electronics

MDPI

Article

Analyzing Crowd Behavior in Highly Dense Crowd Videos Using 3D ConvNet and Multi-SVM

Mahmoud Elmezain [1,2,*], Ahmed S. Maklad [2,3], Majed Alwateer [2], Mohammed Farsi [2] and Hani M. Ibrahim [2,4]

1 Computer Science Department, Faculty of Science, Tanta University, Tanta 31527, Egypt
2 College of Computer Science and Engineering, Taibah University, Yanbu 966144, Saudi Arabia; amaklad@taibahu.edu.sa (A.S.M.); mwateer@taibahu.edu.sa (M.A.); mafarsi@taibahu.edu.sa (M.F.); habdelrhman@taibahu.edu.sa (H.M.I.)
3 Information Systems Department, Faculty of Computers and Artificial Intelligence, Beni-Suef University, Beni-Suif 62521, Egypt
4 Mathematics & Computer Science Department, Faculty of Science, Menoufiya University, Shibin el Kom 32511, Egypt
* Correspondence: mahmoud_osman_s@yahoo.com

Abstract: Crowd behavior presents significant challenges due to intricate interactions. This research proposes an approach that combines the power of 3D Convolutional Neural Networks (ConvNet) and Multi-Support Vector Machines (Multi-SVM) to study and analyze crowd behavior in highly dense crowd videos. The proposed approach effectively utilizes the temporal information captured by the 3D ConvNet, which accounts for the spatiotemporal characteristics of crowd movement. By incorporating the third dimension as a temporal stack of images forming a clip, the network can learn and comprehend the dynamics and patterns of crowd behavior over time. In addition, the learned features from the 3D ConvNet are classified and interpreted using Multi-SVM, enabling a comprehensive analysis of crowd behavior. This methodology facilitates the identification and categorization of various crowd dynamics, including merging, diverging, and dense flows. To evaluate the effectiveness of the approach, experiments are conducted on the Crowd-11 dataset, which comprises over 6000 video sequences with an average length of 100 frames per sequence. The dataset defines a total of 11 crowd motion patterns. The experimental results demonstrate promising recognition rates and achieve an accuracy of 89.8%. These findings provide valuable insights into the complex dynamics of crowd behavior, with potential applications in crowd management.

Keywords: crowd behavior; dense crowd videos; 3D convolutional neural networks; crowd management

check for updates

Citation: Elmezain, M.; Maklad, A.S.; Alwateer, M.; Farsi, M.; Ibrahim, H.M. Analyzing Crowd Behavior in Highly Dense Crowd Videos Using 3D ConvNet and Multi-SVM. *Electronics* **2024**, *13*, 4925. https://doi.org/10.3390/electronics13244925

Academic Editor: George A. Tsihrintzis

Received: 9 November 2024
Revised: 25 November 2024
Accepted: 6 December 2024
Published: 13 December 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Crowd behavior analysis (CBA) provides insightful information about how people behave in social contexts. These information can be used to improve decision making in real time, strengthen safety protocols, plan cities more efficiently, and manage public areas and events like conferences, exhibitions, train stations, sporting venues, new entertainment events, and smart cities [1–5].

There has been a noticeable rise in both crime and terrorism in recent years. Particularly in busy areas and at events, video surveillance has emerged as an essential tool for deterring crime and violence as well as crowed management [6–8]. The art of crowd activity analysis is still in its infancy. It is widely recognized that mob behavior can predict potential outcomes for an event. If events could be anticipated in advance, crowd control might prove to be highly successful [9]. Studying and comprehending crowd patterns and their collective behavior is crucial given the concerning rate of population growth [10]. The advantages of automatic video surveillance in congested areas cannot be emphasized enough [11–13]. At overcrowded event entrances, it is common to see people pushing each other or using

openings in the crowd for getting in faster, putting pedestrians' comfort and safety at risk. This demonstrates how crucial it is to carry out crowd behavior investigation [14,15].

Video surveillance has become crucial for creating safe areas and improving quality of life in the age of smart cities. To attain the best feasible area coverage, surveillance cameras are placed at predetermined distances [2]. Nevertheless, computer vision for crowd analysis faces difficulties due to the intricate nature of group and individual interactions [1,2,14,16]. The abundance of cameras in cities, transit hubs, airports, sporting arenas, and other public gathering places offers a chance to collect important data. Due to a reduction in interpersonal trust, human behavior analysis is important not just in crowded locations but also in less congested indoor and outdoor spaces. It is crucial to provide an all-encompassing and flexible framework that describes crowd behaviors based on appearance and motion, regardless of camera perspective or context [2].

It is crucial to keep an eye out for unusual actions in regular circumstances in order to improve people's safety and offer prompt support [17]. Numerous crowd tragedies have been shown to have their primary cause in crowd-level aberrant behaviors (CABs). While research on visual anomaly detection for CABs is very sparse, these detection techniques have shown great success in the last ten years in identifying anomalous behaviors at the individual level. When seen regionally, CABs typically do not show a noticeable deviation from normal behavior, unlike individual-level anomalies, which might vary depending on the circumstances [18]. Increasing the identification rate is a challenging challenge because of the diversity of the crowds [19].

Owing to the novelty of the analyzing crowd behavior field, a suitable taxonomy to arrange its various subtasks is still needed [6]. Researchers in computer vision have focused their attention on the difficult task of identifying and localizing abnormal events in crowd situations [13].

Through proactive identification and handling of possible hazards or problematic behavior, we can cultivate a safer atmosphere and provide timely support for those in need. Nevertheless, the intricacy of the enormous volume of data produced presents obstacles in its efficient application for forensic investigations or real-time observation [3]. The challenges of manually watching crowds for crowd control highlights the value of automated analysis [2,6]. At present, human monitoring techniques are ineffective, labor-intensive, tedious, and time-consuming, which increases the risk of missing suspicious activity amid crowds and [12,20,21], and increases the need for automatic analyses.

The application of computer vision techniques is essential in the development of tools that facilitate the automatic interpretation and analysis of crowd behavior [10]. It is still highly challenging to process such data automatically, and thus, more methodological and technological developments are required [1,2,22]. Automatic video surveillance in public, populated areas has been a topic of active research [3]. Autonomous surveillance frameworks are, therefore, essential for the modern world and smart cities to spot and prevent unusual activities [5,13,20].

Scholars from many fields such as artificial intelligence (AI), including machine learning (ML) and deep learning (DL), and social science have expressed an interest in comprehending crowd phenomena [10]. The ML and DL approaches are utilized in video data analysis. These methods address a range of challenges, such as behavior analysis, scene comprehension, scene labeling, human activity recognition, item localization, crowd counting, crowd flow, and event recognition. The increasing accessibility of ML and data processing infrastructures has made crowd analysis a valuable instrument for addressing social, economic, and environmental issues in smart communities [5]. The DL has recently demonstrated encouraging outcomes when compared to conventional techniques by extracting high-level representation and resolving issues through an end-to-end pipeline [3]. DL-based methods have recently shown promise in real-time detection of single-scene video anomalies [12]. The crowd estimation model is constructed using convolutional neural networks with multiple custom layers [23].

Our research makes the following notable contributions:

1. Proposed Approach: The research introduces an approach that combines 3D Convolutional Neural Networks(CNN) (ConvNet) and Multi-Support Vector Machines (Multi-SVM) to study and analyze crowd behavior in highly dense crowd videos. This approach addresses the challenge of understanding complex interactions and dynamics within crowded environments.
2. Varying Temporal Depth Analysis: In our investigation, we explore the effects of varying temporal depth by changing the kernel temporal depth across the layers. In the homogeneous setting, we examine four networks with kernel temporal depths (d) of 1, 3, 5, and 7. This analysis allows us to understand the impact of different temporal depths on the network's ability to capture and interpret crowd behavior patterns
3. Comprehensive Analysis of Crowd Behavior: The combination of 3D ConvNet and Multi-SVM allows for a comprehensive analysis of crowd behavior. The learned features from the 3D ConvNet are classified and interpreted using Multi-SVM, facilitating the identification and categorization of various crowd dynamics, including merging, diverging, and dense flows.
4. Evaluation on Crowd-11 Dataset: To evaluate the effectiveness of our approach, we conduct experiments on the Crowd-11 dataset, which consists of over 6000 video sequences with an average length of 100 frames per sequence. This dataset defines a total of 11 crowd motion patterns, providing a valuable resource for assessing our approach's performance.
5. Promising Results: The experimental results demonstrate promising recognition rates and achieve an accuracy of 89.8% at depth-3. These results highlight the efficacy of the proposed method in capturing and interpreting crowd behavior in highly dense crowd videos.
6. Practical Applications: The findings of this research shed light on the complex dynamics of crowd behavior, providing valuable insights for applications in crowd management, security, and urban planning. The understanding of crowd behavior can contribute to the development of more efficient crowd management strategies and enhance safety and security in crowded environments.

