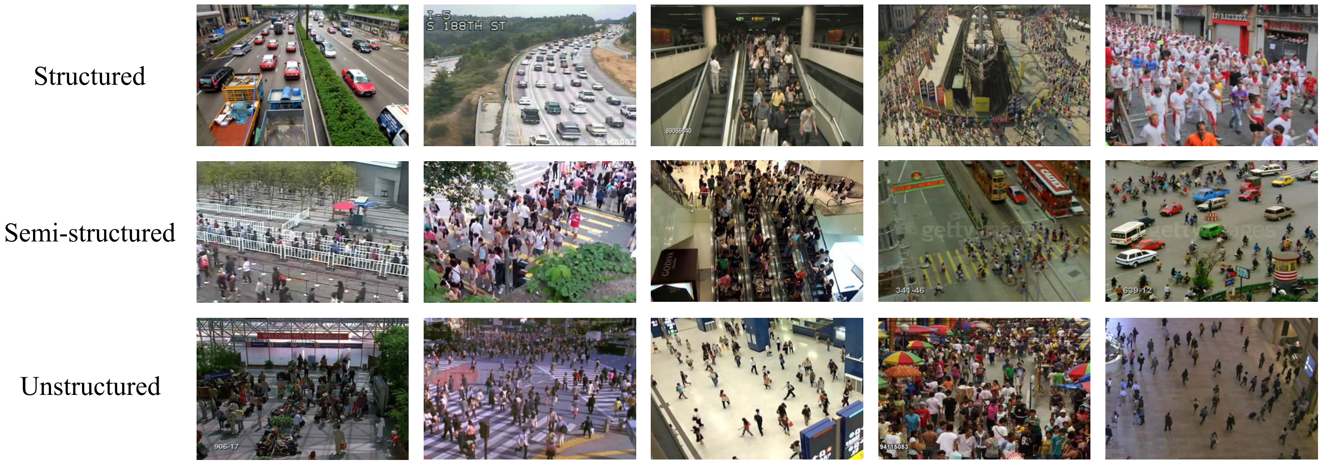

FIGURE 5. The CMD is a challenging dataset because it consists of 413 video clips with a variety of motions, varied perspective views, occlusions, and tracking noise. The CMD video clips are categorized into structured, semi-structured, and unstructured crowd scenes.

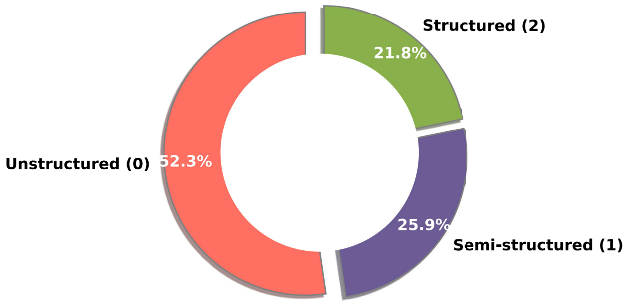

FIGURE 6. Distribution of different types of crowd scenes in CMD dataset.

pre-defined strategies, FCNN and its variants can be easily fine-tuned for specific tasks.

Fig. 8 presents the confusion matrix for the multiclassification of motion patterns into structured, semi-structured, and unstructured categories. The matrix serves as an evaluative tool for the performance of our ADASYN-FCNN classification model in discriminating among these distinct data types. In the matrix, rows correspond to actual labels, while columns represent predicted labels. A preponderance of values along the diagonal line indicates high accuracy in the model's predictions. The elevated values along the diagonal of the matrix corroborate the model's exceptional accuracy, affirming its robustness and reliability across diverse data categories. Notably, the sparse off-diagonal elements signify a reduced incidence of false positives and negatives, a crucial benefit in contexts where the ramifications of misclassification are considerable. Moreover, the balanced allocation of diagonal elements substantiates the model's ability for impartial class discrimination, effectively differentiating between unstructured, semi-structured, and structured data categories. This equilibrated performance underscores the model's robustness and indicates its aptitude for strong generalization, rendering the ADASYN-FCNN model exceptionally suitable for managing diverse and imbalanced datasets in real-world applications.

TABLE 1. Summary of multiclassification performance for FCNN and augmented patterns models.

