Endsemester Evaluation

AI3011: Machine Learning & Pattern Recognition

Instructor: Prof. Siddharth S.

Drone-based Crowd Surveillance

Chanakya Rao Korati, Moksh Soni & Vaibhav Chopra

The Problem

- Across the world, public law & order is a major bone of contention in politics. Therefore, one would expect that successive governments, irrespective of their standing on the political spectrum, would have sought to better policing norms and infrastructure.
- However, such is not the case.
- Law Enforcement Organizations often find their resources spread thin, and manpower spread even thinner.
- Hence, it is not feasible for them to permanently deploy personnel at every crowd formation just in case they become violent/dense.

The Problem

- It is also not feasible, and in fact against the fundamental directive of any LEO, to refrain completely from sending any forces at all to such crowd formations.
- Some middle grounds would be periodic interventions or the posting of a single policeman in crowds across the country.
- These have their own problems as, if the crowd were to turn violent, a single policeman wouldn't be able to do anything, and violence would have escalated beyond any point of peaceful redressal in the 35-40 minutes it takes for reinforcements to arrive in some parts of the country.

The Problem

- Oftentimes, protests are against the police themselves, therefore their very presence is enough to escalate violence.
- Peaceful protests must be allowed in a free nation, but it must not infringe on public safety. When such protests turn violent, action must be taken, and it must be taken swiftly.
- However, even in this Digitial age, not everything is filmed. Therefore, oftentimes, the case arises that the police write one thing in their FIRs and the defense insists another series of event had transpired.
- In such a situation, the decision is left to the deliberation of the courts and the mood of the judge.

Why?

- Necessity is the mother of invention, and there is nothing in this world more necessary than preserving innocent life. That is the objective of the policeman who swears an oath to serve and protect. That is the directive of the government one elects to safeguard one's rights and one's life.
- An efficacious deployment of our solutions will help in deescalating and, in fact, diffusing, violence in crowds. This will save countless lives. The manpower & resources freed can then be dedicated to other areas, thereby further bettering the law and order situation.

The Solution

- Drones have the ability to access area that cannot be manned on foot. Also, one drone operator can cover much more area than a constable on foot.
- Drones, if designed well, at sufficient heights, blend into the environment and are virtually inconspicuous.
- They are also not very costly in the sense that the cost is a one-time investment that pays for itself in man hours and resources saved over its lifetime.
- Drones are also being equipped with increasingly high computational power, enough to handle ML models [].

- [1] “DroneNet: Crowd Density Estimation using Self-ONNs for Drones This publication was made possible by the PDRA award PDRA7-0606-21012 from the Qatar National Research Fund (a member of The Qatar Foundation). The statements made herein are solely the responsibility of the authors.,” ar5iv. Accessed: Mar. 15, 2024. [Online]. Available: https://ar5iv.labs.arxiv.org/html/2211.07137

The Solution

- Fundamentally, there are two evolutions, rather devolutions if you're talking socially and not mathematically that can arise from a crowd: a mob and a stampede. One has to do with violence and the other with sparsity.
- Therefore, we devised two models that can check whether a crowd is too dense or if it's violent. An immediate message is then sent to the relevant policing authority, who can then dispatch forces to mitigate the crowd.
- Hence, resources and manpower can be dedicated to tasks more efficiently without having to constantly worry about escalation.

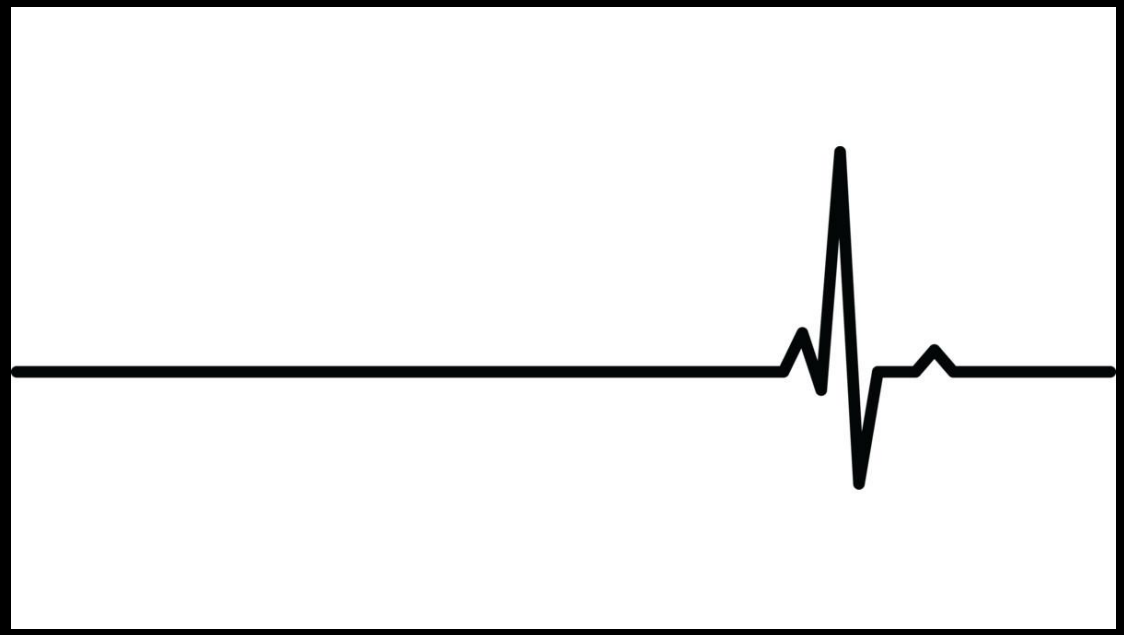
Our Project

- Solves critical issues related to public safety and law enforcement efficiency through advanced drone surveillance and machine learning techniques.
- Focusses on real-time threat detection, violence prediction & crowd density estimation.
- Enables managing public gatherings more safely and effectively, predicting and preventing potential violence, and ensuring efficient law enforcement responses.

Violence Detection

By correctly analyzing the onset of violence in a crowd in a time-sensitive manner, our project will save lives by enabling crucial interventions.

LEOs will be notified if certain thresholds are exceeded.



Crowd Density Estimation

- Understanding crowd density is vital for crowd management.
- Strategies must be designed, and protocols implemented to ensure the mob (and the instigators) are contained without the loss of innocent life.
- Even non-violent crowds must be dispersed before they escalate into stampedes, causing significant Loss of Life.