The organization of the research paper is as follows. Section 2 offers a comprehensive examination of the existing research, delving into previous studies and their discoveries. Section 3 introduces the proposed approach, which is divided into two subsections that explain the methodology and implementation in detail. Section 4 concentrates on the experimental results, showcasing and evaluating the data gathered during the study. Lastly, Section 5 concludes the paper by summarizing the main findings and emphasizing their importance within the wider scope of the research subject.

2. Related Work

We outline the latest research in terms of CBA techniques in this section. The intention is to focus on research pertaining to artificial intelligence methods that deal with CBA, such as ML and DL.

2.1. Resent Machine Learning Approaches

In congested environments, Lazaridis et al. [16] devised a method for precisely recognizing and classifying unusual behavior. They proposed using crowd density heat maps and optical flow data to identify panic and fight incidents as anomalous occurrences. Bouhlel et al. [24] proposed a microscopic technique for locating and tracking individuals while calculating their distances from one another. Bhardwaj et al. [9] presented a multicolumn convolutional neural-network-based method for forecasting mob behavior. The collected attributes and their object density maps were linearly mapped using their methodology. Chaudhary et al. [10] have surveyed many crowd analysis models that have been created and used in computer vision. Their objective was to present a thorough synopsis of the literature on many elements of crowd analysis, like human behavior, anomaly

identification, crowd count, the significance of crowd analysis, and the advantages and disadvantages of recent advancements in this field.

2.2. Deep Learning Approaches

Bhushanam et al. [25] provided an extensive analysis of deep learning methods to address the complex temporal structures in unconstrained video analysis that make it difficult to represent features and classify complex motions. Bhuiyan et al. [26] examined the methods and strategies currently in use for crowd analysis from crowd videos, with a focus on methods that applied deep learning to identify aberrant behavior. Their findings motivate embarking on a laborious but thrilling expedition into crowd analysis, categorization, and the identification of any unusual activity.

To address challenges related to crowd monitoring, V. Sharma et al. [7] provided a combined technique to do crowd density estimation and CBA. They created a new CNN-based architecture that takes scale information into account and overcomes scale variation issues in order to predict crowd density.

Employing deep learning methods, Jadhav et al. [21] have created an intelligent system for crowd surveillance and suspicious activity identification. In particular, the integration of Long Short-Term Memory (LSTM) and Fully Convolutional Networks (FCN) was utilized to analyze crowd behavior, demonstrating the potential of deep learning approaches for reshaping the surveillance industry. Abdullah et al. [20] proposed a system for crowd tracking and anomaly detection using deep learning and a social force model. They created super-pixels via an enhanced watershed transform. A conditional random field is applied. Multiperson tracking is carried out after the computed social force has been used. The DL feature pyramid network is used for multiobject categorization. The decision for abnormality detection and the identification of the uncommon objects from the scene was made using Jaccard similarity. For MCC-based CBP, Tripathy et al. [27] presented an end-to-end trainable two-stream multiscale deep architecture. To manage human-scale fluctuation, the first stream extracts multiscale spatial characteristics from the frames using a deep convolution neural network. The second stream uses a multilayer dilated convolution long short-term memory to extract multiscale temporal information from de-background frames. In order to identify seven behavior categories in the PETS2009 event recognition dataset, Rezaei et al. [3] examined a deep architecture for crowd event recognition. Using optical flow images as input, they implemented a combined handcrafted and Conv-LSTM-AE approach to extract a high-level representation of data and perform classification for real-time event recognition.

Alia et al. [14] introduced a method with two main parts. First, they visualized deep optical flow and wheels to construct maps of movement information. Second, they used a classifier based on EfficientNet-B0 in combination with a false reduction technique to detect pushing behavior at the video patch level. With an accuracy of 86%, they also presented a unique strategy based on patches to augment the data in order to address the issue of imbalanced classes in small-scale pushing behavior datasets.

Sharma et al. [2] used many benchmark video datasets to conduct a comparative study on deep learning techniques for vision-based human activity identification (HAR). Within four different subcategories, they investigated CNNs and Recurrent Neural Networks (RNNs). Along with their experimental datasets and efficacy, they also covered a variety of techniques. In order to demonstrate the developments in HAR, they also contrasted deep learning techniques with manually designed feature-based approaches.

With an emphasis on the psychological aspects of crowds, Sánchez et al. [6] provided models that use Deep Learning for the identification of crowd irregularities. They stressed the need of developing difficult real-world datasets to improve on current approaches and integrating affective aspects into studies on crowd behavior. They emphasized the necessity for precise solutions to keep an eye on crowd behavior and pointed out several opportunities for development.

El-Etriby and colleagues [28] presented a three-phase framework. They started by segmenting the video stream based on background and foreground to create blocks of spatiotemporal flow. Subsequently, in the extracted flow field, samples from every flow block were handled as a two-dimensional distribution. Ultimately, they classified each block clip in the scene as either abnormal or normal by statistically characterizing the abnormalities found using the gradient ascent method.

Using a variety of CNN models, Kuppusamy et al. [17] investigated anomalous behaviors in humans and identified aberrant behaviors in videos. Their analysis showed that 3D CNN outperformed machine learning methods in identifying different types of anomalous human behaviors using a variety of datasets.

Tyagi et al. [22] carried out an extensive evaluation and contrast of several CBA methods. Using various datasets and taxonomies, they performed a comparative examination of current approaches and presented their results. They emphasized the importance of key elements for crowd analysis, such as tracking, recognizing objects, density estimate, crowd counting, and identifying anomalies in crowded scenarios.

A behavior analysis technique based on generative models was presented by Noor et al. [29] to help crowd organizers make well-informed decisions when using IoT services. Their technique comprised the spatiotemporal flow block segmentation of security camera video clips in order to marginalize dense flow fields and categorize them as normal or unusual.

To address the difficulties in crowd evaluation, Sharma et al. [7] presented an integrated approach for crowd density estimate and CBA. They put forth a method to identify anomalous crowd behavior by utilizing motion maps and characteristics derived from the distribution of energy levels.

A multiobject tracking method was created by Petal et al. [30] to reduce identification mistakes, identity switches, and short-term object occlusion. They introduced an adaptive approach based on item dimensions to detect physical distance violations and used a severity degree of loitering to identify anomalous behavior. Their approach was assessed in situations involving pedestrian mobility and traffic.

Patwal et al. [31] covered crowd estimation methods based on deep learning as well as conventional methods. They looked at the best datasets and evaluated the ten most recent studies on deep learning and crowd counting, offering insightful illustrations of the possible applications of AI in this field. Alotibi et al. [32] developed a mobile-based model for people counting in both densely and sparsely populated public spaces, and they modified the last few layers of the VGG-16 model. Their goal was to increase the model's efficiency while taking a variety of scenario variables into consideration.

In order to identify violent behavior, Singh et al. [19] developed a Generative Adversarial Network based on Conditional Autoregressive-Tunicate Swarm Algorithm for crowd behavior identification. To address the issues of high processing power and fluctuation between negative and positive samples, Sampath et al. [11] devised an algorithm that can recognize irregularities in films. The approach uses an unsupervised learning technique called spatiotemporal inter-fused autoencoder to overcome the problem of fewer negative examples. This technology develops a spatiotemporal convolutional neural network (CNN), which has a simple architecture and needs little processing resources. Amrish et al. [33] provided a review of CNNs use in human crowd analysis. Their study addresses the difficulties and constraints of existing approaches and emphasizes CNNs' promise to overcome these constraints. In both structured and unstructured dense crowd scenes, Alhothali et al. [13] suggested a method to identify anomalies. Their approach uses a deep CNN to first identify moving things and people in the scene, then it uses spatial and temporal data to track those objects and people. One of seven classifications is selected by a support vector machine model to categorize anomalous events. Luo et al. [18] conducted a thorough investigation using the multiscale motion consistency network (MSMC-Net) with crowd motion learning framework to address the significant issue of VAD for CABs. Chandrakala et al. [12] emphasized an overview of deep-learning-based video anomaly detection. A thematic taxonomy is provided, comprising four main categories and multiple

subcategories. They concluded that techniques based on adversarial autoencoders and generative adversarial networks offer a higher detection rate. Mahmoudi et al. [4] presented a review and a comparison of deep learning solutions in terms of explainability, memory size, computation time, precision, and flexibility. Based on the kind of data gathered and the needs of the customers, they offered a customized method for identifying harmful actions.

3. Suggested Approach

The primary focus of our work is to develop crowd-flow solutions using advanced modeling techniques. Our specific goal is to design a model that effectively captures and analyzes the collective behavior of crowds by considering their motion patterns over time. By acquiring insights from this crowd behavior model, we aim to detect and recognize significant various classes within the crowd. To achieve this, our proposed approach consists of three crucial components, illustrated in Figure 1.