Method	Accuracy	Precision	Recall	F1-score
FCNN	0.8867	0.8953	0.8867	0.8863
SMOTE-FCNN	0.9204	0.9297	0.9204	0.9210
ADASYN-FCNN	**0.9397**	**0.9481**	**0.9397**	**0.9416**

C. COMPARATIVE ANALYSIS OF ADASYN-FCNN AND SMOTE-FCNN METHODS

In this section, two of the most popular data balancing techniques, SMOTE and ADASYN, are compared for the purpose of motion pattern-based crowd scene classification. Table 1 provides a detailed analysis of the performance metrics of FCNN and its augmented pattern models, SMOTE-FCNN and ADASYN-FCNN, in a multiclassification task. ADASYN-FCNN outperforms both FCNN and SMOTE-FCNN, demonstrating its superior ability to correctly classify instances across multiple classes. In general, the FCNN architecture can model complex relationships in the data. The synergistic combination of FCNN with SMOTE or ADASYN provides a more fine-grained understanding of feature interactions, leading to higher performance. Moreover, FCNNs are known for their ability to learn useful feature representations. When combined with SMOTE or ADASYN, which adds variability to the data, the model might capture a richer set of features, leading to higher accuracy, precision, and recall.

In addition, our models were evaluated using robust techniques like cross-validation, which strengthens the case that ADASYN-FCNN's high scores are indicative of the model's generalizability. The superior performance of ADASYN-FCNN compared to SMOTE-FCNN can be attributed to several key methodological differences between the two

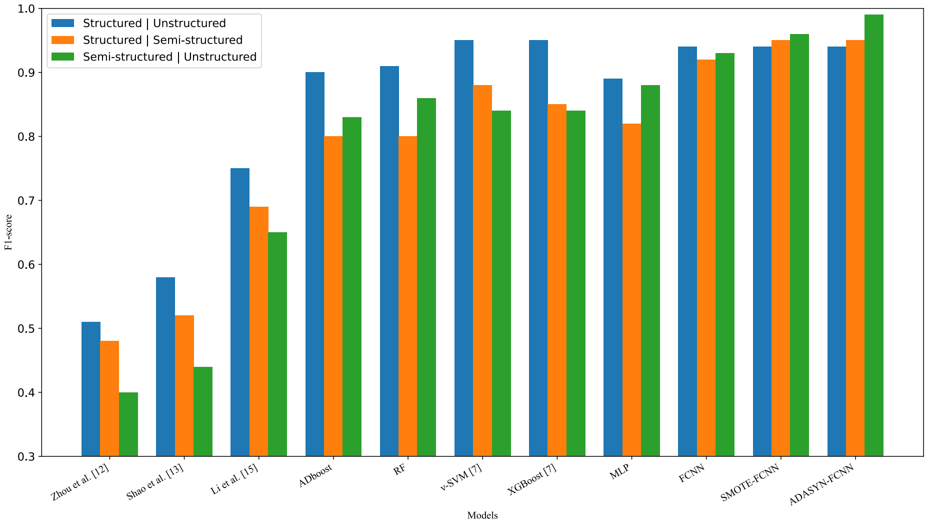

FIGURE 7. Evaluating F1-score performance: a comparative analysis of machine learning algorithms across diverse data structures for binary classification. The reported outcomes of methods by Zhou et al. [12], Shao et al. [13], and Li et al. [15] are sourced from [7].

oversampling techniques. Unlike SMOTE, which uniformly generates synthetic samples for each minority class instance, ADASYN dynamically adjusts the number of synthetic samples based on the local difficulty of classification for each minority instance. This targeted approach enables ADASYN to concentrate on instances that are more challenging to classify, thereby enhancing the model's overall effectiveness. Additionally, as described in subsection III-B, ADASYN employs a k-Nearest Neighbors algorithm to generate synthetic samples in proximity to minority instances that are commonly misclassified. This localized strategy results in a model that is both precise and context-sensitive, which likely accounts for its elevated performance metrics. Furthermore, the adaptive and localized nature of ADASYN's synthetic sample generation leads to a more balanced representation of minority classes in the feature space, mitigating the risk of model overgeneralization that is often associated with SMOTE. Lastly, ADASYN's adaptive capabilities make it particularly adept at handling noisy data, as its flexible sample generation is sensitive to the complexities inherent in the data distribution.

