



iJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 Issue: VIII Month of publication: August 2021

DOI: <https://doi.org/10.22214/ijraset.2021.36414>

www.ijraset.com

Call:  08813907089

E-mail ID: ijraset@gmail.com

Estimation of Crowd Density from UAV Images based on Deep Learning

Sarita Chauhan¹, Ashwani Kumar Jha², Kunal Mathur³, Tanmay Sharma⁴ Payal Sharma⁵

¹Asst.Prof, ^{2,3,4,5}Student, Dept. Of Electronics and Communication Engineering, MLV Textile and Engineering College, Rajasthan, India

Abstract: Crowd monitoring is necessary to improve safety and controllable movements to minimize risk, especially in high crowded events, such as Kumbh Mela, political rallies, sports event etc. In this current digital age mostly crowd monitoring still relies on outdated methods such as keeping records, using people counters manually, and using sensors to count people at the entrance. These approaches are futile in situations where people's movements are completely unpredictable, highly variable, and complex. Crowd surveillance using unmanned aerial vehicles (UAVs), can help us solve these problems. The proposed paper uses a UAV on which an IP Camera will be attached to get media, we then use a convolutional neural network to learn a regression model for crowd counting, the model will be trained extensively by using three widely used crowd counting datasets, ShanghaiTech part A and part B, UCF-CC 50 and UCF-QNRF.

Keywords: Deep Learning, Neural Network, Computer Vision, Image Processing, UAV.

I. INTRODUCTION

Crowd counting is a fairly emerging interdisciplinary issue that has attracted the attention of sociologists, psychologists, engineers, and scientists in recent years. The exponential growth of the world's population, along with rapid urbanisation, has resulted in a rise in the incidence of unusually concentrations of population. They are caused by a variety of factors, including social events like the Kumbh Mela, sporting events, and political rallies.

Video surveillance for security purposes; overcrowding detection for disaster management, public safety design, and traffic monitoring; simulation studies for a better understanding of crowd dynamics, and so on are all important uses of crowd analysis. Counting people and estimating their density are two of the most important challenges in crowd analysis.

The task of counting the number of people in a scene is referred to as crowd counting, whereas crowd density estimate is the prediction of the equivalent density map. In this digital age, most crowd monitoring still relies on traditional methods like storing records, manually counting people, and counting people at the entry with sensors. In instances where people's movements are totally unpredictable, extremely variable, and complex, these tactics are useless.

Crowd monitoring employing unmanned aerial vehicles (UAVs) may be able to assist us in resolving these issues. The suggested solution employs a UAV equipped with an IP camera to collect media, with the obtained photos of the crowd area being sent over to communication infrastructure for further analysis.

The photos are segmented and categorised in the second stage to estimate crowd density. This involves combining well-known picture segmentation techniques such background removal (Davies, Yin, and Velastin 1995), image processing and pattern recognition (Marana et al. 1998), information fusion (Velastin et al. 1994), and feature extraction points to extract crowd features (Conte et al. 2010).

In the last step, a convolutional neural network will be used to construct a crowd counting regression model. The model will be extensively trained using three popular crowd counting datasets.

II. LITERATURE REVIEW

A. Crowd Counting Datasets

Existing crowd counting datasets can be split into three categories based on image acquisition methods: surveillance-view datasets, free-view datasets, and drone-view datasets. Surveillance-view datasets are compiled from crowd photos captured by surveillance cameras in discrete indoor settings or small-area outdoor sites.

Typical surveillance-view datasets include UCSD, Mall, WorldExpo, and ShanghaiTech Part B. Images from the Internet are included in free-view datasets. These datasets have a wide range of characteristics. Many free-view datasets for evaluation criteria are also available, including UCF CC 50, UCF-QNRF, and ShanghaiTech Part A & B.

B. Crowd Counting Methods

- 1) *Detection-based Methods:* A moving window-like detector is used to recognise and count the number of persons in an image. The detection approaches involve well-trained classifiers capable of extracting low-level characteristics. Although these algorithms work well for detecting faces, they struggle in crowded photos since the majority of the target items are obscured.
- 2) *Regression-based Methods:* Low-level characteristics could not be extracted using the above strategy. Here methods based on regression come out on top. Crop the image into patches, then extract the low-level features for each patch. This approach usually consists of two primary components. The extraction of low-level features such as foreground features, texture, edge features, and gradient features is the initial step. The second component is converting the collected features into counts using a regression function, such as linear regression, piecewise linear regression, ridge regression, or Gaussian process regression.
- 3) *Density Estimation-based Methods:* In this the first step is to construct a density map for the objects. The method then creates a linear mapping between the extracted features and their corresponding object density maps. Random forest regression can also be used to learn non-linear mapping.
- 4) *CNN-based Methods:* It's the tried-and-teste way to use convolutional neural networks (CNNs). Instead of looking at individual image patches, we use CNNs to create an end-to-end regression technique. This method uses the complete image as input and generates a crowd count directly. CNNs perform exceptionally well in regression and classification applications, as well as in the generation of density maps.