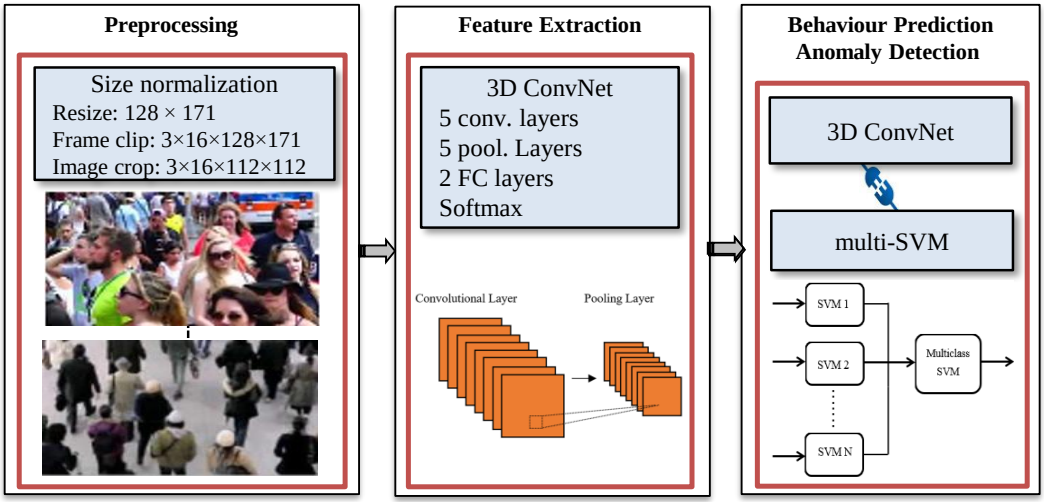

Figure 1. Crowd behavior investigation and analysis: roadmap for an intelligent approach.

3.1. Preprocessing

For this study, we will utilize the publicly available Crowd-11 dataset as a valuable resource [2]. This dataset contains more than 6000 video sequences, with each sequence averaging 100 frames in length with various resolution. To begin, we initiate a preprocessing step where we resize all video frames to dimensions of 128 × 171. The videos are subsequently divided into nonoverlapping clips containing 16 frames each, which serve as input for the network. Additionally, we normalize the image size to 112 pixels × 112 pixels. The choice of the 112 × 112 size is a balanced compromise between real-time performance and recognition accuracy. In the event that the size is insufficiently large, the resulting images would suffer from a loss of vital information, ultimately causing a decrease in accuracy when it comes to recognition. Conversely, if the size were excessively large, the input vector's dimension would significantly increase, resulting in higher computational costs without a significant improvement in recognition accuracy. After conducting experiments, we have determined that the size of 112 × 112 is the optimal choice. Under this setup, the videos are partitioned into nonoverlapping 16-frame clips, which are utilized as input for the networks. Each collection of 16 frames constitutes a processing unit or tensor. As a result, during the training phase, the dimensions of a single tensor are 3 × 16 × 112 × 112, where the value 3 signifies the three color channels (red, green, and blue). A visual depiction of the tensor's structure can be found in Figure 2.

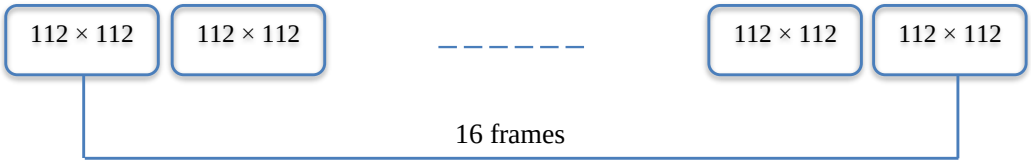

Figure 2. The structure of a single tensor: 3 color channels multiplied by 16 frames, further multiplied by 112 pixels in height and 112 pixels in width.

3.2. Feature Extraction

Three-dimensional CNN (3D ConvNets) relies on the TensorFlow framework, and is constructed for extracting features from video images. It is also known as spatiotemporal ConvNets, and they are specialized neural network architectures that can learn spatiotemporal features from video data [34]. They enable the analysis of volumetric data, find applications in various domains, and offer advantages in tasks such as action recognition, video understanding, medical imaging, and point cloud processing.

The architecture of 3D ConvNets extends that of 2D ConvNets by incorporating an additional dimension for time. This allows them to consider both spatial and temporal dimensions of the input data. The foundational components of 3D ConvNets comprise 3D convolutional layers, pooling layers, and fully connected layers. To extract spatiotemporal features, 3D convolutional layers utilize filters that convolve across the input volume's spatial and temporal dimensions. Lower spatial dimensions and higher-level representations are captured by downsampling the feature maps using pooling layers.

The following equation can be used to get the 3D convolutional layer's output feature map:

$$O = \left(\frac{W - F + 2P}{S}\right) + 1 \tag{1}$$

where O represents the output spatial size of the feature map, W denotes the input spatial size of the feature map, F indicates the spatial size of the convolutional filter (kernel), P signifies the padding size, and S represents the stride size.

The output of a 3D max-pooling layer can be calculated using the following equation:

$$O = \left(\frac{W - F}{S}\right) + 1. \tag{2}$$

Dropout is a regularization technique commonly used in 3D ConvNets. Although there is no specific equation for dropout, it can be represented as follows:

$$\text{Output} = \text{Input} \times \text{Mask}. \tag{3}$$

The input tensor to the dropout layer is denoted as 'Input', and a binary mask called 'Mask' is used, having the same shape as the input tensor. Each element of the mask is set to 1 with probability $(1 - p)$ and 0 with probability p. The mask is randomly generated for each forward pass and applied element-wise to the input tensor. These equations capture the fundamental operations in 3D ConvNets: 3D convolution, 3D max-pooling, and dropout. They determine the output sizes and transformations of the input data as it passes through the network. We conduct an empirical investigation to determine an optimal architecture for 3D CNN (ConvNets). Building upon the insights gained from 2D ConvNets [34], through our observations, we have found that deeper architectures with small receptive fields, utilizing 3×3 convolution kernels, produce the most favorable results. In our architecture search study, we focus on maintaining a fixed spatial receptive field of 3×3 and varying only the temporal depth of the 3D convolution kernels. The networks we employ are designed to process video clips as inputs and predict class labels for 10 different actions. The video frames are uniformly resized to dimensions of 128×171. Subsequently, the videos are partitioned into nonoverlapping 16-frame clips, which serve as input for the networks. The input dimensions for these clips are specified as $3 \times 16 \times 128 \times 171$. To introduce

variation during training, we employ jittering by randomly cropping the input clips to a size of $3 \times 16 \times 112 \times 112$. The choice of a video length of 16 frames in this study has a practical and methodological basis, tied to the balance between computational efficiency and the ability to capture meaningful temporal dynamics in video data. A sequence of 16 frames provides enough temporal depth to capture motion patterns and interactions within the video. This length is sufficient for analyzing common crowd behaviors like merging, diverging, or turbulent flow. Shorter sequences (e.g., 8 frames) may not include enough temporal information, making it difficult to discern complex behaviors. Processing longer sequences (e.g., 32 or 64 frames) significantly increases computational cost due to the larger size of the input tensor. Using 16 frames strikes a balance by keeping computational requirements reasonable while still capturing meaningful temporal dynamics.

Based on our subset of experiments, the 3D ConvNets utilized in our networks consist of a total of five convolution layers and five pooling layers. Each convolution layer is immediately followed by a corresponding pooling layer. In addition to these layers, there are two fully connected layers incorporated, followed by a softmax loss layer responsible for predicting action labels. The five convolution layers in the network are configured with the following number of filters: 64, 128, 256, 256, and 256, respectively. Our findings indicate that for optimal performance in the 3D ConvNets, a $3 \times 3 \times 3$ kernel with a stride of $1 \times 1 \times 1$ is the preferred choice. As for the 3D pooling layers, they have a size of $2 \times 2 \times 2$ with a stride of $2 \times 2 \times 2$, except for the first pooling layer (pool1), which employs a kernel size of $1 \times 2 \times 2$ and a stride of $1 \times 2 \times 2$. The primary objective of this particular arrangement is to retain temporal information in the early stage. Each fully connected layer within the network comprises 4096 output units. The initial fully connected layer extracts a feature output of dimensionality 4096, which is then utilized as the input for the classification model. The architecture of the 3D ConvNets neural network is depicted in Figure 3.