Table 2 showcases the performance metrics of three distinct models: FCNN, SMOTE-FCNN, and ADASYN-FCNN, evaluated across three different data structure combinations. Each of the models demonstrates commendable performance, with accuracy and F1-score predominantly surpassing the 0.90 mark. Among them, ADASYN-FCNN emerges as the most exemplary, excelling across all data structure combinations and performance metrics. It notably attains an F1-score of 0.99 in the semi-structured | unstructured categories. While all models exhibit strong performance, the ADASYN-FCNN distinguishes itself as the most robust and effective choice among them.

D. COMPARATIVE ANALYSIS OF DIFFERENT LOSS FUNCTIONS

The choice of a loss function plays a crucial role in training a machine learning model, particularly for multiclassification tasks. It serves as the objective function that the optimization algorithm seeks to minimize, thereby directly influencing the model's ability to generalize well to unseen data. Different loss functions capture different types of errors and imbalances in the data, making the choice of an appropriate loss function crucial for achieving high performance in a given application.

Based on Table 3, MultiMargin Loss outperforms other loss functions like Focal Loss, Huber Loss, and CrossEntropy Loss across all the key performance metrics. The superior performance of MultiMargin Loss in multiclassification tasks can be attributed to several key features. Firstly, it is designed to effectively handle class imbalances, a common challenge in such tasks, as evidenced by its outstanding performance across all key metrics. Secondly, its focus on maximizing the margin between classes leads to better-defined decision boundaries, contributing to the model's high precision and recall rates. This margin maximization also enables the model to generalize better to unseen data. Additionally, MultiMargin Loss is less sensitive to outliers, providing a more robust performance in datasets with noisy or extreme values. Finally, it often converges faster than other loss functions, offering both computational efficiency and superior performance.

In our research, we introduce a loss function termed 'Joint Loss,' which combines the strengths of both CrossEntropy Loss and MultiMargin Loss. The empirical results indicate that Joint Loss achieves unparalleled performance, registering an accuracy of 93.97%, a precision of 94.81%, a recall of

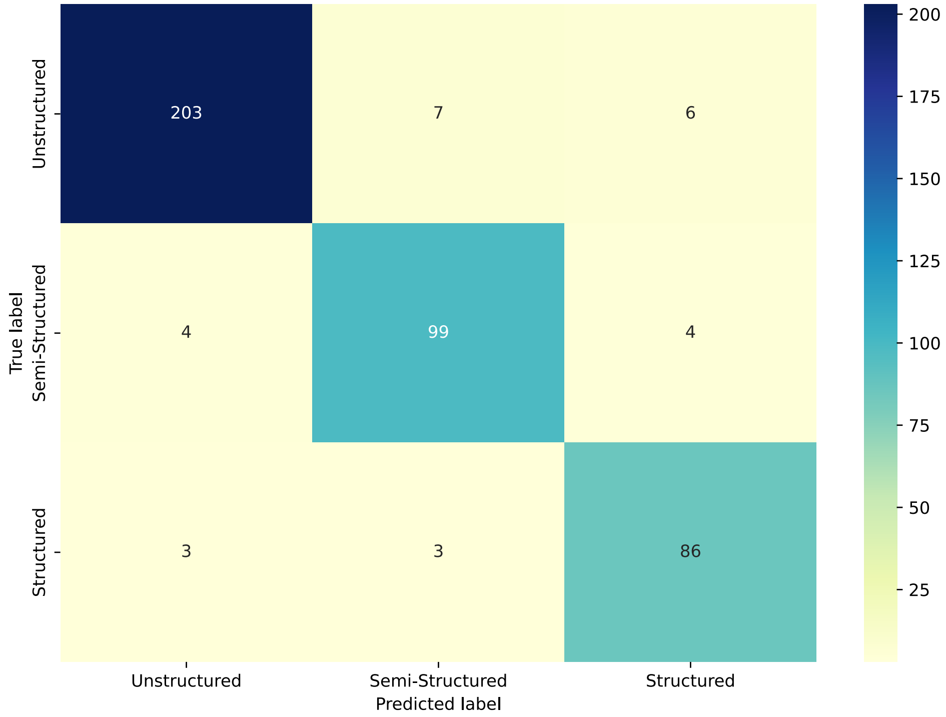

FIGURE 8. Confusion matrix for the classification of structured, semi-structured, and unstructured motion patterns.