III. PROPOSED SYSTEM FLOW CHART

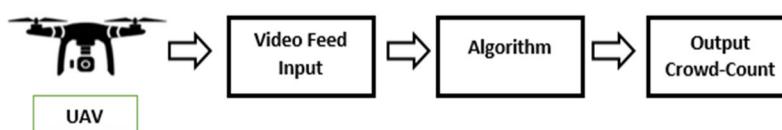


Fig. 1 Block diagram

The following design (Fig. 1) is proposed to overcome the challenges in the existing method and to produce a better solution. This project has both software and hardware requirement.

IV. SOFTWARE & HARDWARE DESIGN

A. Unmanned Aerial Vehicle

An unmanned aerial vehicle, sometimes known as a drone, is an aircraft that does not have a human pilot, crew, or passengers on board. Unmanned aerial vehicles (UAVs) are part of an unmanned aircraft system, which also includes a ground-based controller and a communications system with the UAV. UAV flight can be controlled remotely by a human operator, as in a remotely piloted aircraft (RPA), or with varying degrees of autonomy, such as autopilot help, up to fully autonomous aircraft with no human interaction. UAVs were created in the twentieth century for military duties that were "too dull, unclean, or dangerous" for people. Control technologies have improved and costs have decreased, therefore their use in the twenty-first century is fast expanding. Aerial photography, product deliveries, agribusiness, enforcement and monitoring, infrastructure inspections, science, and drone racing are just some of the applications.

These are the essential components required

- 1) *Motors:* Brushless DC motors are used in the majority of quadcopters. Brushless motors have a rotor with a permanent magnet and a number of electromagnets (also known as poles) surrounding it. Brushless motors can have anything between two and fourteen poles. The more poles the motor has, the more accurately it can be controlled. Brushless motors contain three wires that regulate the three phases of the motor. We can make the motor spin clockwise or counter clockwise by adjusting the connections on these wires.



Fig. 2 Brushless Motor

- 2) *Electronic Speed Controllers (ESC)*: Electronic Speed Controller, or ESC, is the device that controls Brushless DC motors. There will be three sets of wires on the ESC. The three wires of your brushless motor are connected by three heavy-gauge wires. Two more heavy-gauge wires run from your power distribution board to the ESC and motors, supplying voltage. There will be three smaller wires that connect to our flight controller.

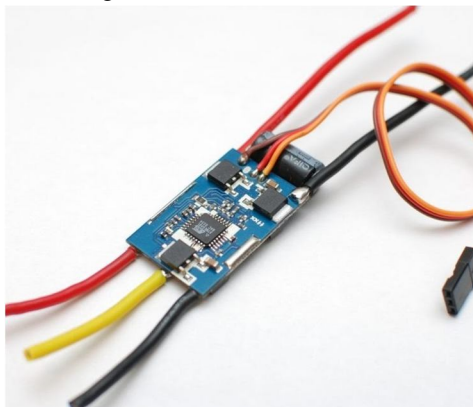


Fig. 3 Electronic Speed Controllers

- 3) *Flight Controller*: Our quadcopter's brain is the Flight Controller. This is the device that sends signals to our ESCs to control the speed of our motors. The onboard radio receiver will send signals to the flight controller, allowing us to control our quadcopter remotely. The ESCs then send signals to control the motor speeds.

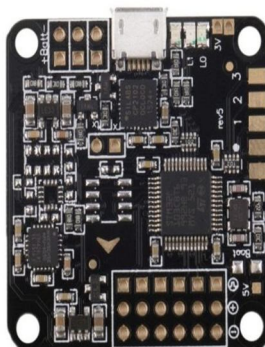


Fig. 4 Flight Controller

- 4) *Radio Transmitters & Receivers*: A handheld radio transmitter will be used to control the quadcopter, which will have an on-board radio receiver. The number of channels that the receiver and transmitter support, as well as the frequency at which they operate, are both specified. Each radio channel in the quadcopter's system controls a particular function or component. To control a quad, we'll need at least six channels, though most users start with more. Extra channels can be utilised to control illumination or to position a camera using a gimbal motor. One channel for throttle, one channel for turning right and left, one channel for pitching forward and backwards, and one channel for rolling left and right in other words, throttle, yaw, pitch, and roll are employed. A channel can also be used to change between flight modes.