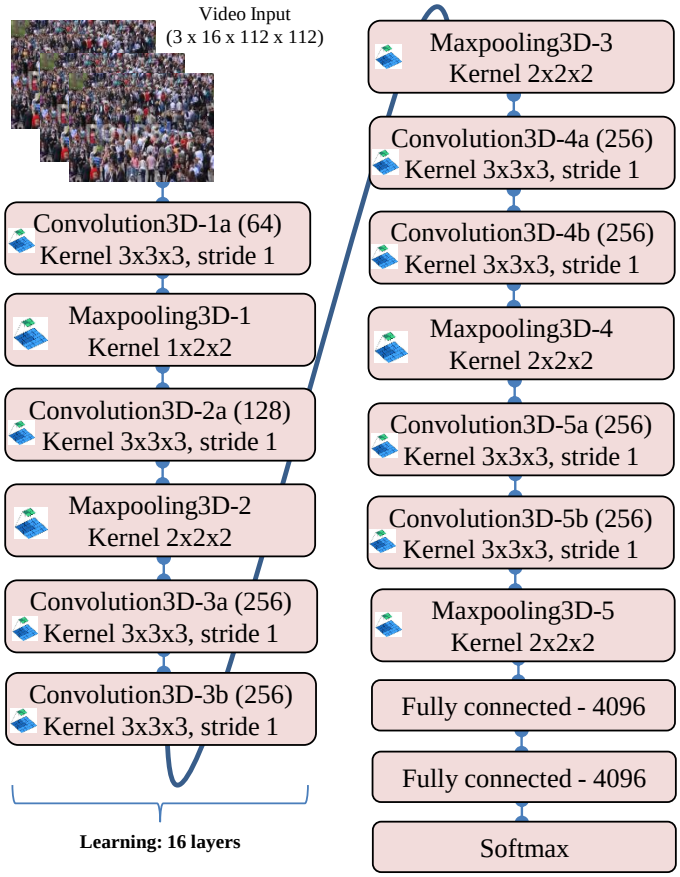

Figure 3. The 3D ConvNet architecture consists of eight convolution layers, five max-pooling layers, and two fully connected layers. Subsequently, a softmax output layer is employed after these layers.

3.3. Behavior Prediction and Anomaly Detection

Combining deep learning architectures like 3D ConvNets with Multi Support Vector Machine (multi-SVM) in a hybrid model offers several advantages. It can potentially improve performance by leveraging the strengths of both approaches, capturing complex patterns and benefiting from SVM's ability to create clear decision boundaries [35,36]. The hybrid model may also enhance generalization, handle imbalanced datasets, reduce training time, and require fewer computational resources. However, the effectiveness of the hybrid model depends on the specific task and dataset, and careful experimentation and fine-tuning are crucial for optimal results. The spatiotemproal features are extracted using 3D ConvNets as inputs to multi-SVM either for training or prediction processes.

Multi-SVM, also known as Multiclass Support Vector Machine, is an extension of the binary Support Vector Machine (SVM) algorithm to handle multiclass classification problems. It is a powerful machine learning technique used for a binary classification to tasks, but it can be modified to handle multiple classes by incorporating strategies like one-vs-one (OvO) or one-vs-all (OvA) (Figure 4). In the multiclass classification problem, the goal is to classify the instances into one of many predefined classes. Multi-SVM achieves this by training multiple binary SVM classifiers, where each classifier is responsible for distinguishing between a pair of classes.

For the approach of OvO, a binary SVM classifier is learned for every pair of classes. For N classes, this results in $N(N-1)/2$ binary classifiers. The decision function for each binary classifier can be represented as:

$$f_{ij}(x) = sign(w_{ij}^T x + b_{ij}) \tag{4}$$

where $f_{ij}(x)$ is the decision function for the binary classifier between classes i and j, x is the input instance, w_{ij} is the weight vector, and b_{ij} is a bias term. The final outcomes for an input instance is identified by a voting scheme, where each classifier contributes a vote based on its predicted class, and the class with the highest number of votes is chosen as the final prediction.

The OvA (One-vs-All) approach involves training a binary Support Vector Machine (SVM) classifier for each class, where each classifier is learned to distinguish that particular class from the rest of the classes. For N classes, this results in N binary classifiers. The decision function for each binary classifier can be represented as:

$$f_i(x) = sign(w_i^T x + b_i) \tag{5}$$

where $f_i(x)$ is the decision function for the binary classifier between class i and the rest, x refers to the input instance, w_i is the weight vector, and b_i is a bias term.

The final outcomes for an input instance is identified by selecting the class with the highest decision value among all the binary classifiers. Both the OvO and OvA approaches leverage the principles of SVM, including the use of kernel functions to model nonlinear decision boundaries. The decision functions can be transformed using a kernel function, such as the radial basis function (RBF) kernel, to handle nonlinear separability. Multi-SVM inherits the strengths of binary SVM, such as the ability to effectively handle feature spaces with high dimensions and nonlinear decision boundaries. However, it also has limitations, including sensitivity to hyperparameters, such as the regularization parameter (C) and the kernel function. Additionally, the OvA approach may struggle with imbalanced class distributions.

The choice of Multi-SVM (Multiclass Support Vector Machine) over other methods, such as decision trees, random forests, or softmax layers, was driven by its superior ability to handle complex, high-dimensional, and nonlinear data, which are typical of spatiotemporal crowd behavior features. Multi-SVM excels in creating clear decision boundaries by maximizing class margins, making it effective for imbalanced datasets where some classes are underrepresented. Decision trees and random forests, while alternatives, struggle with overfitting and require substantial computational resources for high-dimensional data.

Random forests, in particular, may demand many trees to achieve comparable accuracy, increasing model complexity. A softmax layer, though integrated into the deep learning architecture, lacks the flexibility and precision of Multi-SVM in refining decision boundaries, especially in scenarios with intricate class relationships. By separating feature extraction (via 3D ConvNet) from classification (via Multi-SVM), this approach ensures modularity, computational efficiency, and better handling of nuanced decision regions. Multi-SVM's "one-vs-one" or "one-vs-all" strategies further enhance its adaptability, making it the most accurate and efficient choice for classifying complex crowd behavior patterns.

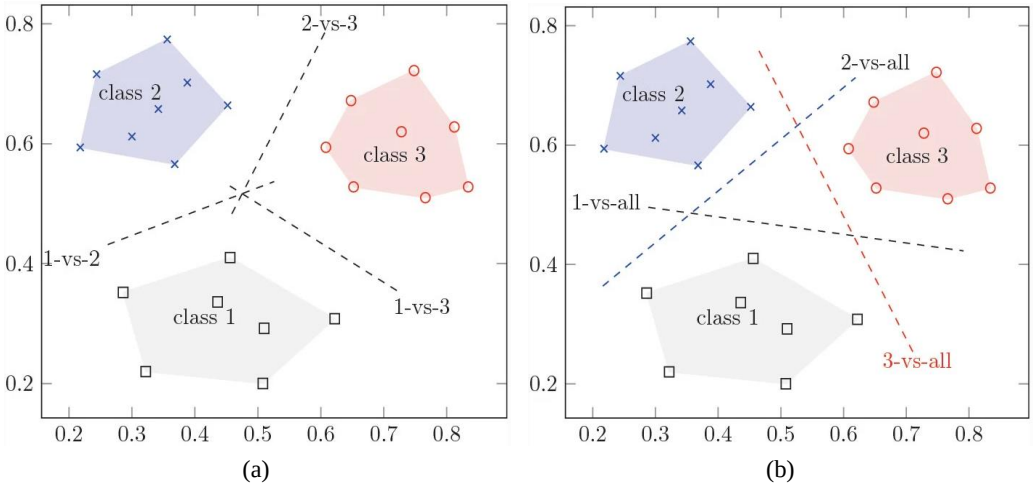

Figure 4. Multiclass Support Vector Machines. (**a**) One-Versus-One. (**b**) One-Versus-All.

4. Experimental Results and Discussions

In this study, we used a public dataset of Crowd-11 https://paperswithcode.com/dataset/crowd11 (accessed on 12 April 2024) [2]. This dataset contains more than 6000 video sequences, with each sequence averaging 100 frames in length. The Crowd-11 dataset consists of over 6000 video sequences, with each sequence having an average length of 100 frames. The dataset was split into three subsets: test, training, and validation. A fixed number of videos from each class were reserved for testing, ensuring that the test set remained independent from the training and validation sets. The remaining videos were divided, with 90% allocated for training and 10% for validation. This splitting process was performed separately for each class to maintain balanced representation. To optimize model performance and assess its generalization ability, we employed a 10-fold cross-validation strategy during the training phase. In this approach, the training set was divided into 10 parts, where each part was used once for validation, and the remaining parts were used for training. This procedure was repeated iteratively to ensure robust performance evaluation. Additionally, to avoid data leakage, videos from the same scene (a specific region observed from a unique perspective) were not shared between the training, validation, and test sets. Preprocessing involved resizing the videos to 112 × 112 pixels and dividing them into nonoverlapping 16-frame clips, which were then used as input for the 3D ConvNet model.

Both the 3D ConvNet architecture and Multi-SVM were trained in a combined, sequential manner as part of a single pipeline. The 3D ConvNet was first trained on the video clips extracted from the training subset of the Crowd-11 dataset. During training, the 3D ConvNet learned spatiotemporal features from the video data. The outputs of the ConvNet (i.e., the learned features) were then used as inputs to the Multi-SVM model for classification. The Multi-SVM was trained on these features to categorize the different crowd behaviors. Both the training of the 3D ConvNet and the Multi-SVM utilized the same training subset to ensure consistent feature extraction and classification. The validation subset was used to tune the hyperparameters of the entire system (ConvNet and Multi-SVM) during training. The final model was evaluated using the test set to assess its

performance on unseen data. This integrated approach, where the deep learning model and Multi-SVM work together within the same pipeline, ensures that the features extracted by the 3D ConvNet are optimally used for classification by the Multi-SVM.