TABLE 2. Evaluation of FCNN, SMOTE-FCNN, and ADASYN-FCNN models in binary classification contexts: A stands for accuracy, P for precision, R for recall, and F1 for F1-score.

Model	Structured | Unstructured				Structured | Semi-structured				Semi-structured | Unstructured			
	A	P	R	F1	A	P	R	F1	A	P	R	F1
FCNN	**0.94**	0.95	**0.94**	**0.94**	0.92	0.93	0.92	0.92	0.94	0.94	0.94	0.93
SMOTE-FCNN	**0.94**	**0.96**	**0.94**	**0.94**	0.94	**0.96**	0.94	**0.95**	0.96	0.96	0.96	0.96
ADASYN-FCNN	**0.94**	**0.96**	**0.94**	**0.94**	**0.95**	**0.96**	**0.95**	**0.95**	**0.98**	**0.99**	**0.98**	**0.99**

TABLE 3. Evaluation of loss functions based on ADASYN-FCNN in terms of accuracy, precision, recall, and F1-score.

Loss Function	Accuracy	Precision	Recall	F1-score
Focal Loss	0.9061	0.9191	0.9061	0.9085
Huber Loss	0.9134	0.9225	0.9134	0.9150
CrossEntropy Loss	0.9278	0.9354	0.9278	0.9288
MultiMargin Loss	0.9374	0.9439	0.9374	0.9380
Joint Loss	**0.9397**	**0.9481**	**0.9397**	**0.9416**

93.97%, and an F1-score of 94.16%. These metrics surpass those achieved by either CrossEntropy Loss or MultiMargin Loss individually, epitomizing the best of both. By amalgamating CrossEntropy Loss, known for its effectiveness in probability estimation, with MultiMargin Loss, renowned for its robustness to class imbalance and margin maximization, Joint Loss capitalizes on the complementary advantages. This hybrid approach results in better-defined decision boundaries and improved generalization to unseen data. Additionally, the lessened sensitivity to outliers and faster convergence rates characteristic of MultiMargin Loss are preserved in this Joint Loss formulation.

E. STUDY α AND δ VALUES FOR THE JOINT LOSS

To evaluate the performance of the model under various configurations of the Joint Loss function $L_{CCE+MCM}$, a heatmap was generated. The heatmap serves as an empirical guide to selecting optimal values of α and δ for balancing the Categorical Cross-Entropy and Multi-Class Margin Losses in the Joint Loss function. By performing that, the model's performance can be fine-tuned, as reflected by the F1-score. As shown in Fig. 9, extensive empirical evaluation indicated that the model achieves optimal performance with $\alpha = 0.10$ and $\delta = 0.50$, yielding the highest F1-score of 0.95. This suggests that the model benefits from a lower emphasis on the Categorical Cross-Entropy Loss and a moderate emphasis on the Multi-Class Margin Loss when combined into the Joint Loss function $L_{CCE+MCM}$. Therefore, in our experiments, we fixed the values of α and δ at these levels.

F. COMPARATIVE ANALYSIS USING 10-FOLD CROSS-VALIDATION

Utilizing cross-validation in machine learning algorithms is essential for achieving dependable, trustworthy, and