Potential Impact

- Significantly enhances the operational capabilities of law enforcement and emergency response teams, improve public safety, and potentially save lives by preventing violence and efficiently managing crowds.
- Drones, amongst other technologies, will prove crucial in maintaining law and order as we enter the later stages of the digital era [2].
- Leverages the ever-growing computational and capture technologies in drones in conjunction with newer, faster, more accurate ML models in order to ensure good public safety.
- Commercially viable solution, as demonstrated by Vigilant Solutions' ALPR.

[2] E. F. PhD, “17 Types of Innovative Police Technology,” University of San Diego Online Degrees. Accessed: Mar. 15, 2024. [Online]. Available: https://onlinedegrees.sandiego.edu/10-innovative-police-technologies/

Literature Review

Literature Review

- **Recent developments in drone and ML technology have made substantial headway in assisting police forces in mitigating mob violence**. Researchers have been actively exploring the integration of drones in existing Law Enforcement infrastructure to provide real-time intelligence and situational awareness to LEOs.
- Modern ML algorithms (trained on rather vast datasets ranging from almost 20,000 videos pulled from sources all the way from YouTube to CCTV footage in Shanghai) have shown rather promising results.
- This allows for proactive measures to be taken.
- **While these computational and hardware advancements do offer significant improvements in policing strategies, there remains significant room for refinement and enhancement in leveraging them effectively.**

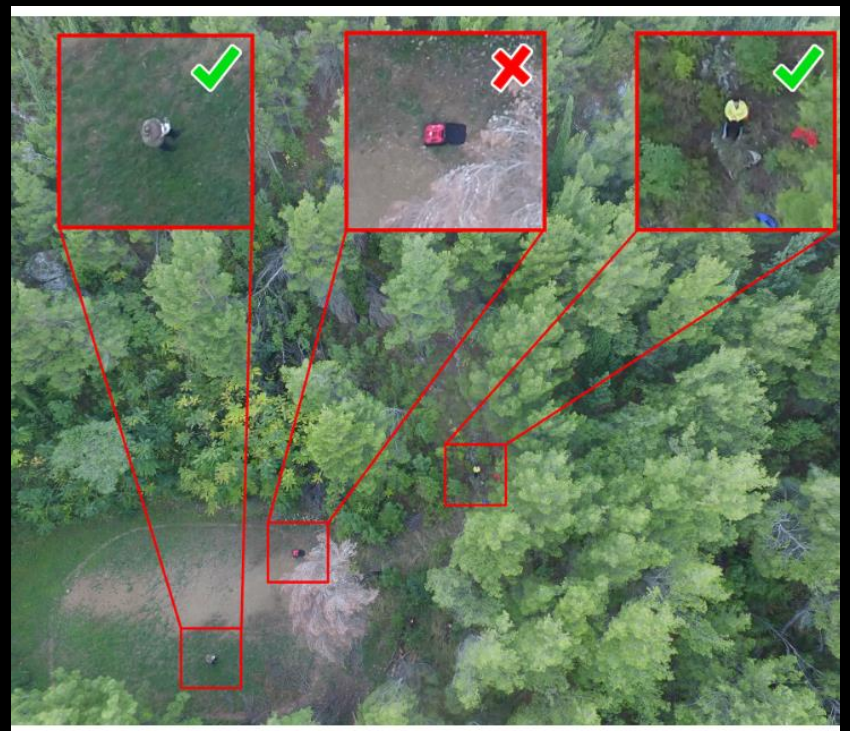
Drone-Based Search and Rescue with Aerial Person Detection [4]

- Aerial drone footage inspection is crucial for land search and rescue (SAR) operations. However, manually inspecting footage is slow, tedious, and prone to errors.
- The researchers incorporated various techniques such as model architecture selection, online data augmentation, transfer learning, and image tiling to enhance performance.

[4] “Rethinking Drone-Based Search and Rescue with Aerial Person Detection,” ar5iv. Accessed: Mar. 15, 2024. [Online]. Available: https://ar5iv.labs.arxiv.org/html/2111.09406

Drone-Based Search and Rescue with Aerial Person Detection [4]

- This research is adjoint to ours as this is **also aims to automate something that is manually very laborious, and sometimes outright impossible**.
- It also aims to identify humans in an environment that's particularly difficult to isolate.



ample of visual search problem in SAR context. As evident from the image, it is re
vithout distinctive colors from still, high-resolution images. The aerial image is taker

[4] “Rethinking Drone-Based Search and Rescue with Aerial Person Detection,” ar5iv. Accessed: Mar. 15, 2024. [Online]. Available: https://ar5iv.labs.arxiv.org/html/2111.09406

Drone-Based Search and Rescue with Aerial Person Detection [4]

Model	PRC (%)	RCL (%)	AP (%)	ATI (s)
SAR-APD evaluation (ours)				
AIR with NMS (ours)	90.5	87.8	86.5	1
AIR with MOB (ours)	**94.9**	**92.9**	**91.7**	1

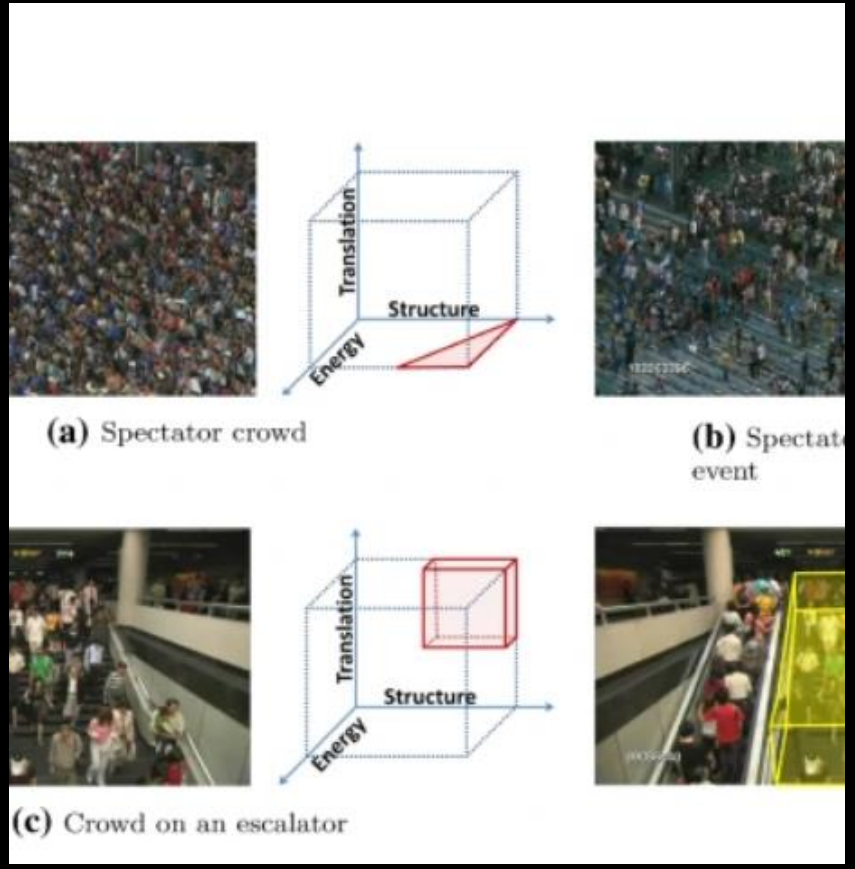
Bounding boxes are rectangular regions drawn around detected objects in images or frames of a video. They define the spatial extent of the objects and are commonly used in object detection tasks to localize and identify objects within an image. MOB (Merging of Overlapping Bounding Boxes) and NMS (Non-Maximum Suppression) are key techniques in object detection. MOB merges overlapping bounding boxes to reduce redundancy, while NMS removes redundant boxes post-detection based on confidence scores and overlap.