There are two distinct classifications or groupings of crowds that we can identify.

(a) The crowd displays a range of dynamic, densely packed, and efficiently coordinated traits, including smooth, chaotic, intersecting, merging, and diverging flows. These crowds have the ability to effortlessly navigate predefined paths or divide into segments. It is worth emphasizing that the overall description of motion is not affected by the number of flows present in the scene, unless there are instances of interference or distinct behaviors among the groups (Figure 5).

(b) On the other hand, there exist crowds that do not exhibit identifiable streams or cohesive patterns. These crowds can be categorized as nonflowing, congested, motionless, restless, or interacting crowds. Within such gatherings, individuals exhibit independent behavior, wherein their actions do not significantly influence the collective movement of the entire assemblage (Figure 5).



Figure 5. Two distinct classifications of crowds: crowds that are dynamic as in the top figure and crowds with no perceivable streams as as in the bottom figure.

Laminar flow pertains to the synchronized and uninterrupted movement of individuals within a crowd, resembling a seamless stream. This concept is closely associated with the viscosity of the crowd and is primarily observed in dense crowds where individuals have limited space to deviate from their neighbors without disrupting the overall flow. In contrast, turbulent flow refers to the chaotic and irregular movement, akin to the behavior observed in fluid dynamics. Unlike laminar flow, turbulent flow is characterized by unpredictable variations, swirling eddies, and the mingling of individuals within the crowd.

Important traits of turbulent flow are as follows. (1) Unpredictable Motion: The fluid particles move erratically and irregularly in turbulent flow. The vortices, velocities, and directions of the flow are continuously changing. (2) Mixing and Turbulence: Mixing occurs when fluid particles move across various flow levels and zones. More mass, heat, and momentum are transported through the fluid as a result of this mixing. (3) High Energy of Turbulence: Compared to laminar flow, turbulent flow contains more energy. Increased kinetic energy and energy dissipation in the form of heat are the results of the uneven motion and mixing. (4) Broad Velocity Distribution: In contrast to the parabolic velocity profile observed in laminar flow, turbulent flow exhibits a more uniformly distributed velocity across the cross-section. At different points in the flow, the fluid particles' velocities differ dramatically.

When two or more streams of people or crowds meet or cross paths, it is referred to as a "crossing flow". It happens when distinct groups of people traveling in opposite directions come into contact with one another in a same area, possibly resulting in confrontations, interactions, or traffic jams. Crossing flows can happen in a number of places, including

entrances, hallways, crossroads, and any other place where many streams of people must move past one another.

The term "merging flow" describes the occurrence of two or more distinct streams of people or crowds coming together to form a single, unified flow. This convergence happens when various groups of individuals come together at a shared location while traveling in different directions. Combining flow can happen in a variety of places, such as the entry to a building, a crosswalk in a corridor, or a gathering place in a public area. In these situations, disparate crowds or separate groups may combine to form a single, cohesive flow.

A situation when a single stream of people or a crowd splits off or diverges into multiple courses or directions is referred to as a diverging flow. This phenomenon occurs when a cohesive group or stream of individuals splits off into different paths or regions. Diverging flow can occur in a variety of settings, including exits from a building, intersections in a corridor that lead to different locations, and participant dispersal after a meeting or event. When people are widely scattered and their straight lines do not change even though their speed and direction change, this is referred to as a "gas free" environment. On the other hand, in a gas-jammed scenario, there is a great deal of congestion and people's trajectories clash. Two types of crowd behavior can be distinguished from each other. The first group is still, like a throng standing by or observing, whereas the second group includes those who are moving, like cheering or dancing. People move violently and erratically toward one another when they are in an interactive crowd. The background, an eleventh class, is also introduced to depict spaces that might include other dynamic elements but do not have any crowds.

To ensure reliable testing, the Crowd-11 dataset, which is sufficiently large, is split into three subsets. These subsets are utilized for learning the model, conducting cross-validation, and performing test evaluations (Table 1). This division was done to ensure that we have sufficient data for both training our model and evaluating its performance. Thus, it allows us to assess the model's performance and make informed decisions about its effectiveness and potential for real-world applications. For each class C_i, the test, train, and validation processes are determined as follows: the test split contains m videos, the train split contains $0.9 \times (NC_i - m)$ videos, and the validation split contains $0.1 \times (NC_i - m)$ videos. Here, m represents a fixed test size applied to all classes and NC_i represents video numbers in a particular class. The objective of this approach is to ensure a balanced evaluation of the classes. To maintain the model's ability to generalize, scenes from the same class are not shared between the three sets. A scene is characterized as a unique region observed from a particular perspective.

Table 1. Crowd-11 database with respect to the number of videos either for training or testing.

Label	Class Names	# Videos Training	# Videos Prediction
0	Laminar Flow	870	434
1	Turbulent Flow	595	297
2	Crossing Flows	508	255
3	Merging Flow	197	98
4	Diverging Flow	122	62
5	Gas Free	352	177
6	Gas Jammed	346	174
7	Static Calm	491	246
8	Static Agitated	273	137
9	Interacting Crowd	165	83
10	No Crowd	260	130

Upon examining the findings obtained with the 2D ConvNet [34], it is found that the best outcomes are obtained with deeper designs with tiny receptive fields using 3×3 convolution kernels. Therefore, we keep the spatial receptive field of 3×3 constant in our architecture search investigation and just change the temporal depth of the 3D convolution kernels.

In the interest of clarity, from now on, we will refer to these measurements as $c \times l \times h \times w$, where c is the number of channels, l is the length in frames, and h and w are the height and width of each frame, respectively. Additionally, the dimensions of 3D convolution and pooling kernels will be expressed as $d \times k \times k$, where d denotes the kernel's temporal depth and k its spatial size.

The network architectures are specifically designed to take in video clips as inputs and produce class label predictions based on a set of 11 different actions. To achieve input standardization, every video frame is shrunk to 128×171, or roughly half the resolution of the crowd-11 frames. Once the videos have been divided into nonoverlapping 16-frame chunks, the networks can use them as inputs. Consequently, $3 \times 16 \times 128 \times 171$ is the network's input dimension. During the training process, we employ random cropping to resize the input clips to a size of $3 \times 16 \times 112 \times 112$. The network architecture comprises of five convolution layers, each followed by a pooling layer. There are five repetitions of this pattern. For predicting action labels, there are also two fully connected layers and a softmax loss layer. The five convolution layers use 64–256 different filters. Specifically, the number of filters for each layer is as follows: 64, 128, 256, 256, and 256, respectively.

Within the architectural design, the convolution kernels are uniformly sized with a value of d, representing the temporal depth of the kernel. The specific value of d will be fine-tuned later to achieve an optimal 3D architecture. To maintain consistency in input and output sizes, the convolution layers are applied with appropriate padding (both spatial and temporal) and a stride of 1. This ensures that the dimensions of the input and output remain unaltered throughout the convolutional process. The architecture incorporates max pooling layers with a kernel size of $2 \times 2 \times 2$ for all layers except the initial one. These pooling layers employ a stride of 1, resulting in an output signal size that is reduced by a factor of 8 compared to the input signal. However, the first pooling layer has a distinct kernel size of $1 \times 2 \times 2$. The specific kernel size of $1 \times 2 \times 2$ is selected to prevent premature merging of the temporal signal and to accommodate the clip length of 16 frames, as depicted in Figure 3. This choice allows for a maximum of four temporal pooling operations to be executed before the temporal signal is entirely collapsed. The architecture comprises two fully connected layers, both with 4096 outputs. The training process commences from scratch and employs mini-batches comprising 30 clips. An initial learning rate of 0.003 is set, and after every 4 epochs, the learning rate is divided by 10. The optimization process concludes at approximately 2.1 million iterations, approximately equivalent to 14 epochs.

We examine the effects of changing the temporal depth of the kernels between layers in our studies. In particular, we investigate four distinct networks with uniform environments. The term depth-d refers to the temporal depth of these networks. The temporal depths of the kernels in the networks are 1, 3, 5, and 7, respectively. It is important to remember that the depth-1 network is the same as performing separate 2D convolutions to every frame. We also test networks with different temporal depths in addition to the homogeneous scenario. We specifically look into two networks: one that has a decreasing temporal depth pattern of 7-5-5-3-3 and another that has a rising temporal depth pattern of 3-3-5-5-7 from the first to the fifth convolution layer. It is noteworthy to notice that at the final pooling layer, all of these networks share the same output signal size. Because of this, the fully connected layers' parameter count is constant throughout these networks. Because of the different kernel temporal depth, the differences in the number of parameters mostly happen in the convolution layers. These variations in parameter counts, however, pale in comparison to the millions of parameters seen in fully connected layers.