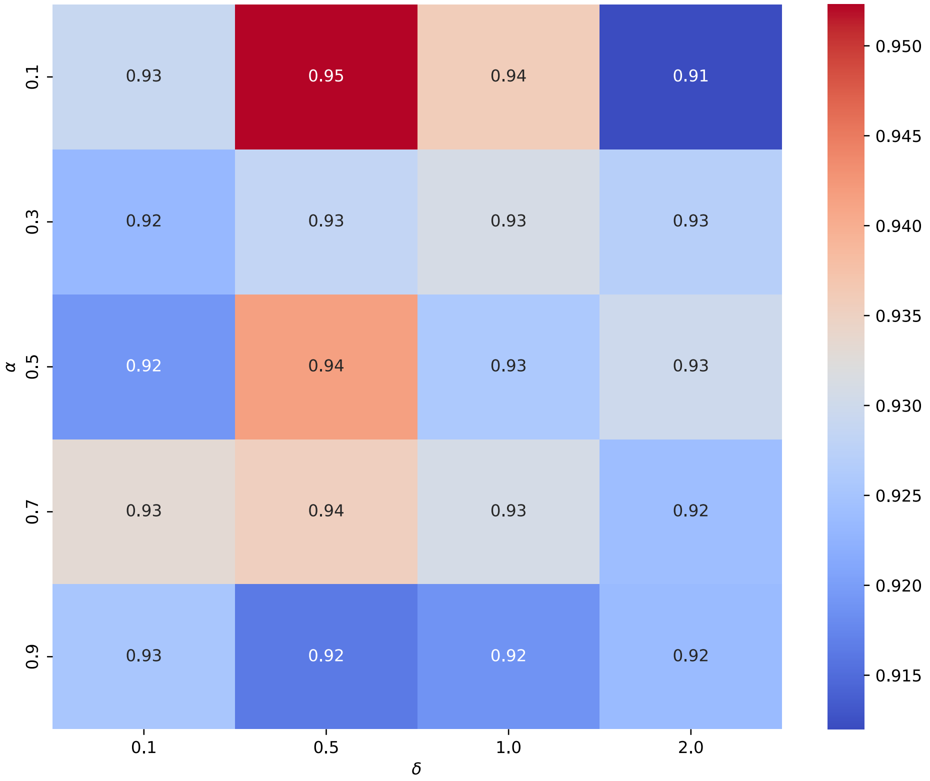

FIGURE 9. F1-score heatmap across various α and δ values for the Joint Loss.

TABLE 4. Evaluation of multiclassification models with and without 10-fold cross-validation.

	Without Cross-validation				With Cross-validation			
Model	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
ADboost	0.65	0.67	0.65	0.64	0.74	0.76	0.74	0.74
RF	0.73	0.73	0.73	0.73	0.79	0.80	0.79	0.78
ν-SVM [7]	0.76	0.76	0.76	0.75	0.80	0.83	0.80	0.81
XGBoost [7]	0.77	0.77	0.77	0.77	0.81	0.82	0.80	0.80
MLP	0.77	**0.83**	0.77	**0.80**	0.79	0.83	0.80	0.80
ADASYN-FCNN	**0.80**	0.81	**0.80**	**0.80**	**0.94**	**0.95**	**0.94**	**0.94**

broadly applicable outcomes, especially when dealing with an uneven class distribution. Segmenting the initial dataset into various subgroups for both training and validation, cross-validation minimizes the likelihood of model overfitting and allows for an exhaustive assessment of the model's efficacy across diverse data points. This technique is crucial when confronting class imbalances, a frequent challenge that can distort the model's predictive precision. When combined with stratified sampling, cross-validation guarantees that each subset contains a balanced representation of all classes, thereby enabling a well-rounded and sturdy evaluation. It also facilitates the use of various resampling techniques across different folds, enhancing the model's generalization capabilities on imbalanced datasets. As a result, cross-validation acts as both a robust evaluation technique and an effective tool for mitigating challenges posed by class imbalance, thereby validating the model's true predictive power.

Table 4 highlights the substantial influence of 10-fold cross-validation on model performance. Notably, this impact extends to diverse models such as Adaptive Boosting (ADboost), Random Forest (RF), ν-Support Vector Machines (ν-SVM), Extreme Gradient Boosting (XGBoost), Multilayer Perceptron (MLP), and ADASYN-FCNN, as evidenced by their consistent performance improvement across all key metrics. Each model exhibits improved performance metrics when cross-validation is applied, underscoring the added robustness provided by this evaluation technique. ADASYN-FCNN stands out for its consistent and reliable performance, excelling with and without the use of cross-validation.