Fig. 4 Radio Transmitters & Receivers

- 5) *Batteries:* We need a battery for quadcopter to fly. Larger batteries have more capacity, allowing for a longer flight time. However, as the battery's capacity increases, so does its weight, and adding weight to your quadcopter will reduce flying time because our motors will need to use more energy to hoist the payload.



Fig. 5 Batteries

So, these were the basic elements of a quadcopter. By putting all these pieces together, we design a quadcopter.



Fig. 6 Quadcopter

B. Software Design

The Histogram of Oriented Gradient Descriptor (HOG) is a feature descriptor for object detection in computer vision and image processing. This is one of the most often used methods (algorithm) for detecting objects.

- 1) *Importing the libraries*

```
import cv2
import imutils
import numpy as np
import argparse
```

Fig. 7 Installed Libraries

- 2) *Model creation for human detection*

```
HOGCV = cv2.HOGDescriptor()
HOGCV.setSVMDetector(cv2.HOGDescriptor_getDefaultPeopleDetector())
```

Fig. 8 Model Creation

3) Detect () Method

```
def detect(frame):
    bounding_box_cordinates, weights = HOGCV.detectMultiScale(frame,
    winStride = (4, 4), padding = (8, 8), scale = 1.03)

    person = 1
    for x,y,w,h in bounding_box_cordinates:
        cv2.rectangle(frame, (x,y), (x+w,y+h), (0,255,0), 2)
        cv2.putText(frame, f'person {person}', (x,y),
        cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0,0,255), 1)
        person += 1

    cv2.putText(frame, 'Status : Detecting ', (40,40),
    cv2.FONT_HERSHEY_DUPLEX, 0.8, (255,0,0), 2)
    cv2.putText(frame, f'Total Persons : {person-1}', (40,70),
    cv2.FONT_HERSHEY_DUPLEX, 0.8, (255,0,0), 2)
    cv2.imshow('output', frame)

    return frame
```

Fig. 9 Detect Method

4) Human Detector() Method

```
def detectByCamera(writer):
    video = cv2.VideoCapture(0)
    print('Detecting people...')

    while True:
        check, frame = video.read()

        frame = detect(frame)
        if writer is not None:
            writer.write(frame)

        key = cv2.waitKey(1)
        if key == ord('q'):
            break

    video.release()
    cv2.destroyAllWindows()
```

Fig. 10 Human Detector Method

5) Argparse() method

The function argparse() simply parses and returns the arguments supplied to our script through your terminal as a dictionary. Within the Parser, there will be three arguments:

- a) *Image*: The location of the image file on your computer.
- b) *Video*: The location of the video file on your computer.
- c) *Camera*: A variable if set to 'true' will call the Detect() method.

```
def argsParser():
    arg_parse = argparse.ArgumentParser()
    arg_parse.add_argument("-v", "--video", default=None, help="path
to Video File ")
    arg_parse.add_argument("-i", "--image", default=None, help="path
to Image File ")
    arg_parse.add_argument("-c", "--camera", default=False, help="Set
true if you want to use the camera.")
    arg_parse.add_argument("-o", "--output", type=str, help="path to
optional output video file")
    args = vars(arg_parse.parse_args())

    return args
```

Fig. 11 Argparse Method

V. RESULTS

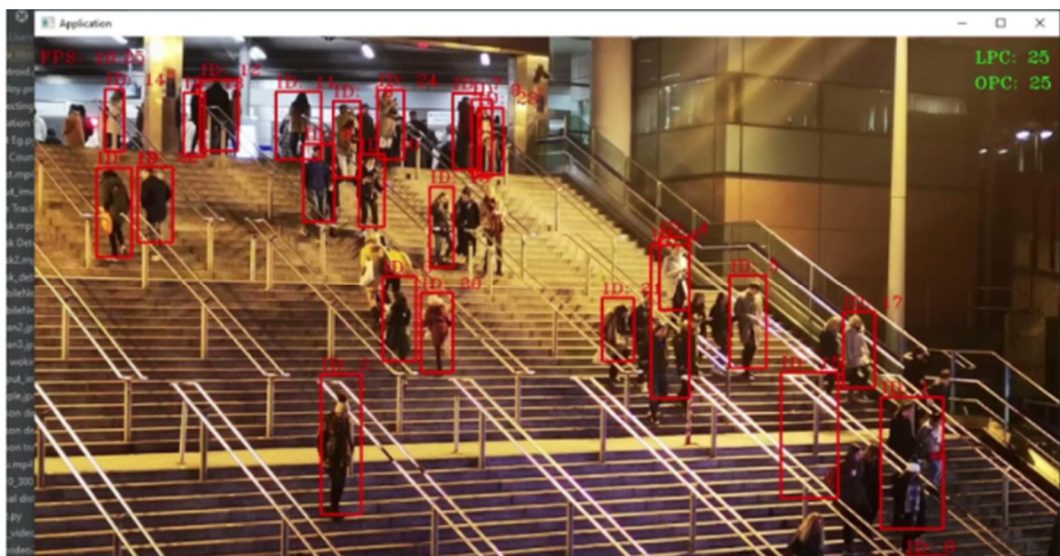
Now, after running the code with multiple images and video as input, we will get our output