- PRC: Precision
- RCL: Recall
- AP: Average Precision
- ATI: Average Time per Image

- AIR: Aerial Inspection Retinanet (their new algorithm)
- MOB: Merging of Overlapping Bounding Boxes
- NMS: Non-Maximum Suppression

Multi-scale Crowd Feature Detection using Vision Sensing and Statistical Mechanics Principles [7]

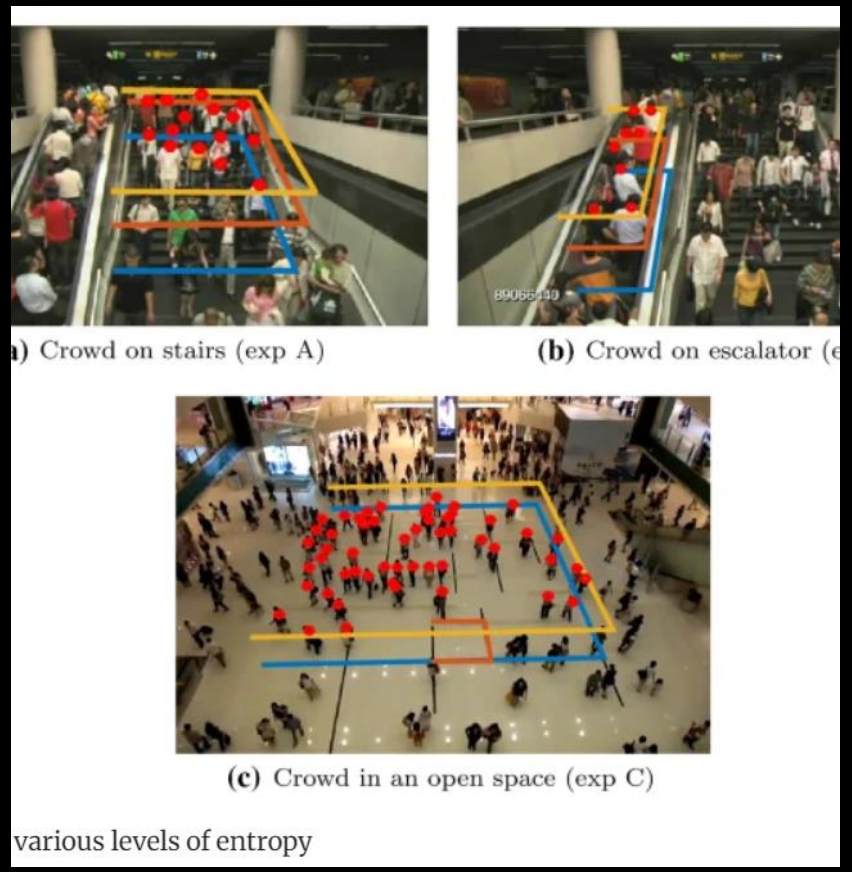
- **Previously, researchers have explored various methodologies for analyzing crowd behavior through visual data.**
- **In this paper, Zayar et al. drew analogies between human crowds and molecular thermodynamic systems. A novel descriptor that utilizes the concept of Entropy to gauge crowd movement effectively was created and tested.**



[7] B. Arbab-Zavar and Z. A. Sabeur, "Multi-scale crowd feature detection using vision sensing and statistical mechanics principles," Machine Vision and Applications, vol. 31, no. 4, p. 26, Apr. 2020, doi: 10.1007/s00138-020-01075-4

Multi-scale Crowd Feature Detection using Vision Sensing and Statistical Mechanics Principles [7]

- While we must confess we did not entirely understand the finer aspects of this paper, we did feel the researchers' approach offered a promising avenue for machine comprehension of crowd behavior, providing new complementary capabilities to existing crowd descriptors. Particularly so because they actually demonstrate its efficacy in analyzing spectator crowds.



[7] B. Arbab-Zavar and Z. A. Sabeur, “Multi-scale crowd feature detection using vision sensing and statistical mechanics principles,” Machine Vision and Applications, vol. 31, no. 4, p. 26, Apr. 2020, doi: 10.1007/s00138-020-01075-4