We conduct experiments to investigate the effects of different temporal depths by altering the kernel temporal depth between the layers. We study four networks designated

as depth-d in the homogeneous context, where d is the homogeneous temporal depth. The temporal depths of the kernels in these networks are 1, 3, 5, and 7, respectively. Interestingly, the depth-1 network is the same as doing separate 2D convolutions on every frame. In the context of varying temporal depth, we also examine two additional networks. One network has an increasing temporal depth pattern of 3-3-5-5-7 across the first to the fifth convolution layers, while the other network has a decreasing temporal depth pattern of 7-5-5-3-3. It is worth mentioning that all of these networks produce output signals of the same size at the last pooling layer, resulting in an equal number of parameters for the fully connected layers. During our experimentation, the variations in the number of parameters are mainly concentrated in the convolution layers as a result of the different kernel temporal depths. It is noteworthy, however, that these distinctions pale in comparison to the millions of characteristics found in the completely connected layers. With temporal depth-3, the 3D ConvNet is able to effectively learn spatiotemporal features that capture the key transitions and interactions within crowd behavior. Depth-1, which treats each frame independently, cannot capture the motion dynamics as effectively, while deeper depths might start to blend or lose focus on key behavioral changes within the crowd. Temporal depth-3 allows the model to retain critical temporal information while maintaining computational efficiency. Longer sequences may increase the computational burden, reducing the model's ability to generalize due to higher risk of overfitting.

Based on Figure 6, it is evident that among the homogeneous networks, Depth-3 achieves the best performance. Notably, Depth-1 performs noticeably worse than the other networks, which we believe is due to its restricted capacity to simulate motion. Depth-3 is the best performing network when compared to the networks with different temporal depth, even though the performance difference is smaller. We also experimented with using the full input resolution and/or extending the spatial receptive field (e.g., 5×5) and saw comparable outcomes. This indicates that the choice of a $3 \times 3 \times 3$ kernel for 3D ConvNets is optimal based on our subset of experiments, and consistently outperforms 2D ConvNets for video classification. After extraction the features using 3D ConvNets, we employ them as input to a multiclass linear SVM for learning models or testing.

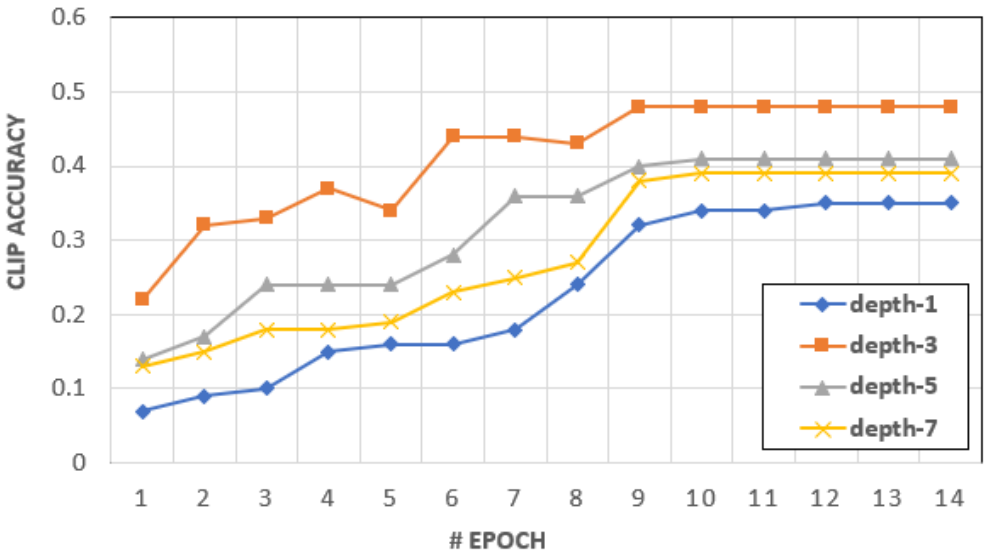

Figure 6. The performance of networks with varying kernel temporal depths.

The generalization ability and applicability of the model across different temporal depths (1, 3, 5, and 7) depend on the nature of the crowd behaviors and the specific scenarios. Temporal depth-1, which processes each frame independently, is best suited for static or relatively simple crowd behaviors, such as Static Calm or Static Agitated crowds, where individual movements are less influenced by surrounding people. However, it struggles with more dynamic behaviors like Merging Flow or Crossing Flows, which involve com-

plex interactions between different groups. Temporal depth-3 strikes an optimal balance, capturing enough temporal context to model moderate to high-density behaviors such as Turbulent Flow and Merging Flow, while remaining computationally efficient. This depth generalizes well in scenarios like public event monitoring, transportation hubs, and crowd safety surveillance, where crowd interactions are important, but computational efficiency is also critical. Temporal depths 5 and 7 capture more extensive temporal dynamics, which can be beneficial for highly complex behaviors and environments with longer-term crowd interactions, such as Interactive Crowds or Gas Jammed scenarios. However, these deeper depths increase computational costs and may lead to overfitting, particularly in environments with limited training data. While depth-3 is well-suited for a variety of applications, deeper temporal depths can be valuable for detecting subtle variations in crowd behavior over time, especially in crowded or chaotic environments.

Table 2 displays the outcomes obtained from our utilization of 3D ConvNet and multi-SVM on the Crowd-11 database, considering different depths. Our approach involves employing a single center crop per clip, which is then passed through the network for clip prediction. To make predictions for videos, we average the clip predictions of 10 randomly extracted clips from each video. Among the networks with varying temporal depths, depth-3 achieves the highest accuracy, reaching 89.8%. Although there is a slight difference in performance between depth-5 and depth-7, it is not substantial.

Table 2. Results obtained from the integration of 3D ConvNet and multi-SVM with respect to various depths on the Crowd-11 database.

Network	Accuracy	
	Per Clip	**Per Video**
3D ConvNet + multi-SVM (depth 1)	64.8%	68.7%
3D ConvNet + multi-SVM (depth 3)	86.3%	89.8%
3D ConvNet + multi-SVM (depth 5)	80.7%	82.6%
3D ConvNet + multi-SVM (depth 7)	78.5%	81.9%

Figure 7 explores the confusion matrix results for the dataset crowd-11 using 3D ConvNets and multi-SVM. Certainly, the classification of merging dense flows can often lead to confusion with other dense or static crowds, primarily due to their slow movement. The difficulty arises from the fact that diverging flows, which are less common, do not always exhibit a consistent increase in the magnitude of the velocity field. There have been observations of misclassifications between different flow types, specifically Laminar Flow and Merging Flow, Crossing Flows and Merging Flow, Gas Free and Laminar Flow, and Gas Jammed and Gas Free. Figure 7 presents the confusion matrix for the 3D ConvNet and Multi-SVM model at temporal depth-3, providing a clear visualization of the classification performance across the 11 crowd behavior categories. Categories such as Laminar Flow and Gas Free demonstrate high classification accuracy, likely due to their distinct and easily recognizable movement patterns. Laminar Flow, characterized by smooth and uniform movement, is easy for the model to distinguish, while Gas Free, which involves individuals moving without significant interference, also shows clear patterns that the model can effectively identify. However, other categories, like Merging Flow and Crossing Flows, show lower accuracy, with frequent misclassifications between them. This is likely due to the similarities in their complex crowd dynamics, where multiple groups of people interact or cross paths, creating visual and temporal overlap that the model struggles to differentiate. Additionally, behaviors such as Turbulent Flow and Static Agitated also exhibit misclassification, possibly because both involve chaotic or disordered movements, making them harder for the model to distinguish accurately. The proposed approach, employing 3D ConvNets and multi-SVM with a depth of 3, achieved a recognition rate of 89.8%. We assess our method using the accuracy metric found in Equation (6). A measure

of a model's total prediction accuracy is called accuracy, which is determined by dividing the total number of samples by the percentage of accurately predicted samples (including true positives and true negatives):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

True Positive (*TP*) denote the number of successfully predicted positive samples in the context of accuracy measurement, whereas True Negatives (*TN*) denote the number of correctly anticipated negative samples. On the other hand, False Negatives (*FN*) show how many falsely projected negative samples there are, while False Positives (*FP*) show how many falsely predicted positive samples there are.