a) Human Detection from Image as an Input



b) People Count from Image as an Input



c) Live People Count from Video as an Input

VI. CONCLUSION

This work proposes a method for crowd counting in near real time using NN. Counting through the use of a video feed acquired through uav can be deployed for disaster management, emergency evacuation, and large public gatherings without having to setup explicit systems for the same.

The project has a large potential scope in the future and can be implemented using satellite imagery in the future. The project is expandable, and it may be used to track or study crowd movement, which could be useful in managing riots, rallies, and other events. The proposed system architecture can also be used to create density maps for cars to monitor real-time traffic.

REFERNCES

- [1] Zhang, Yingying, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma. (2016) "Single-image crowd counting via multi-column convolutional neural network." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 589-597
- [2] Deepak Babu Sam, Shiv Surya, and R. Venkatesh Babu. Switching convolutional neural network for crowd counting. In IEEE Conference on Computer Vision and Pattern Recognition, pages 4031–4039, 2017.
- [3] Deepak Babu Sam, Shiv Surya, and R. Venkatesh Babu. Switching convolutional neural network for crowd counting. In IEEE Conference on Computer Vision and Pattern Recognition, pages 4031–4039, 2017.
- [4] Vandoni, E. Aldea, and S. Le Hégarat-Mascle. Evaluating crowd density estimators via their uncertainty bounds. In 2019 IEEE International Conference on Image Processing (ICIP), pages 4579–4583. IEEE, 2019.
- [5] Wan, W. Luo, B. Wu, A. B. Chan, and W. Liu. Residual regression with semantic prior for crowd counting. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4036–4045, 2019.
- [6] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-image crowd counting via multi-column convolutional neural network," in CVPR, 2016, pp. 589–597
- [7] Bahmanyar, R., Vig, E., Reinartz, P.: MRCNet: crowd counting and density map estimation in aerial and ground imagery. CoRR abs/1909.12743 (2019)
- [8] Chen, X., Bin, Y., Sang, N., Gao, C.: Scale pyramid network for crowd counting. In: WACV, pp. 1941–1950 (2019)
- [9] Boyu Wang, Huidong Liu, Dimitris Samaras, and Minh Hoai. Distribution matching for crowd counting. In NeurIPS, 2020.
- [10] Meng-Ru Hsieh, Yen-Liang Lin, and Winston H. Hsu. Drone-based object counting by spatially regularized regional proposal network. In ICCV, 2017.
- [11] Sindagi, V.A., Patel, V.M.: CNN-based cascaded multi-task learning of high-level prior and density estimation for crowd counting. In: 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS). pp. 1–6. IEEE(2017)
- [12] Sindagi, V.A., Patel, V.M.: A survey of recent advances in CNN-based single image crowd counting and density estimation. Pattern Recognition Letters 107, 3–16 (2018)
- [13] Zhu, P., Wen, L., Bian, X., Ling, H., Hu, Q.: Vision meets drones: A challenge. arXiv preprint arXiv:1804.07437 (2018)

AUTHORS PROFILE

- 1) Smt. Sarita Chauhan is working at MLVTEC, Bhilwara in the capacity of Assistant Professor in ECE Department and has more than 18 years of experience. She has more than 23 publications in International Journals & Author of 2 books under RTU. Her field of interest are NT, VLSI, VHDL, ANFIS, NN, AE, EDA, Image processing etc
- 2) Ashwani Kumar Jha, student of Bachelor of Technology in Electronics and Communication Engineering from MLV Textile and Engineering College, Bhilwara.
- 3) Kunal Mathur, student of Bachelor of Technology in Electronics and Communication Engineering from MLV Textile and Engineering College, Bhilwara.
- 4) Tanmay Sharma, student of Bachelor of Technology in Electronics and Communication Engineering from MLV Textile and Engineering College, Bhilwara.
- 5) Payal Sharma, student of Bachelor of Technology in Electronics and Communication Engineering from MLV Textile and Engineering College, Bhilwara.



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)