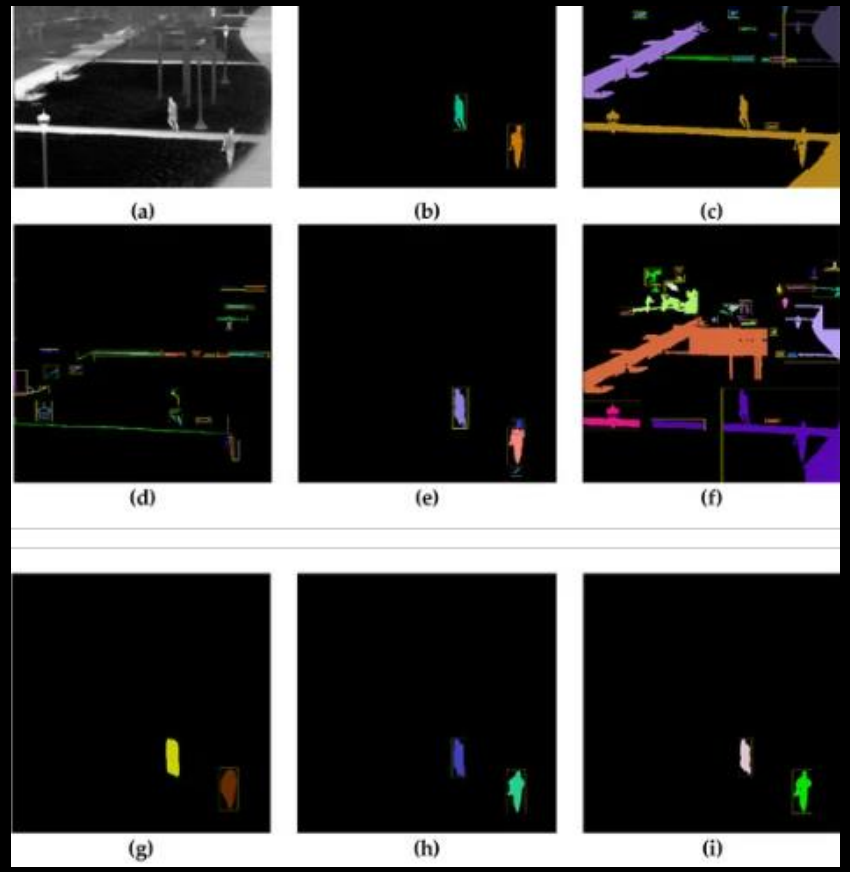
Multi-scale Crowd Feature Detection using Vision Sensing and Statistical Mechanics Principles [7]

- The data was compiled during a match at the Anoeta Stadium in San Sebastain, Spain.
- Macro K Energy: Macro Kinetic Energy

Crowd Behaviour Diagnostics			
Time	Metric	Observed	Expecte
15:17:18	Cluster Structure	low	high
15:17:18	Cluster Flow	high	low
15:17:18	Macro K Energy	low	low
15:17:18	Collectiveness	high	low
15:17:17	Cluster Structure	--	high
15:17:17	Cluster Flow	--	low
15:17:17	Macro K Energy	low	low
15:17:17	Collectiveness	medium	low
15:17:16	Cluster Structure	--	high
15:17:16	Cluster Flow	--	low
15:17:16	Macro K Energy	low	low
15:17:16	Collectiveness	low	low

CNN-Based Person Detection Using Infrared Images for Night-Time Intrusion Warning Systems [8]

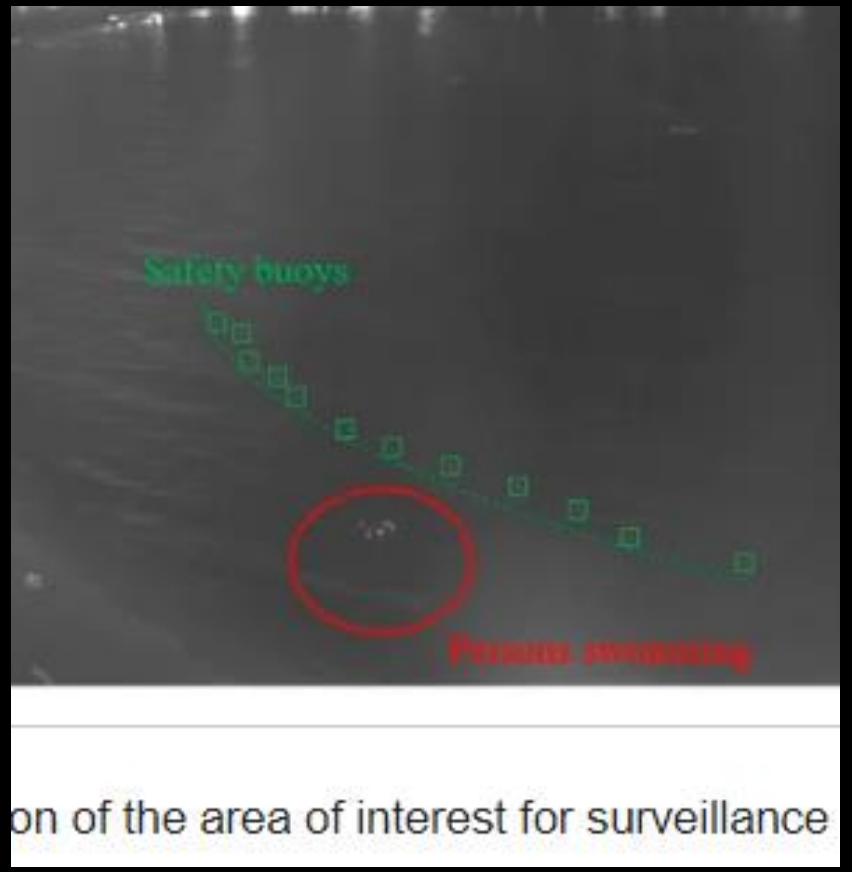
- **We came across this study when we were looking at past research in detection in substandard imagery.**
- Even if the drone's camera isn't very good, or its damaged, or we simply cannot focus on certain parts of the mob and have to raise altitude in order to monitor the entire mob, **we wanted to ensure our footage and models are capable of extracting the necessary features.**



[8] J. Park, J. Chen, Y. K. Cho, D. Y. Kang, and B. J. Son, "CNN-Based Person Detection Using Infrared Images for Night-Time Intrusion Warning Systems," Sensors, vol. 20, no. 1, Art. no. 1, Jan. 2020, doi: 10.3390/s20010034.

CNN-Based Person Detection Using Infrared Images for Night-Time Intrusion Warning Systems [8]

- Usage of infrared cameras for security widespread. Challenges due to their overhead positioning and the small regions occupied by people in resulting images. This study tried a method aimed at accurately and efficiently detecting people in infrared CCTV images at night. To achieve this, **three distinct infrared image datasets were compiled, including footage from a public beach and a pedestrian bridge** captured by a forward-looking infrared (FLIR) camera.



[8] J. Park, J. Chen, Y. K. Cho, D. Y. Kang, and B. J. Son, "CNN-Based Person Detection Using Infrared Images for Night-Time Intrusion Warning Systems," Sensors, vol. 20, no. 1, Art. no. 1, Jan. 2020, doi: 10.3390/s20010034.

CNN-Based Person Detection Using Infrared Images for Night-Time Intrusion Warning Systems [8]

- Simple thresholding: sets a constant threshold to identify pixels corresponding to humans based on their typically higher temperature.
- Adaptive thresholding: dynamically adjusts the threshold based on local intensity values, making it more resilient to lighting changes.
- Background subtraction: relies on the observation that humans typically move in contrast to the static background. By detecting pixel changes over time, humans can be distinguished from the background. This study employs a background subtraction technique based on Gaussian mixture models to represent pixel intensity variations.
- K-means clustering method: segment image regions with distinct intensity distributions. By identifying separate distributions for foreground (e.g., humans) and background pixels, K-means facilitates the separation of these elements.