V. CONCLUSION

In this study, we introduce a streamlined fully connected neural network designed to classify crowds into three distinct categories: structured, semi-structured, and unstructured. This classification is predicated on the uniformity or variability of the movement patterns exhibited by the assembled objects. For crowds deemed to be structured, the observed motion is both coherent and uniform, signifying that either the entire crowd or specific segments within it display consistent directional and velocity characteristics. On the contrary, unstructured crowds demonstrate inconsistent and unstable motion, leading to a diverse array of velocities and trajectories. Semi-structured crowds introduce a heightened level of complexity, occupying an intermediate position between structured and unstructured categories, thus introducing challenges when attempting to classify them into distinct categories.

Through trajectory-based scene classification, it offers valuable insights into the crowd's inherent dynamics, offering a foundation for multiple applications, including crowd control, urban planning, public safety, and the formulation of tailored strategies for surveillance and interference. In essence, the approach offers a sophisticated tool for understanding and responding to the complexities of crowded environments, contributing to both theoretical knowledge and practical applications in the field of crowd analysis.

This study lays the groundwork for several promising avenues for future research and practical applications in crowd analysis. Future work could delve into more granular sub-categorizations of crowd behaviors to better capture the wide range of dynamics observed. Additionally, the model could be tailored for specific scenarios such as sporting events, protests, or concerts, where crowd characteristics may differ.

The integration of human expertise through a human-in-the-loop system could further refine the model's classifications, particularly in ambiguous or high-stakes situations. Collaborative efforts with professionals in urban planning, psychology, and law enforcement could translate these theoretical models into actionable strategies for effective crowd management and public safety. Lastly, the potential for integrating the FCNN model into an Internet of Things (IoT) framework could offer real-time, sensor-based crowd management solutions. These future directions strive to both deepen theoretical understanding and create practical solutions that are ready for implementation in the field of crowd analysis.

ACKNOWLEDGMENT

The authors extend their appreciation to the Research Management Centre (RMC) at Universiti Teknologi Malaysia (UTM) for funding this research work under the Professional Development Research University grant (R.J130000.7113.06E45).

REFERENCES

[1] V. D. Huszár, V. K. Adhikarla, I. Négyesi, and C. Krasznay, "Toward fast and accurate violence detection for automated video surveillance applications," *IEEE Access*, vol. 11, pp. 18772–18793, 2023.

[2] Y. Himeur, S. Al-Maadeed, H. Kheddar, N. Al-Maadeed, K. Abualsaud, A. Mohamed, and T. Khattab, "Video surveillance using deep transfer learning and deep domain adaptation: Towards better generalization," *Eng. Appl. Artif. Intell.*, vol. 119, Mar. 2023, Art. no. 105698.

[3] M. U. Farooq, M. N. M. Saad, and S. D. Khan, "Motion-shape-based deep learning approach for divergence behavior detection in high-density crowd," *Vis. Comput.*, vol. 38, pp. 1553–1577, Feb. 2021.

[4] A. Hafeezallah, A. Al-Dhamari, and S. Abd Rahman Abu-Bakar, "Multi-scale network with integrated attention unit for crowd counting," *Comput., Mater. Continua*, vol. 73, no. 2, pp. 3879–3903, 2022.

[5] A. Hafeezallah, A. Al-Dhamari, and S. A. R. Abu-Bakar, "U-ASD Net: Supervised crowd counting based on semantic segmentation and adaptive scenario discovery," *IEEE Access*, vol. 9, pp. 127444–127459, 2021.

[6] A. Al-Dhamari, R. Sudirman, and N. H. Mahmood, "Transfer deep learning along with binary support vector machine for abnormal behavior detection," *IEEE Access*, vol. 8, pp. 61085–61095, 2020.

[7] A. K. Pai, P. Chandrahasan, U. Raghavendra, and A. K. Karunakar, "Motion pattern-based crowd scene classification using histogram of angular deviations of trajectories," *Vis. Comput.*, vol. 39, no. 2, pp. 557–567, Feb. 2023.

[8] A. S. Patel, R. Vyas, O. P. Vyas, M. Ojha, and V. Tiwari, "Motion-compensated online object tracking for activity detection and crowd behavior analysis," *Vis. Comput.*, vol. 39, no. 5, pp. 2127–2147, May 2023.

[9] W. Zhao, Z. Zhang, and K. Huang, "Gestalt laws based tracklets analysis for human crowd understanding," *Pattern Recognit.*, vol. 75, pp. 112–127, Mar. 2018.