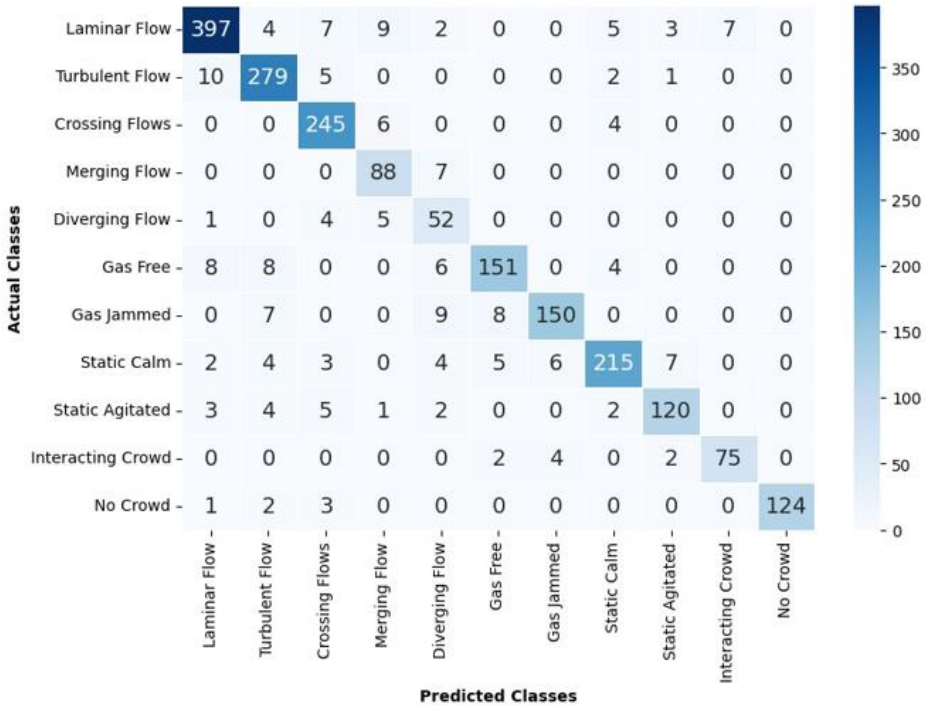

Figure 7. Confusion3 matrix for the dataset crowd-11 using 3D ConvNets and multi-SVM with respect to depth-3.

C. Dupont et al. [2] employed two approaches (networks), V3G and C3D, to predict the classes of the dataset crowd-11 (Table 3). The C3D approach utilizes a sequence of 3D convolutions, where the temporal stack of images forming a clip is represented by the third dimension of the input. This approach involves a network architecture comprised of five layers of 3D convolutions, followed by 3D pooling operations. Subsequently, three Fully Connected (FC) layers are employed in the network. V3G, on the other hand, is a network specifically designed for analyzing crowd behavior. This approach combines the fundamental concepts of both C3D and VGG networks [37], incorporating Batch Normalization (BN) [38] as well. The structure of V3G follows that of the VGG network, but with some key differences. Batch Normalization (BN) layers are integrated after each 3D pooling layer in the C3D technique, which also includes 3D convolutional layers. Besides, the three Fully Connected (FC) layers are transformed into 1×1 convolution layers. A high dropout ratio of 0.8 is used to avoid overfitting.

For the C3D convolution method, the model was trained on the Crowd-11 database, achieving an accuracy score of 46.9%. Additionally, accuracy scores were computed on a per-video basis by averaging the scores of overlapped clips throughout the entire video, resulting in a score of 47.4%. With the V3G model, 57.8% per-clip and 59.4% per-video accuracy were attained by a version that was exclusively trained. The two-stream model

produced poorer results than the top C3D fine-tuned model, nevertheless. The improved action time variation modeling made possible by temporal-wise convolutions carried out in 3D space within the C3D fine-tuned model is responsible for this. To further enhance its performance, the C3D model uses an extended temporal window and early fusion of appearance and temporal information, both of which are well-captured by the network.

Table 3. Comparing the results of our suggested approach with those of tests using a comparable crowd-11 dataset.

Network	Accuracy per Clip	Accuracy per Video
C3D Crowd-11 [2]	46.9%	47.4%
V3G Crowd-11 [2]	57.8%	59.4%
Our Method	86.3%	89.8%

When compared our method with the methods mentioned in Table 3 by C. Dupont et al. [2] , the results is promising due to the suggested approach of 3D ConvNet + multi-SVM with respect to depth 3. Our method exclusively achieved a per-clip accuracy of 86.3% and a per-video accuracy of 89.8%. The comparative analysis demonstrates the effectiveness of the proposed hybrid model, which integrates 3D ConvNet for spatiotemporal feature extraction with Multi-SVM for classification. On the Crowd-11 dataset, the hybrid model achieves a per clip accuracy of 86.3% and a per video accuracy of 89.8%, outperforming several state-of-the-art approaches. When compared to fully end-to-end models, such as C3D and V3G, the hybrid model shows significant advantages. The C3D model, which employs 3D convolutional layers to process video clips, achieves a per clip accuracy of 46.9% and a per video accuracy of 47.4%, struggling to generalize well due to its lack of specialized classification capabilities. The V3G model, which combines features from C3D and VGG architectures, improves performance to 57.8% per clip and 59.4% per video, but it still falls short of the hybrid model due to its reliance on a single end-to-end architecture, which can be less robust when dealing with complex crowd behaviors. The hybrid model's strength lies in its separation of feature extraction and classification, allowing Multi-SVM to provide robust decision boundaries that enhance the overall generalization ability. This is particularly evident in the ablation study, where the standalone 3D ConvNet with temporal depth-3 achieves a per clip accuracy of 79.5%, indicating a noticeable decline when Multi-SVM is removed. Similarly, the study of different temporal depths confirms that depth-3 provides the optimal balance of temporal context, achieving the highest accuracy (89.8%) compared to depth-1 (68.7%), which fails to capture sufficient temporal dynamics, and depth-7 (81.9%), which introduces computational inefficiency and risks overfitting.

The hybrid model used in this study combines 3D ConvNet for feature extraction and Multi-SVM for classification, offering distinct advantages compared to end-to-end network models. One key benefit is improved generalization, as the 3D ConvNet focuses on learning spatiotemporal features, while the SVM excels in creating robust decision boundaries, especially for smaller or imbalanced datasets. This separation allows each component to leverage its strengths independently. Additionally, the hybrid model provides flexibility, enabling the SVM to adapt to different classification tasks without significant retraining. It also mitigates overfitting risks by training the feature extractor and classifier separately, making it particularly effective for datasets of limited size. The computational efficiency of the hybrid model is another advantage, as it requires training only the SVM after feature extraction, unlike end-to-end models that require backpropagation through the entire network. Furthermore, the hybrid approach enhances interpretability by allowing independent analysis of the extracted features, which helps understand the model's focus. These advantages make the hybrid model a compelling choice, particularly in scenarios where dataset size, computational resources, or task-specific needs limit the feasibility of end-to-end models.

5. Conclusions

Our research addresses the challenging task of understanding crowd behavior in highly dense crowd videos by proposing an approach that combines 3D ConvNet and Multi-SVM considering the spatiotemporal characteristics of crowd movement. By incorporating the third dimension as a temporal stack of images forming a clip, the network learns to capture the dynamics and patterns of crowd behavior over time.

The approach effectively leverages the temporal information captured by 3D ConvNet and incorporating varying temporal depths 1, 3, 5 and 7. Furthermore, the investigation of varying temporal depth by changing the kernel temporal depth across the layers provides insights into the impact of different temporal depths on the network's performance in capturing and interpreting crowd behavior patterns. By conducting experiments on the Crowd-11 dataset, which encompasses a wide range of crowd motion patterns in over 6000 video sequences, our proposed approach showcases compelling recognition rates when compared to other existing studies. Notably, at depth-3, our approach achieves an impressive accuracy of 89.8%. These outcomes serve as strong evidence of the method's effectiveness in comprehending and analyzing crowd behavior within densely populated crowd videos. By successfully combining ConvNet and Multi-SVM, our approach enables the identification and categorization of various crowd dynamics, including merging, diverging, and dense flows. This comprehensive analysis provides valuable insights into the complex interactions and dynamics within crowded environments. Future work will focus on integrating contextual information, fine-grained behavior classification, real-time monitoring, and multimodal data fusion to enhance the understanding and analysis of crowd behavior in highly dense crowd videos.

Author Contributions: Conceptualization, M.E. and A.S.M.; methodology, M.E. and M.A.; software, M.A. and A.S.M.; validation, M.E., A.S.M. and H.M.I.; formal analysis, M.A.; investigation, H.M.I.; resources, H.M.I.; data curation, M.A.; writing—original draft preparation, M.F.; writing—review and editing, M.E. and A.S.M.; visualization, M.F.; supervision, M.E. and H.M.I.; project administration, A.S.M.; funding acquisition, A.S.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by [Deputyshipp for Research and Innovation, Ministry of Education in Saudi Arabia] project number 442-162.

Data Availability Statement: The data presented in this study are openly available in [Crowd 11-Kaggle] at [DOI: 10.1109/CVPRW.2017.271], reference number [2].