Table 4. Pixel-level detection score for the beach datase

Method	Precision	Recall
mple thresholding	0.013	0.971
aptive thresholding	0.802	0.601
kground subtraction	0.214	0.590
-means clustering	0.687	0.725
Baseline CNN	0.510	0.471
Our method	0.750	0.796
Our method*	0.645	0.760

(*) indicates that temporal input is used.

Table 5. Object-level detection score for the beach datas

Method	Precision	Recall
mple thresholding	0.131	0.953
aptive thresholding	0.769	0.836
kground subtraction	0.313	0.661
-means clustering	0.622	0.684
Baseline CNN	0.971	0.585
Our method	1.000	0.877
Our method*	0.961	0.860

(*) indicates that temporal input is used.

Our Project, subject to this review

However, there are several challenges that have not yet been addressed. Drones are now manually flown and directed to remain in course with crowds. There are little-to-no failsafes in case of operator failure or failure in transmitting-receiving equipment. Also, obviously, there is a latency in the operator receiving the footage & actually judging the environment. Currently, military drones are used often, and quite effectively in Search & Rescue Operations in theatres of war abroad. However, they pose their own technical, political & ethical concerns because of which they can't be used domestically.

Grainy images of almost static environments are much, much easier to process to identify anomalies than the feed from a drone a couple hundred feet in the air.

We plan on automating the entire process of checking the sparsity and violence associated with a crowd and alerting the needed authorities within the drone's own hardware.

Pictorial Depiction

Drone-based Crowd Survillence

Dataset & Feature Preprocessing

Real Life Violence Situations Dataset [9] [10]

- This dataset contains 1000 videos indicative of violent actions and 1000 videos indicative of non-violent actions. They are collected from a wide variety of sources, including but not restricted to YouTube videos, many real street fights situations in several environments and conditions, CCT and actual drone videos
- The dataset has been published as having two sub-folders: one labelled 'Violent' and the other labelled 'Non-violent'.
- This dataset was collated as part of a research paper to construct an end-to-end deep neural network model for the purpose of recognizing violence in videos.

[9] "Real Life Violence Situations Dataset." Accessed: Mar. 15, 2024. [Online]. Available: https://www.kaggle.com/datasets/mohamedmustafa/real-life-violence-situations-dataset

[10] M. M. Soliman, M. H. Kamal, M. A. El-Massih Nashed, Y. M. Mostafa, B. S. Chawky, and D. Khattab, "Violence Recognition from Videos using Deep Learning Techniques," in 2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS), Dec. 2019, pp. 80–85. doi: 10.1109/ICICIS46948.2019.9014714.

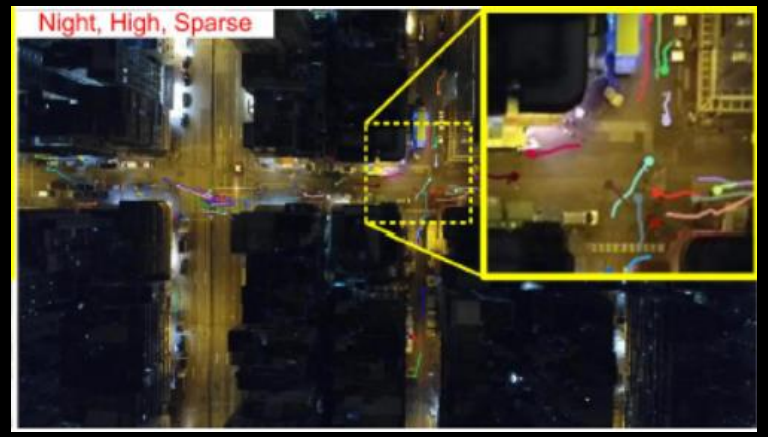
Real Life Violence Situations Dataset Preprocessing [9] [10]

- **Setting Parameters**:
  - **dataset_path**, **classes['violence','non-violence']**, **frames_per_video =10**, **video_per_class_count=1000**, and **num_classes =2** are initialized. These parameters define the location of the dataset, the categories, the number of frames to be sampled from each video, the number of videos to be taken from each class , and the number of classes respectively.
  - In total 20,000 frames are extracted from the videos.
- **Frame Extraction from Videos** (**extract_frames_from_video** function):
  - For each video, this function captures a fixed number (**frames_per_video**) of frames evenly spaced throughout the video. The frames are resized to a uniform shape of 224x224 pixels to ensure consistency in input size for the CNN.
- **Dataset Creation** (**create_datasets** function):
  - This function prepares the dataset for the model by iterating over each class directory, sampling a specified number (**sample_count**) of video files, and extracting frames using the **extract_frames_from_video** function.
  - The frames are stored in an array **X**, and their corresponding labels in an array **y**.

Real Life Violence Situations Dataset Preprocessing [9] [10]

- **Splitting Dataset**:
  - The collected frames (**X**) and labels (**y**) are split into training, validation, and test sets using the **train_test_split** function. This ensures that the model is trained on one subset of the data, validated on another, and finally tested on a different subset to evaluate its performance.
- **Data Wrapping in Dataset Class** (**ViolenceDataset** class):
  - This custom class extends PyTorch's **Dataset** class. It prepares the frames and labels for use in a DataLoader, which allows for efficient batching and shuffling of data during model training. The frames are converted to tensors and their dimensions are permuted to match the expected input structure for PyTorch's CNNs (channel, height, width).
- **DataLoader Preparation**:
  - **DataLoader** objects for training, validation, and test sets are initialized to manage batches of data, facilitating more efficient computation by enabling parallel processing and reducing memory overhead during model training.

Drone Crowd Dataset



This dataset consists of 112 video clips with 33,600 high resolution (i.e., 1920x1080) frames captured in 70 different scenarios. The researchers very laboriously provided 20,800 people trajectories with 4.8 million head annotations and several video-level attributes in sequences.