[10] S. A. M. Saleh, A. H. Kadarman, S. A. Suandi, S. A. A. Ghaleb, W. A. H. Ghanem, S. Shuib, and Q. S. Hamad, "A tracklet-before-clustering initialization strategy based on hierarchical KLT tracklet association for coherent motion filtering enhancement," *Mathematics*, vol. 11, no. 5, p. 1075, Feb. 2023.

[11] A. Roy, N. Biswas, S. K. Saha, and B. Chanda, "Classification of moving crowd based on motion pattern," in *Proc. IEEE Region Symp. (TENSYMP)*, Jun. 2019, pp. 102–107.

[12] B. Zhou, X. Tang, H. Zhang, and X. Wang, "Measuring crowd collectiveness," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1586–1599, Aug. 2014.

[13] J. Shao, C. C. Loy, and X. Wang, "Learning scene-independent group descriptors for crowd understanding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 6, pp. 1290–1303, Jun. 2017.

[14] X. Li, M. Chen, and Q. Wang, "Measuring collectiveness via refined topological similarity," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 12, no. 2, pp. 1–22, Mar. 2016.

[15] X. Li, M. Chen, and Q. Wang, "Quantifying and detecting collective motion in crowd scenes," *IEEE Trans. Image Process.*, vol. 29, pp. 5571–5583, 2020.

[16] A. Al-Dhamari, R. Sudirman, and N. H. Mahmood, "Abnormal behavior detection using sparse representations through sequential generalization of k-means," *TURKISH J. Electr. Eng. Comput. Sci.*, vol. 29, no. 1, pp. 152–168, Jan. 2021.

[17] A. Al-Dhamari, R. Sudirman, and N. H. Mahmood, "Abnormal behavior detection in automated surveillance videos: A review," *J. Theor. Appl. Inf. Technol.*, vol. 95, no. 19, pp. 5245–5263, 2017.

[18] K. Oksuz, B. C. Cam, S. Kalkan, and E. Akbas, "Imbalance problems in object detection: A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3388–3415, Oct. 2021.

[19] M. Ghorbani, A. Kazi, M. S. Baghshah, H. R. Rabiee, and N. Navab, "RA-GCN: Graph convolutional network for disease prediction problems with imbalanced data," *Med. Image Anal.*, vol. 75, Jan. 2022, Art. no. 102272.

[20] G. Wang, J. Wang, and K. He, "Majority-to-minority resampling for boosting-based classification under imbalanced data," *Int. J. Speech Technol.*, vol. 53, no. 4, pp. 4541–4562, Feb. 2023.

[21] Y. Liu, Y. Liu, B. X. B. Yu, S. Zhong, and Z. Hu, "Noise-robust oversampling for imbalanced data classification," *Pattern Recognit.*, vol. 133, Jan. 2023, Art. no. 109008.

[22] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural Netw.*, vol. 106, pp. 249–259, Oct. 2017.

[23] D. Roy, R. Pramanik, and R. Sarkar, "Margin-aware adaptive-weighted-loss for deep learning based imbalanced data classification," *IEEE Trans. Artif. Intell.*, early access, May 11, 2023, doi: 10.1109/TAI.2023.3275133.

[24] W. H. Warren, J. B. Falandays, K. Yoshida, T. D. Wirth, and B. A. Free, "Human crowds as social networks: Collective dynamics of consensus and polarization," *Perspect. Psychol. Sci.*, Aug. 2023, Art. no. 17456916231186406.

[25] A. K. Pai, A. K. Karunakar, and U. Raghavendra, "Scene-independent motion pattern segmentation in crowded video scenes using spatio-angular density-based clustering," *IEEE Access*, vol. 8, pp. 145984–145994, 2020.

[26] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, Jun. 2008, pp. 1322–1328.

[27] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020.