Acknowledgments: The authors would like to extend their appreciation to Taibah University for its supervision support.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Sharma, V.; Gupta, M.; Pandey, A.K.; Mishra, D.; Kumar, A. A Review of Deep Learning-based Human Activity Recognition on Benchmark Video Datasets. *Appl. Artif. Intell.* **2022**, *36*, 2093705. [CrossRef]
2. Dupont, C.; Tobias, L.; Luvison, B. Crowd-11: A Dataset for Fine Grained Crowd Behaviour Analysis. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 21–26 July 2017. [CrossRef]
3. Rezaei, F.; Yazdi, M. Real-time crowd behavior recognition in surveillance videos based on deep learning methods. *J. Real-Time Image Process.* **2021**, *18*, 1669–1679. [CrossRef]
4. Mahmoudi, S.A.; Amel, O.; Stassin, S.; Liagre, M.; Benkedadra, M.; Mancas, M. A Review and Comparative Study of Explainable Deep Learning Models Applied on Action Recognition in Real Time. *Electronics* **2023**, *12*, 2027. [CrossRef]
5. Mu, M. Network as a Sensor for Smart Crowd Analysis and Service Improvement. *IEEE Netw.* **2023**, *37*, 144–152. [CrossRef]
6. Sánchez, F.L.; Hupont, I.; Tabik, S.; Herrera, F. Revisiting crowd behaviour analysis through deep learning: Taxonomy, anomaly detection, crowd emotions, datasets, opportunities and prospects. *Inf. Fusion* **2020**, *64*, 318–335. [CrossRef]
7. Sharma, V.; Mir, R.N.; Singh, C. Scale-aware CNN for crowd density estimation and crowd behavior analysis. *Comput. Electr. Eng.* **2023**, *106*, 108569. [CrossRef]

8. Varghese, E.B.; Thampi, S.M. A Comprehensive Review of Crowd Behavior and Social Group Analysis Techniques in Smart Surveillance. In *Intelligent Image and Video Analytics*; CRC Press: Boca Raton, FL, USA, 2023; pp. 57–84.
9. Bhardwaj, S.; Dwivedi, A.; Pandey, A.; Perwej, Y.; Khan, P.R. Machine Learning-Based Crowd behavior Analysis and Forecasting. *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.* **2023**, *9*, 418–429. [CrossRef]
10. Chaudhary, D.; Kumar, S.; Dhaka, V.S. Video based human crowd analysis using machine learning: A survey. *Comput. Methods Biomech. Biomed. Eng. Imaging Vis.* **2022**, *10*, 113–131. [CrossRef]
11. Sampath, D.K.; Kumar, K. Abnormal Crowd Behaviour Detection in Surveillance Videos Using Spatiotemporal Inter-Fused Autoencoder. *Int. J. Intell. Eng. Syst.* **2023**, *16*, 470–481.
12. Chandrakala, S.; Deepak, K.; Revathy, G. Anomaly detection in surveillance videos: A thematic taxonomy of deep models, review and performance analysis. *Artif. Intell. Rev.* **2023**, *56*, 3319–3368. [CrossRef]
13. Alhothali, A.; Balabid, A.; Alharthi, R.; Alzahrani, B.; Alotaibi, R.; Barnawi, A. Anomalous event detection and localization in dense crowd scenes. *Multimed. Tools Appl.* **2023**, *82*, 15673–15694. [CrossRef]
14. Alia, A.; Maree, M.; Chraibi, M. A Hybrid Deep Learning and Visualization Framework for Pushing Behavior Detection in Pedestrian Dynamics. *Sensors* **2022**, *22*, 4040. [CrossRef] [PubMed]
15. Aldayri, A.; Albattah, W. A deep learning approach for anomaly detection in large-scale Hajj crowds. *Vis. Comput.* **2023**, *40*, 5589–5603. [CrossRef]
16. Lazaridis, L.; Dimou, A.; Daras, P. Abnormal Behavior Detection in Crowded Scenes Using Density Heatmaps and Optical Flow. In Proceedings of the 2018 26th European Signal Processing Conference (EUSIPCO), Rome, Italy, 3–7 September 2018. [CrossRef]
17. Kuppusamy, P.; Bharathi, V. Human abnormal behavior detection using CNNs in crowded and uncrowded surveillance—A survey. *Meas. Sens.* **2022**, *24*, 1005105. [CrossRef]
18. Luo, L.; Li, Y.; Yin, H.; Xie, S.; Hu, R.; Cai, W. Crowd-level abnormal behavior detection via multi-scale motion consistency learning. In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2023; Volume 37, pp. 8984–8992.
19. Singh, J.P.; Kumar, M. Conditional autoregressive-tunicate swarm algorithm based generative adversarial network for violent crowd behavior recognition. *Artif. Intell. Rev.* **2023**, *56* (Suppl. 2), 2099–2123. [CrossRef]
20. Abdullah, F.; Abdelhaq, M.; Alsaqour, R.; Alatiyyah, M.H.; Alnowaiser, K.; Alotaibi, S.S.; Park, J. Context aware crowd tracking and anomaly detection via deep learning and social force model. *IEEE Access* **2023**, *11*, 75884–75898. [CrossRef]
21. Jadhav, C.; Ramteke, R.; Somkunwar, R.K. Smart Crowd Monitoring and Suspicious Behavior Detection Using Deep Learning. *Rev. D'Intell. Artif.* **2023**, *37*, 955–62. [CrossRef]
22. Tyagi, B.; Nigam, S.; Singh, R. A Review of Deep Learning Techniques for Crowd Behavior Analysis. *Arch. Comput. Methods Eng.* **2022**, *29*, 5427–5455. [CrossRef]
23. Sharath Kumar, Y.; Naveena, C. A Deep Learning Based System to Estimate Crowd and Detect Violence in Videos. In *Artificial Intelligence for Societal Issues*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 45–57.
24. Bouhlel, F.; Mliki, H.; Hammami, M. Crowd Behavior Analysis based on Convolutional Neural Network: Social Distancing Control COVID-19. In Proceedings of the 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, Virtual Event, 8–10 February 2021. [CrossRef]
25. Bhushanam, P.N.; Selva Kumar, S. A Comprehensive Analysis on Unconstraint Video Analysis Using Deep Learning Approaches. In Proceedings of the 2023 2nd International Conference on Edge Computing and Applications (ICECAA), Namakkal, India, 19–21 July 2023; pp. 563–567.
26. Bhuiyan, M.R.; Abdullah, J.; Hashim, N.; Al Farid, F. Video analytics using deep learning for crowd analysis: A review. *Multimed. Tools Appl.* **2022**, *81*, 27895–27922. [CrossRef]
27. Tripathy, S.K.; Kostha, H.; Srivastava, R. TS-MDA: Two-stream multiscale deep architecture for crowd behavior prediction. *Multimed. Syst.* **2023**, *29*, 15–31. [CrossRef]
28. El-Etriby, S.; Miraoui, M.; Elmezain, M. Crowd Behavior Analysis Using Discriminative Models in Video Sequences. *J. Comput. Theor. Nanosci.* **2017**, *14*, 2706–2714. [CrossRef]
29. Noor, T.H. Behavior Analysis-Based IoT Services For Crowd Management. *Comput. J.* **2022**, *66*, 2208–2219. [CrossRef]
30. Patel, A.S.; Vyas, R.; Vyas, O.P.; Ojha, M.; Tiwari, V. Motion-compensated online object tracking for activity detection and crowd behavior analysis. *Vis. Comput.* **2022**, *39*, 2127–2147. [CrossRef]
31. Patwal, A.; Diwakar, M.; Tripathi, V.; Singh, P. Crowd counting analysis using deep learning: A critical review. *Procedia Comput. Sci.* **2023**, *218*, 2448–2458. [CrossRef]
32. Alotibi, M.H.; Jarraya, S.K.; Ali, M.S.; Moria, K. CNN-Based Crowd Counting Through IoT: Application For Saudi Public Places. *Procedia Comput. Sci.* **2019**, *163*, 134–144. [CrossRef]
33. Amrish.; Arya, S.; Kumar, S. Convolutional neural network for human crowd analysis: A review. *Multimed. Tools Appl.* **2023**, *83*, 62307–62331. [CrossRef]
34. Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning Spatiotemporal Features with 3D Convolutional Networks. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015. [CrossRef]
35. Elmezain, M.; Mahmoud, A.; Mosa, D.T.; Said, W. Brain Tumor Segmentation Using Deep Capsule Network and Latent-Dynamic Conditional Random Fields. *J. Imaging* **2022**, *8*, 190. [CrossRef]

36. Elmezain, M.; Malki, A.; Gad, I.; Atlam, E.S. Hybrid deep learning model-based prediction of images related to cyberbullying. *Int. J. Appl. Math. Comput. Sci.* **2022**, *32*, 323–334. [CrossRef]
37. Simonyan, K.; Zisserman, A. Two-stream convolutional networks for action recognition in videos. In Proceedings of the 27th International Conference on Neural Information Processing Systems, Montreal, QU, Canada, 8–13 December 2014; NIPS'14; MIT Press: Cambridge, MA, USA, 2014; Volume 1, pp. 568–576.
38. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the 32nd International Conference on International Conference on Machine Learning, Lille, France, 6–11 July 2015; Volume 37, pp. 448–456.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.