The published dataset contains two subfolders: one with the images/frames and another with ground truth of the same

Image Dataset of 33,600 images out of which 13, 500 represent packed crowds and 20, 100 represent sparse crowds. The threshold for determining this was set at 150.

https://github.com/VisDrone/DroneCrowd/tree/master

Drone Crowd Dataset Preprocessing

- **Load Images and Annotations**:
  - **Images**: Load the frames from the dataset's image folder.
  - **Annotations**: Use a MATLAB-compatible library (e.g., **`scipy.io`**) to read the **`.mat`** files containing head annotations.
- **Resize Frames**:
  - To save computation and memory, resize each frame to a manageable dimension
  - Normalize pixel values to [0, 1].
- **Generate Density Maps**:
  - Use the coordinates of head annotations from the **`.mat`** files to generate Gaussian kernel-based density maps. This will help the model learn the crowd density in different regions.
  - Create density maps with the same resolution as the resized frames.
  - Implement the Gaussian kernel function to blur each head annotation, resulting in a smoother density map.

Drone Crowd Dataset Preprocessing

- **Data Augmentation**:
  - Apply augmentations to the frames (e.g., horizontal flips, slight rotations, random cropping) to increase data diversity.
  - Ensure that the same augmentations are applied to the corresponding density maps.
- **Train-Validation-Test Split**:
  - Split the dataset into training, validation, and test sets. Divide based on different scenarios or environmental conditions to simulate real-world settings.
- **Custom DataLoader**:
  - Implement a data loader that pairs each image with its corresponding density map and applies augmentations dynamically.
  - Ensure that the data loader returns batches containing both images and density maps.

ML Methodology

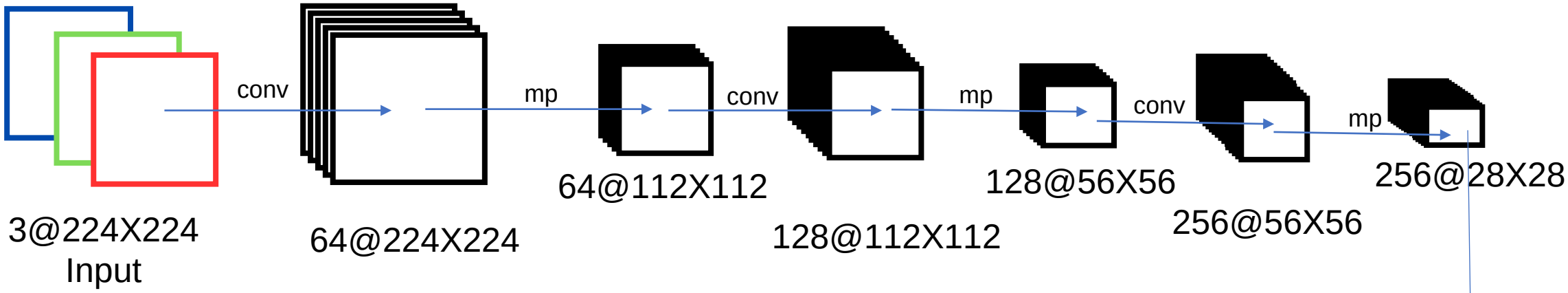
CNN

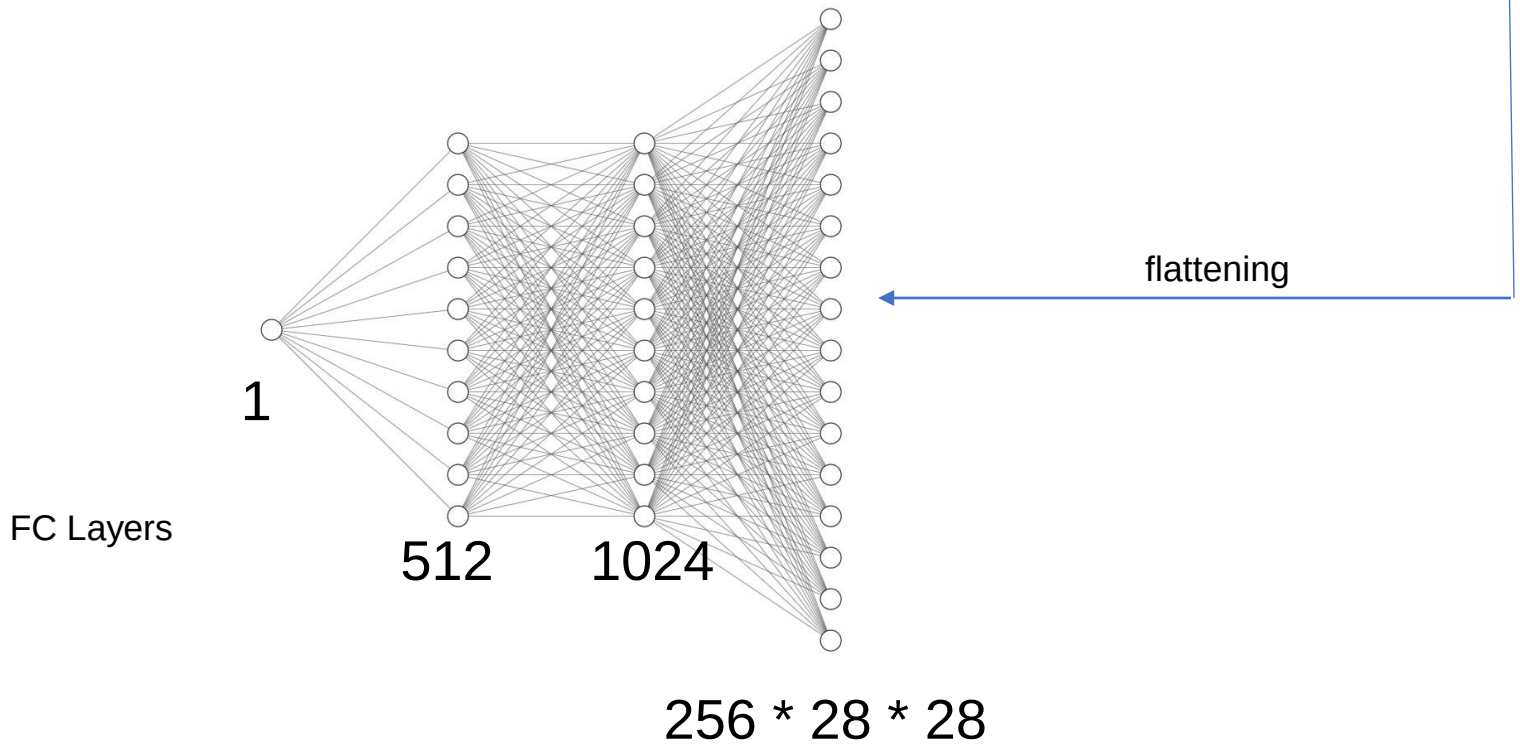
A Convolutional Neural Network (CNN) is a type of deep learning model particularly well-suited for analyzing visual data like images and videos.