[28] PyTorch Contributors. (2023). *Multi Margin Loss*. Accessed: Sep. 5, 2023. [Online]. Available: https://pytorch.org/docs/stable/generated/torch.nn.MultiMarginLoss.html

[29] PyTorch Contributors. (2023). *Cross Entropy Loss*. Accessed: Sep. 5, 2023. [Online]. Available: https://pytorch.org/docs/stable/generated/torch.nn.CrossEntropyLoss.html

[30] H. Wang and Y. Shao, "Fast truncated Huber loss SVM for large scale classification," *Knowl.-Based Syst.*, vol. 260, Jan. 2023, Art. no. 110074.

MOHAMMED SULTAN MOHAMMED (Member, IEEE) received the B.Sc. degree in computer engineering from Hodeidah University, Yemen, in 2005, the M.Sc. degree in computer engineering and networks from The University of Jordan, Jordan, in 2015, and the Ph.D. degree in electrical engineering (computer engineering) from Universiti Teknologi Malaysia (UTM), Malaysia, in 2022. He is currently a Researcher with UTM, under the Postdoctoral Fellowship Scheme (R.J130000.7113.06E45) for the Project "Thermal-Aware Performance Optimization of 3D Dark Silicon Many-Core Systems." His research interests include computer architectures, many-core system-on-chip (MCSoC), network-on-chip (NoC), power and thermal management, machine learning, and computer vision.

AHLAM AL-DHAMARI received the B.Sc. degree in computer engineering from Hodeidah University, Yemen, the M.Sc. degree in computer engineering and networks from The University of Jordan, Jordan, and the Ph.D. degree in electrical engineering from Universiti Teknologi Malaysia (UTM), Malaysia. She was a Postdoctoral Researcher under an international fellowship with UTM. She is currently a Faculty Member with the Computer Engineering Department, Faculty of Computer Science and Engineering, Hodeidah University. Her research interests include computer vision, machine learning, deep learning, image and video processing, computer architectures, big data analysis, and crowd analysis and management. She is a highly active reviewer for well-known international journals and conferences.

WADDAH SAEED received the Ph.D. degree in information technology from Universiti Tun Hussein Onn Malaysia, in 2019. He is currently a Senior Lecturer of data analytics and holds the position of a program lead for M.Sc. data analytics and M.Sc. business intelligence and data mining with the School of Computer Science and Informatics, De Montfort University (DMU), U.K. Before joining DMU, he was a Postdoctoral Research Fellow with the University of Agder, Norway. Prior to that, he held a lecturing position with the Asia Pacific University of Technology and Innovation (APU), Malaysia. His research interests include time series analysis and forecasting, machine learning, and explainable AI.

FATIMA N. AL-ASWADI received the B.Sc. degree in computer science from Hodeidah University, Yemen, in 2005, the M.Sc. degree in computer sciences from King AbdulAziz University, Jeddah, Saudi Arabia, in 2014, and the Ph.D. degree from the School of Computer Sciences, Universiti Sains Malaysia (USM), Malaysia, in 2023. She is currently a Faculty Member with the Computer Science Department, Faculty of Computer Science and Engineering, Hodeidah University. Her areas of specialization encompass ontology learning, deep learning, machine learning, NLP, knowledge mining, and knowledge engineering. Her research interests include both foundational and applied aspects, including text representation, concept extraction, relation discovery, information retrieval, sentiment analysis, data mining, ontology, knowledge graphs, artificial neural networks, classification, and intelligent systems.

SAMI ABDULLA MOHSEN SALEH received the B.Eng. degree in computer engineering from Hodeidah University, Yemen, and the M.Sc. degree in electronic systems design engineering and the Ph.D. degree in computer vision and machine learning from Universiti Sains Malaysia. He is currently a Senior Lecturer with the School of Electrical and Electronic Engineering, Universiti Sains Malaysia. His research interests include broad spectrum of innovative fields, including computer vision, deep learning, swarm intelligence, soft biometrics, and the exciting world of the Internet of Things (IoT).

M. N. MARSONO received the B.Eng. degree in computer engineering and the M.Eng. degree in electrical engineering from Universiti Teknologi Malaysia, in 1999 and 2001, respectively, and the Ph.D. degree in ECE from the University of Victoria, in 2007. He is currently a Professor of electronic and computer engineering with the Faculty of Electrical Engineering, Universiti Teknologi Malaysia. His research interests include computer architecture, embedded systems, domain-specific reconfigurable computing, network algorithmics, and network processing architectures.