Why CNN For our Models?

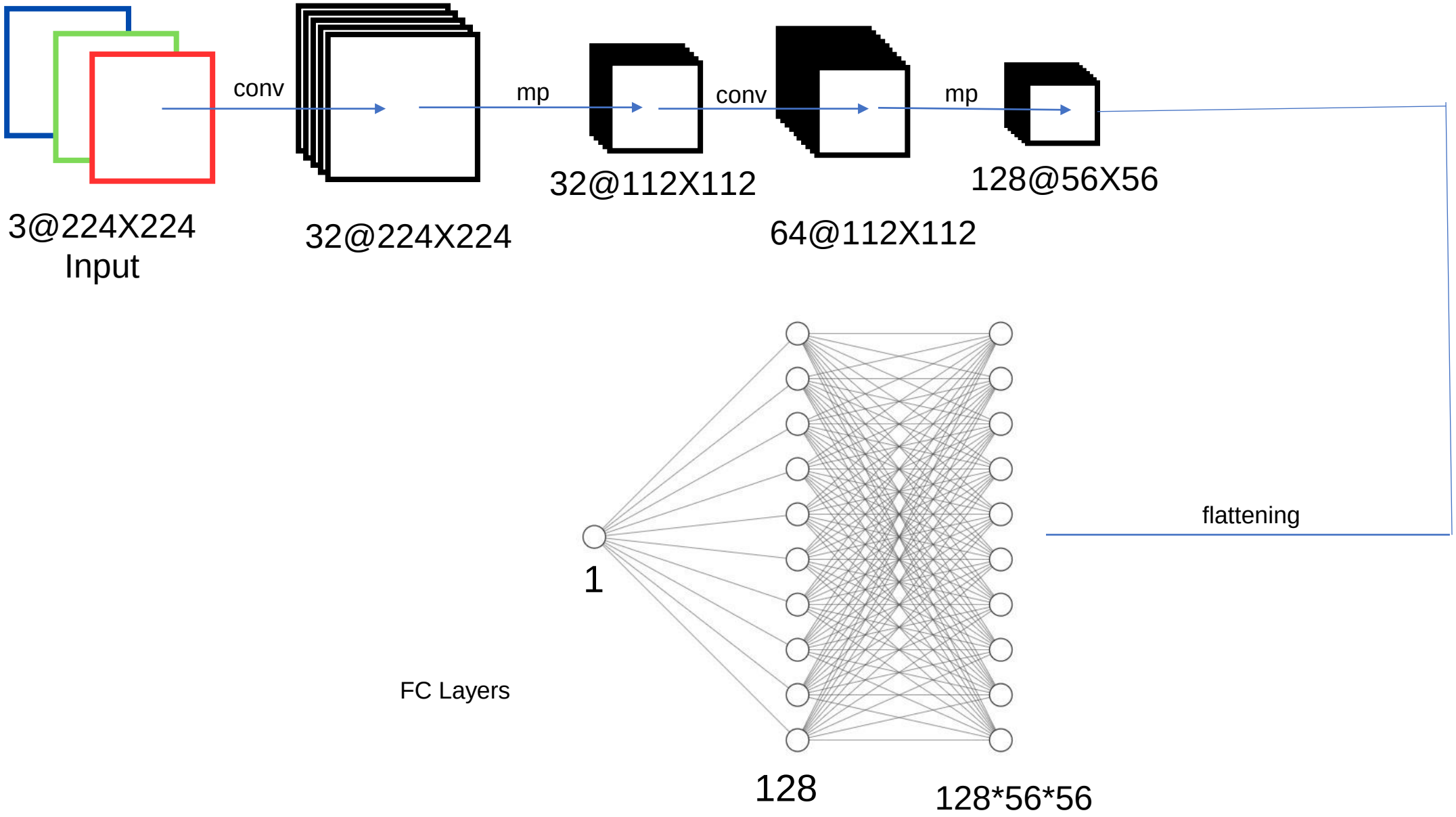
- Hierarchical Feature Learning: CNNs can learn complex and abstract patterns through their multi-layer structure.
- Parameter Sharing: Each convolutional filter (kernel) slides across the entire image to extract relevant features, reducing the number of parameters and improving efficiency
- Transfer Learning: Pre-trained CNN architectures can be fine-tuned on specialized tasks with relatively smaller labeled datasets
- Adaptability: CNNs can automatically adapt and learn the most relevant features specific to the dataset and problem.
- Spatial Hierarchy: Pooling layers within CNNs downsample features, creating a spatial hierarchy.

CrowdDensityNet





ViolenceNet



Challenges Faced

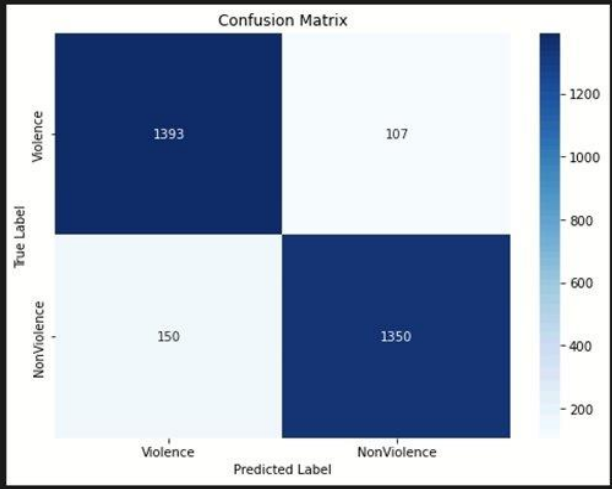
- Hardware Issue: Initially, our CPU struggled with the increasing computational demands, causing significant delays. Switching to a 4GB GPU improved but not ideal solution
- Software Issue: TensorFlow compatibility conflicts were a nightmare, with varying version requirements causing package chaos. We had to isolate dependencies in a Conda environment and switch to PyTorch, which meant extensive retraining and model conversion.
- Workflow Issues: Inconsistent workflows and environments made teamwork a headache. Google Colab eased code and google drive eased dataset sharing, while AnyDesk allowed real-time collaboration.

Performance Metrics

ViolenceNet's

TPR: 0.929
FNR: 0.071

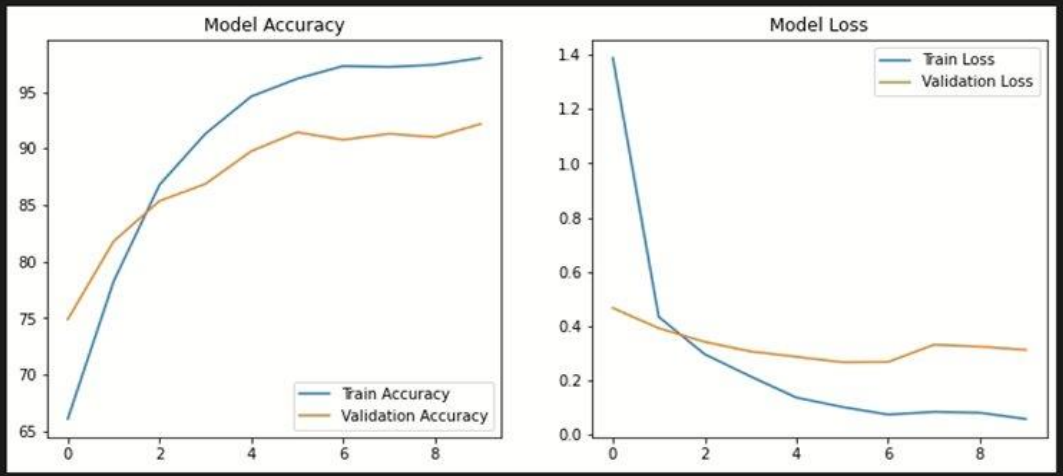
```
Epoch 1/10, Train Loss: 1.3873, Train Acc: 0.6610, Val Loss: 0.4666, Val Acc: 0.7490
Epoch 2/10, Train Loss: 0.4338, Train Acc: 0.7826, Val Loss: 0.3922, Val Acc: 0.8180
Epoch 3/10, Train Loss: 0.2964, Train Acc: 0.8679, Val Loss: 0.3419, Val Acc: 0.8537
Epoch 4/10, Train Loss: 0.2146, Train Acc: 0.9130, Val Loss: 0.3067, Val Acc: 0.8687
Epoch 5/10, Train Loss: 0.1367, Train Acc: 0.9460, Val Loss: 0.2868, Val Acc: 0.8977
Epoch 6/10, Train Loss: 0.1021, Train Acc: 0.9616, Val Loss: 0.2668, Val Acc: 0.9143
Epoch 7/10, Train Loss: 0.0741, Train Acc: 0.9729, Val Loss: 0.2679, Val Acc: 0.9077
Epoch 8/10, Train Loss: 0.0840, Train Acc: 0.9720, Val Loss: 0.3315, Val Acc: 0.9130
Epoch 9/10, Train Loss: 0.0807, Train Acc: 0.9740, Val Loss: 0.3243, Val Acc: 0.9100
Epoch 10/10, Train Loss: 0.0578, Train Acc: 0.9799, Val Loss: 0.3124, Val Acc: 0.9217
```




```
...                precision    recall  f1-score   support

       Violence       0.90      0.93      0.92      1500
    NonViolence       0.93      0.90      0.91      1500

       accuracy                           0.91      3000
      macro avg       0.91      0.91      0.91      3000
   weighted avg       0.91      0.91      0.91      3000
```




```
Final Validation Accuracy: 92.17%
```

CrowdDensityNet's

```
Epoch 1/10, Train Loss: 2.5410, Train Acc: 0.3420, Val Loss: 2.6550, Val Acc: 0.3350
Epoch 2/10, Train Loss: 2.4382, Train Acc: 0.3780, Val Loss: 2.5402, Val Acc: 0.3480
Epoch 3/10, Train Loss: 2.3021, Train Acc: 0.4100, Val Loss: 2.4083, Val Acc: 0.3660
Epoch 4/10, Train Loss: 2.1589, Train Acc: 0.4450, Val Loss: 2.2891, Val Acc: 0.3920
Epoch 5/10, Train Loss: 2.0308, Train Acc: 0.4810, Val Loss: 2.1594, Val Acc: 0.4170
Epoch 6/10, Train Loss: 1.9094, Train Acc: 0.5090, Val Loss: 2.0403, Val Acc: 0.4350
Epoch 7/10, Train Loss: 1.7935, Train Acc: 0.5340, Val Loss: 1.9246, Val Acc: 0.4510
Epoch 8/10, Train Loss: 1.6802, Train Acc: 0.5570, Val Loss: 1.8124, Val Acc: 0.4690
Epoch 9/10, Train Loss: 1.5723, Train Acc: 0.5800, Val Loss: 1.7032, Val Acc: 0.4870
Epoch 10/10, Train Loss: 1.4691, Train Acc: 0.6030, Val Loss: 1.5984, Val Acc: 0.5050
```

Mean Absolute Error:

$$\frac{1}{\sum_{i=1}^{M} N_i} \sum_{i=1}^{M} \sum_{j=1}^{N_i} |z_{i,j} - \hat{z}_{i,j}$$

Mean Squared Error:

$$\sqrt{\frac{1}{\sum_{i=1}^{M} N_i} \sum_{i=1}^{M} \sum_{j=1}^{N_i} |z_{i,j} - \hat{z}_{i,j}|^2}$$

```
Performance Matrix
MAE: 19.8
MSE: 19.25
```

A deployable solution?

- Using Transfer Learning, we should be able to get our models working on drone imagery from great heights so that we don’t have to redesign the model entirely. Currently, the dataset is only part video.
- Our models will work even with basic drones with a camera of sufficient resolution, although the extent & accuracy may vary.
- Some major foreseeable problems include issues with the Right to Privacy and filming people in a public space, rioters identifying the drone and pelting it and the varying levels of what is considered to be violent in different parts of the country (and indeed the world).

Future Scope

- The development of more models for functions like backtracking through the frames to pinpoint the perpetrator of some crime (like a suicide bombing) automatically, dynamic path planning of the drone to mimic crowd movement and the identification of individual instances of violent behaviour within the crowd itself.
- The development of a decision engine to take the output of all these models as an input and combine it to create one comprehensive output set that will then be processed to take decisions like whether to lower the drone or alert the authorities.

Future Scope

- We need to test the drone in real life for violent and non-violent crowds and verify the accuracy and efficiency of our models (and the decision engine).
- We need to check for latency as well. It must not turn out that it takes more time to run the ML model than manually verify the drone feed. Right now, the models are rather efficient, but the accumulation of their outputs and subsequent entry into the decision engine slightly complicated things and increases the time required for computation.
- Testing on both high-end and low-end drones to check if there is any difference and if yes, if it can be mitigated via software.


Thank you!